AUGUST 1992

A RECONSTRUCTION OF THE WATER BALANCE IN WESTERN UNITED STATES LAKE BASINS TO CLIMATIC CHANGE

WRRI Report No. 269

VOLUME II

NEW MEXICO WATER RESOURCES RESEARCH INSTITUTE
New Mexico State University
Box 30001, Dept. 3167
Las Cruces, New Mexico 88003-0001
Telephone (505) 646-4337 FAX (505) 646-6418

A RECONSTRUCTION OF THE RESPONSE OF THE WATER BALANCE IN WESTERN UNITED STATES LAKE BASINS TO CLIMATIC CHANGE

Volume 2

by

Fred M. Phillips and Andrew R. Campbell
Co-Principal Investigators
Department of Geoscience
New Mexico Institute of Mining and Technology

and

Cynthia Kruger, Peggy Johnson, Randall Roberts and Eric Keyes
Graduate Research Assistants
Department of Geoscience
New Mexico Institute of Mining and Technology

TECHNICAL COMPLETION REPORT

Project Numbers 1345662, 1423687

August 1992

New Mexico Water Resources Research Institute

in cooperation with

Department of Geoscience
New Mexico Institute of Mining and Technology

The Research on which this report is based was financed in part by the U.S. Department of the Interior, Geological Survey, through the New Mexico Water Resources Research Institute.

DISCLAMER

The purpose of Water Resources Research Institute technical reports is to provide a timely outlet for research results obtained on projects supported in whole or in part by the institute. Through these reports, we are promoting the free exchange of information and ideas and hope to stimulate thoughtful discussion and actions that may lead to resolution of water problems. The WRRI, through peer review of draft reports, attempts to substantiate the accuracy of information contained in its reports, but the views expressed are those of the authors and do not necessarily reflect those of the WRRI or its reviewers. Contents of this publication do not necessarily reflect the views and policies of the U.S. Department of the Interior, nor does mention of trade names or commercial products constitute their endorsement by the United States government.

ABSTRACT

Changes in the water balance are among the most serious potential consequences of global climate change. Predicting future water balance fluctuations is dependent on understanding the causes of past fluctuations. In arid, hydrologically closed basins, the water balance can be quantified as the ratio of water surface area in the terminal sink or sinks to the basin drainage area. In this study, we have used the oxygen isotope content of carbonate minerals precipitated from lake waters to reconstruct lake surface-area history. We have developed a numerical water-balance and isotope-balance model to simulate the lake's isotopic evolution and thus produced a lake surface-area history.

We have reconstructed the surface-area history of two basins in the southwestern United States. At the Plains of San Agustin, New Mexico, we used the oxygen isotope content of ostracode valves to achieve a high-resolution reconstruction of the interval 36 to 15 ka (thousand years before present). At Searles Lake, California, we used oxygen isotopes in inorganic carbonate minerals to produce a water-balance history for the period 1,180 to 10 ka. Comparing the Searles record with the marine oxygen isotope chronology shows the single strongest influence on the water balance is global glacial/interglacial cycles. Thus water-balance changes can be linked directly to global climatic change.

However, we also see patterns that differ from those of the global glacial cycles. We detect unexplained long-term trends of humid and arid water balance with an apparent periodicity of about 400 kyr. We also observe that the water balance seems to be characterized by relatively stable humid and arid modes, with rapid, unstable transitions between these modes.

Key words: Climate, isotopes, lakes, arid climates, time-series analysis.

PREFACE

This report describes the methods, results, and interpretations for paleohydrological studies on two closed-basin lakes in the western United States. The purpose of the report is to thoroughly document all aspects of the study so that it can be carefully evaluated and, if desired, replicated. Due to this objective, all aspects of the study are treated in much greater detail than the casual reader might find desirable. The authors intend to publish the results in summary form in the periodical literature, and they suggest that the reader interested primarily in a concise description of the results start with those articles.

This report consists of two volumes. The first volume contains a description of the study and graphical presentation of the data and results. It also presents fairly thorough bibliographic studies of previous research in the two lake basins studied and of related experimental studies. These are intended to provide a starting point for other researchers intending to perform additional studies in the same areas, or using similar methods. We suggest that careful consideration of the table of contents will aid the reader in focusing on the aspects of the study that are of interest. The second volume contains appendices which are mostly listings of computer programs, tables of data, and tables of computer inputs and outputs. These are largely presented in graphical form in the first volume, and are intended primarily as documentation and as a data source for future numerical analyses.

The authors offer their sincere thanks to George I. Smith of the U.S. Geological Survey in Menlo Park, California. This study would not have been possible without his active participation and advice. They also thank Kerr-McGee Corporation, and particularly Gail Moulton, for permission to sample the KM-3 core at Trona, California. They gratefully acknowledge "Cato" Lee for permission to drill on his ranch in the San Agustine Basin. They thank Vera Markgraf of the University of Colorado for her advice and participation in the drilling. Other individuals who contributed include Stewart S. Smith, who assisted the sampling at Trona, John Hawley and Robert Weber, who helped locate the San Agustine drilling site, Richard Forester (U.S. Geological Survey, Denver), who shared techniques for isolating ostracodes and assisted with specimen identification, and Annette Schafer-Perini, who gave much useful advice on numerical methods.

TABLE OF CONTENTS

Volume 2

Disclaimer i
Abstract ii
Preface iii
Table of Contents iv

Appendix A Searles Lake Core KM-3 Composite Chronology 1

Appendix B Core Sampling Field Notes and X-Ray Diffraction Results 5

Appendix C Water-Leaching Procedures 20

Appendix D EDTA Dissolution Procedure 23

Appendix E Hierarchical Statistics Analysis of Uncertainty for ^{18}O and ^{13}C Analyses 27

Appendix F Searles Lake Core KM-3 Isotopic Results and Isotopic Correction Factors 54

Appendix G Searles Lake Core KM-3 Isotopic Results for Bulk Leached Samples Run at University of Missouri - Columbia 67

Appendix H Analytical-Solution Lake Isotopic Evolution Model and Results 70

Appendix I Climatological Data for Southeastern California Weather Stations 90

Appendix J Lithologic Log of San Agustin Core SAC-3/4 104

Appendix K Relative Abundance of Ostracode Species with Depth in San Agustin Core SAC-3/4 112

Appendix L San Agustin Isotopic Results 117

Appendix M Transient Isotopic Model for Paleo-Owens River System 122

Appendix N Transient Isotopic Model for Lake San Agustin 193

Appendix O Steady-State Isotopic Model 212

Appendix P Bathymetric Data for Lakes in the Paleo-Owens River 223

Appendix Q Model Inputs for Lake San Agustin 225

Appendix R Numerical Model Results for the Paleo-Owens River System 236

Appendix S Numerical Model Results for Lake San Agustin 250

Appendix A

Searles Lake Core KM-3 Composite Chronology

```
*****************************************************************
*                                                               *
*                         APPENDIX A                            *
*                                                               *
*         SEARLES LAKE CORE KM-3 COMPOSITE CHRONOLOGY           *
*                                                               *
*****************************************************************
```

AGE (Ma)	DEPTH (m)	DEPTH (ft)	DATE TYPE
0.00352	2.400	7.874	C-14
0.00600	5.791	18.999	CL-36
0.00900	12.160	39.895	AIR
0.00980	18.010	59.088	AIR
0.01000	19.930	65.387	CL-36
0.02300	25.055	82.201	C-14
0.02320	27.127	88.999	AIR
0.02560	27.981	91.801	AIR
0.02561	28.956	95.000	AIR
0.02780	29.779	97.700	AIR
0.02860	33.894	111.201	AIR
0.02890	33.985	111.499	AIR
0.02960	34.412	112.900	AIR
0.03080	34.991	114.800	AIR
0.03090	35.204	115.499	AIR
0.03170	35.479	116.401	AIR
0.03171	36.027	118.199	AIR
0.03280	36.698	120.400	AIR
0.03300	38.009	124.701	C-14
0.06550	41.780	137.073	U-Th
0.06980	43.300	142.060	U-Th
0.07780	46.020	150.984	AIR
0.08930	49.950	163.878	U-Th
0.10340	54.410	178.510	AIR
0.10520	55.170	181.004	AIR
0.12230	60.270	197.736	U-Th
0.15380	69.000	226.378	AIR
0.16050	70.860	232.480	U-Th
0.19250	82.320	270.079	U-Th
0.22300	90.830	297.999	AIR
0.23400	95.400	312.992	AIR
0.24500	96.930	318.012	AIR
0.26500	99.820	327.493	AIR
0.27400	110.030	360.991	AIR
0.27500	110.640	362.992	AIR
0.27800	112.470	368.996	AIR
0.28600	114.000	374.016	CL-36
0.31600	116.740	383.005	AIR
0.33400	124.050	406.988	AIR
0.33900	126.500	415.026	CL-36
0.34200	130.450	427.986	AIR
0.34300	130.760	429.003	AIR
0.35000	135.640	445.013	AIR
0.37400	137.160	450.000	AIR
0.37600	138.070	452.986	AIR
0.38200	138.990	456.004	AIR

0.38400	139.750	458.497	AIR
0.38580	140.210	460.007	AIR
0.38600	140.800	461.942	CL-36
0.38620	141.720	464.961	AIR
0.38900	144.780	475.000	AIR
0.41700	145.390	477.001	AIR
0.41800	146.610	481.004	AIR
0.44800	150.570	493.996	AIR
0.44900	150.970	495.308	AIR
0.45000	151.120	495.801	AIR
(0.42600	153.000	501.969	CL-36) not used
0.53600	154.100	505.577	AIR
0.57400	157.700	517.388	AIR
0.57500	157.880	517.979	AIR
0.57580	158.000	518.373	AIR
0.57680	159.000	521.654	AIR
0.57700	160.000	524.934	AIR
0.57700	160.300	525.919	AIR
0.59000	165.800	543.963	AIR
0.59800	166.420	545.997	AIR
0.61000	168.600	553.150	ASH
0.66700	178.640	586.089	AIR
0.66800	179.410	588.615	CL-36
0.73000	185.000	606.955	Paleo-Mag
0.75100	186.540	612.008	AIR
0.75200	186.900	613.189	AIR
0.77100	188.150	617.290	AIR
0.77800	189.950	623.195	AIR
0.79200	190.700	625.656	CL-36
0.79900	192.020	629.987	AIR
0.83400	196.140	643.504	AIR
0.83500	196.660	645.210	AIR
0.89600	204.520	670.997	AIR
0.90000	204.900	672.244	Paleo-Mag
0.90400	207.450	680.610	AIR
0.93200	210.920	691.995	AIR
0.93500	213.660	700.984	AIR
0.95300	218.540	716.995	AIR
0.97000	221.200	725.722	Paleo-Mag
1.00598	248.110	814.009	AIR
1.00724	249.480	818.504	AIR
1.06514	268.530	881.004	AIR
1.06936	271.420	890.486	AIR
1.08838	276.910	908.497	AIR
1.10000	281.100	922.244	Paleo-Mag
1.13288	291.080	954.987	AIR
(1.27000	293.000	961.286	CL-36) not used
1.13280	294.440	966.010	AIR
1.17242	299.310	981.988	AIR
1.17226	306.020	1004.003	AIR
1.17356	308.460	1012.008	AIR
1.20075	322.200	1057.087	Paleo-Mag
1.21385	324.310	1064.009	AIR
1.23581	325.530	1068.012	AIR
1.24570	327.960	1075.984	AIR
1.26366	329.180	1079.987	AIR
1.27661	331.320	1087.008	AIR
1.30255	333.150	1093.012	AIR
1.30749	334.670	1097.999	AIR
1.34590	337.260	1106.496	AIR

1.35535	338.630	1110.991	AIR
1.37880	340.310	1116.503	AIR
1.38627	341.070	1118.996	AIR
1.44413	345.340	1133.005	AIR
1.45801	348.690	1143.996	AIR
1.46151	349.150	1145.505	AIR
1.46598	350.220	1149.016	AIR
1.48191	351.430	1152.985	AIR
1.50684	354.180	1162.008	AIR
1.52665	359.050	1177.985	AIR
1.55356	362.100	1187.992	AIR
1.57289	367.130	1204.495	AIR
1.60230	370.030	1214.009	AIR
1.62263	374.750	1229.495	AIR
1.67000	378.560	1241.995	Paleo-Mag
1.67200	379.170	1243.996	AIR
1.67800	379.630	1245.505	AIR
1.68100	381.610	1252.001	AIR
1.72400	385.270	1264.009	AIR
1.73700	386.330	1267.487	AIR
1.73800	388.010	1272.999	AIR
1.74000	388.620	1275.000	AIR
1.87000	398.980	1308.990	Paleo-Mag
1.87700	403.250	1322.999	AIR
1.92000	408.300	1339.567	Paleo-Mag

Appendix B

Core Sampling Field Notes and X-Ray Diffraction Results

APPENDIX B

FIELD NOTES and X-RAY DIFFRACTION RESULTS

Field Notes: Kerr-McGee core shed, Trona, California
October 28-30, 1986

ABBREVIATIONS USED

lt=light	dk=dark	lam=laminated
xtals=crystals	mass=massive	
CAL=calcite	ARAG=aragonite	DOL=dolomite
GAY=gaylussite	PIRS=pirsonnite	TRON=trona
THEN=thenardite	HAL=halite	NORT=northupite

X-Ray Diffraction results from runs at NMIMT
Samples scanned from 25-35 degrees 2-theta

M= major phase
I= intermediate phase
TR= trace phase

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
65.5	20.18	mud, dk green lam; GAY xtals	M-calcite	M-gaylussite
66.1	20.34	mud, alternating green/ white (ARAG?)finely lam;	M-calcite, TR-dolomite	
66.6	20.49	as 66.6	I-aragonite, TR-calcite TR-dolomite	
68.1	20.95	mud, green finely lam; GAY xtals, borax	M-calcite, TR-Gaylussite(?) TR-dolomite	
68.6	21.11	mud, green/white finely lam;	I-dolomite, TR-calcite TR-aragonité(?)	
69.1	21.26	mud, lt green/white lam;	I-dolomite, TR-calcite	
69.6	21.42	mud, green/white lam;	TR-calcite, TR-dolomite TR-aragonite(?)	
70.1	21.57	mud, green/white lam, lam more widely spaced than above;	I-calcite, TR-aragonite, TR-calcite	
70.6	21.72	as 70.1	TR-calcite, TR-aragonite, TR-dolomite	
71.0	21.85	as 70.1 ** possible depth error (.7 ft too deep last 7 samples)*****	I-aragonite, TR-calcite TR-dolomite	
71.6	22.03	mud, green/white lam, lam more frequent than 70.1-71.0;	I-aragonite, TR-calcite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
72.1	22.18	mud, white/green lam white lam thicker than above;	I-aragonite, TR-calcite	
72.6	22.34	mud, dk green/white lam, white lam less frequent than above (several mm apart);	I-aragonite, TR-calcite	
73.1	22.49	mud, yellow-green/ white lam, lam finer than 72.6;	I-aragonite, TR-calcite, TR-dolomite	
73.6	22.65	mud, green/white lam, lam as 72.6;	I-aragonite, TR-dolomite	
74.1	22.80	mud, lt green/white lam, white lam (~2 mm) parted from green;	I-aragonite, TR-calcite	
74.6	22.95	as 74.1; possible GAY xtals	I-aragonite, TR-dolomite, TR-calcite	
75.3	23.17	as 74.1;	I-aragonite, TR-calcite	
75.8	23.32	mud, lt grey-green, friable;GAY xtals	I-calcite, I-dolomite	
80.0	24.62	mud, lt grey, friable; GAY xtals **** HAND SAMPLE ****	I-calcite, I-dolomite	
81.0	24.92	mud, lt grey,dense;	I-calcite, TR-dolomite TR-pirssonite(?)	
81.6	25.11	mud, dk green,friable; large borax xtals, small (PIRS?) xtals	I-calcite, TR-dolomite TR-pirssonite(?)	
81.9	25.20	mud, dk green;borax xtals, large (PIRS?) xtals, sm trona (?)	I-calcite, TR-dolomite TR-pirssonite? TR-trona	M-pirssonite I-trona TR-dolomite
		********* BOTTOM OF PARTING MUD *********		
89.5	27.54	mud, dk green; GAY(?) xtals	M-calcite	
90.7	27.91	mud, dk green lam; GAY xtals	M-calcite	
91.5	28.15	mud,dk green; mainly HAL(?)xtals,few large borax xtals	TR-calcite, TR-aragonite(?)	
95.5	29.38	mud, lt green-yellow lam; GAY xtals	I-calcite	
96.6	29.72	mud, dk green; mainly GAY xtals	M-calcite	I-gaylussite I-burkeite

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
104.9-115.5		****PHOTOS****	trona and halite units	
111.7	34.37	mud, lt green finely lam; no salt xtals (above large trona seam)	I-calcite	
113.3	34.86	mud, lt green finely lam;(below trona seam)	TR-aragonite, TR-calcite	
114.5	35.23	mud, very lt green lam;	I-calcite, TR-aragonite	
115.9		****HAND SAMPLE****	Algal mats(?)	
116.4	35.82	mud, green, finely lam; some small borax xtals	M-calcite	
118.5	36.46	mud ,finely lam;	I-calcite, TR-aragonite	
120.0	36.92	mud, lt grey-green; small trona xtals	I-calcite	
		*********** TOP OF BOTTOM MUD **********************		
124.9	38.43	mud, lt green/orange lam;	M-calcite, TR-dolomite	
125.7	38.68	mud, green lam; possible borax	M-calcite	
126.3	38.86	mud, green finely lam; coarse GAY xtals	I-calcite, I-dolomite	
126.9	39.05	mud, buff, lam; GAY xtals	I-dolomite, TR-calcite	
127.7	39.29	mud, dk green, lam; GAY xtals,HAND SAMPLE	I-calcite, TR-dolomite	
128.7	39.60	mud, dk green; ~95% GAY xtals	TR-dolomite, TR-calcite	
128.9	39.66	mud, buff/green lam; NO macroscopic GAY xtals	I-dolomite, TR-calcite, TR-aragonite(?)	
129.8	39.94	mud, lt green/buff lam; small GAY xtals	M-calcite,	
130.1	40.03	mud, lt green/buff finely lam;(above THEN seam)	I-dolomite, TR-calcite	
130.4	40.12	mud, buff,crudely lam; (below THEN seam)	M-calcite, I-dolomite	I-gaylussite I-dolomite
131.5	40.46	mud, dk green/buff lam;	I-aragonite, TR-calcite TR-dolomite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
132.3	40.71	mud, dk green lam; ~80% GAY xtals	TR-dolomite, TR-calcite TR-aragonite	
134.4	41.35	mud, lt green, mass;	M-calcite, TR-dolomite	M-gaylussite TR-dolomite
136.4	41.97	mud, grey, mass; THEN(?) xtals (above THEN seam)	M-calcite, TR-dolomite	
137.5	42.31	mud, green-grey finely lam; (below THEN seam)	M-calcite, TR-dolomite	
139.0	42.77	mud, dk green/white lam;	I-aragonite, TR-calcite TR-dolomite	I-aragonite TR-calcite TR-dolomite
140.5	43.23	as 139.0, lam spaced further apart; coarse GAY xtals	M-dolomite, Tr-calcite	M-dolomite TR-calcite
141.8	43.63	mud, dk green/white lam, lam spaced further apart than 140.5;	I-aragonite, I-dolomite TR-calcite	I-dolomite I-aragonite TR-calcite
142.8	43.94	mud, dk green, finely lam;	I-dolomite TR-calcite TR-aragonite	

***** LOTS OF TELESCOPING IN CORE BOXES (150-200 ft); DEPTHS DUBIOUS! *****
(MEASUREMENTS RECORDED TO TENTH OF FOOT DOES NOT IMPLY CONFIDENCE AT THIS LEVEL!!!!!!!!)

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
151.1	46.49	mud, grey-green; mainly GAY xtals	I-dolomite, I-calcite	
153.0	47.08	as 151.1;	I-calcite	
154.1	47.42	as 151.1;	I-calcite	
158.6	48.80	mud, lt green, finely lam;	M-dolomite	
164.0	50.46	mud, lt green, mass;	M-dolomite	
170.0	52.31	mud, dk green; small salt(?) xtals	I-dolomite	
178.5	54.92	mud, green, mass;	TR-calcite, TR-dolomite	
183.0	56.31	mud, lt green, finely lam; few salt(?) xtals	I-calcite, I-dolomite	
187.4	57.66	mud, lt to dk green, finely lam; small salt(?) xtals	I-calcite, I-dolomite	
189.4	58.28	as 187.4; more salt(?) xtals	M-calcite, TR-dolomite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
191.9	59.05	mud, dk grey-green, lam; GAY xtals	M-calcite	
194.9	59.97	mud, green, lam;	I-dolomite	
199.9	61.51	as 194.9;GAY xtals	I-dolomite, TR-calcite	
205.0	63.08	mud, green,lam; GAY xtals	M-calcite, I-dolomite	
211.1	64.95	mud, green; GAY xtals	I-calcite	
218.0	67.08	mud, dk green-black; few salt(?) xtals	I-dolomite	
221.0	68.00	mud, dk green/buff lam, lam 2-10mm @; small salt(?) xtals	M-dolomite	M-dolomite
224.1	68.95	as 221.0;	I-dolomite	
226.6	69.72	mud, dk green-black, finely laminated, orange stained;	I-dolomite	
256.1	78.80	mud, green-buff; salt(?) xtals	TR-dolomite, TR-calcite	
257.0	79.08	mud, lt green, orange stained; salt(?) xtals	I-calcite, TR-dolomite	

*************** LIMITED CORE RECOVERY; DEPTH DUBIOUS!!!!!********

277.5	85.38	mud, lt grey/dk green crumbled mud; salt(?) xtals	TR-dolomite, TR-calcite	

*************** END OF LIMITED CORE RECOVERY ********************

288.1	88.65	mud, lt grey-green, mass; salt(?) xtals	TR-calcite	
298.1	91.72	as 288.1, less consolidated;	I-calcite	
308.0	94.77	mud, lt green, mass; small salt(?) xtals	I-calcite	
326.5	100.46	mud, lt green, orange-stained; ~ 50% salts	TR-calcite, TR-aragonite(?)	I-gaylussite TR-Northupite TR-aragonite?
349.9	107.66	as 326.5; more salt	I-northupite	I-northupite
359.3	110.55	as 326.5;	TR-calcite Unknown peaks 25-28 2-theta	
369.5	113.69	mud, lt green; small salt xtals	TR-aragonite TR-calcite? unknown peaks	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
370.5	114.00	mud, lt green, orange staining; small salt xtals	TR-calcite	
371.8	114.40	mud, lt green; trona	TR-calcite	
373.3	114.86	as 371.8;	TR-calcite	
375.0	115.38	mud, lt green;	I-calcite	
446.1	137.26	mud, green-black; few salt(?) xtals	TR-calcite TR-thenardite?	
447.5	137.69	mud, lt green, orange stained; few salt(?) xtals	TR-calcite	
448.5	138.00	as 447.5; less salt	TR-calcite TR-thenardite?	
449.4	138.28	as 447.5;	I-calcite	
453.0	139.38	mud, dk green; mainly salt	TR-dolomite	
455.5	140.15	as 453.0;	M-northupite TR-dolomite	
457.0	140.62	mud, dk green; trona,other salts(?)	I-northupite TR-calcite?	
459.5	141.38	mud, green; salts(?)	TR-calcite	
476.0	146.46	as 459.6;	I-calcite	
481.0	148.00	mud, green-grey; more salt than 459.6;	I-calcite,	
483.4	148.74	mud, lt green, orange-stained; large salt(?) xtals	I-calcite	
484.4	149.05	as 483.4; trona	I-calcite	
485.7	149.45	mud, green, orange-stained; large salt(?) xtals	I-calcite, TR-pirssonite	
486.0	149.54	as 485.7;	I-calcite	
487.8	150.09	mud, dk green; small salt(?) xtals	I-calcite	
489.0	150.46	mud, lt green; small salt(?) xtals	I-calcite	
490.0	150.77	mud, lt green, orange-stained; large salt(?) xtals	TR-northupite TR-dolomite TR-calcite(?)	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
491.2	151.14	mud, lt green, orange-stained; smaller salt(?) xtals than above	M-calcite	
492.0	151.38	as 491.2; more salt(?) than 491.2	M-calcite TR-pirssonite?	
493.2	151.75	as 491.2;	I-calcite	
493.9	151.97	mud, green; from thin bed of mud in between halite seams (@~6-8 cm)	I-calcite	
502.1	154.49	mud, green; large halite xtals	TR-calcite	
546.1	168.03	mud, dk green; PIRS xtals	I-calcite	
547.4	168.43	as 546.1;	M-calcite	
547.7	168.52	mud, dk green-black; less PIRS xtals than 546.1	I-calcite	
548.5	168.77	mud, green; many small salt(?) xtals	M-calcite	
549.5	169.08	as 548.5;	I-calcite	
550.5	169.38	mud, lt green; large PIRS(?) xtals	M-calcite, TR-dolomite	I-dolomite I-pirssonite
551.7	169.75	mud, green; PIRS(?) xtals	M-calcite	
552.6	170.03	mud, dk green, orange-stained; small PIRS? xtals	I-calcite	
553.8	170.40	as 552.6;	I-calcite Tr-pirssonite?	
555.0	170.77	mud, green, finely lam; little salt	M-calcite TR-dolomite	
556.0	171.08	mud, lt green; small salt? xtals	M-calcite TR-dolomite	
557.1	171.42	mud, lt green, orange-stained; small salt? xtals	I-calcite TR-dolomite	
558.2	171.75	mud, dk green-black; small salt? xtals	I-calcite TR-dolomite	
559.2	172.06	mud, green;white powdery substance, small salt?	TR-calcite TR-dolomite	
560.1	172.34	mud, green, finely lam; some salts?	I-calcite I-dolomite TR-thenardite TR-northupite?	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
561.0	172.62	as 560.1;	I-calcite, TR-dolomite	
562.2	172.98	mud, dk green-black;PIRS xtals	M-calcite, TR-dolomite	
563.2	173.29	mud, green;PIRS xtals	I-calcite, TR-dolomite	
564.3	173.63	mud, dk green, finely lam;salt? xtals	I-calcite, I-dolomite	
565.3	173.94	mud, green;mainly PIRS? xtals	M-calcite, I-dolomite	I-dolomite TR-pirssonite
567.0	174.46	as 565.3;	I-calcite, TR-dolomite	
568.7	174.98	mud, dk green; mainly PIRS? xtals	I-calcite, TR-dolomite	
569.9	175.35	mud, lt green, orange-stained; few salt? xtals	M-dolomite, TR-northupite?	
571.2	175.75	mud, lt green, orange-stained; large salt? xtals	I-dolomite, TR-calcite	
572.5	176.15	as 569.9;	I-dolomite	
573.8	176.55	mud, yellow-black, finely lam;	TR-dolomite	
574.9	176.89	as 573.8;	I-dolomite	
575.4	177.05	mud, lt green-yellow, finely lam; large salt? xtals	M-dolomite, I-calcite TR-northupite	
576.4	177.35	as 575.4;	TR-dolomite	
577.6	177.72	mud, green-grey, hard,dense; small salt? xtals	M-calcite	
578.6	178.03	as 577.6, some orange-staining	I-calcite, TR-dolomite	
579.5	178.31	mud, yellow-green, hard, dense, finely lam;few small salt? xtals	I-calcite, TR-dolomite	
580.7	178.68	as 579.5;	I-calcite, TR-dolomite	
582.2	179.14	mud, yellow-black, hard, dense; white stringers?, small salt? xtals	I-calcite, TR-dolomite?	
583.3	179.48	mud, green-grey, hard,dense, some orange-staining;	TR-thenardite TR-trona Tr-calcite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
584.2	179.75	mud, orange-black; some salt? xtals, trona?	TR-calcite, TR-dolomite?	
587.8	180.86	mud, dk green-black, orange-stained; large salt? xtals,(below 2ft seam of halite)	TR-calcite	
590.9	181.82	mud, yellow-dk green, lam; small salt? xtals	I-dolomite	
592.6	182.34	as 590.9; salt? xtals larger, more frequent	M-northupite, TR-dolomite, TR-calcite	
594.8	183.02	mud, lt green-yellow, salt? xtals	M-dolomite	
598.5	184.15	mud, lt green-yellow, hard, orange-stained, small salt? xtals	I-calcite, I-dolomite	I-dolomite TR-pirssonite
601.5	185.08	mud, orange-green; large salt? xtals	TR-thenardite	
603.0	185.54	as 601.5;(just below thin trona seam)	I-northupite, I-pirssonite TR-calcite, TR-dolomite	
605.0	186.15	mud, orange-green; ~50% salt xtals	I-pirssonite TR-calcite	
607.9	187.05	mud, green-grey; large salt? xtals	TR-northupite TR-calcite TR-dolomite?	
609.9	187.66	mud, orange-green; ~50% salt? xtals	M-pirssonite M-northupite TR-dolomite	
611.5	188.15	as 609.9; (0.7 ft above trona seam)	TR-thenardite TR-dolomite	
615.0	189.23	mud, orange-green; samll xtals	TR-northupite	
624.0	192.00	as 615.0; small PIRS? xtals	I-calcite	
631.8	194.40	mud, dk green-orange, hard, dense; small salt? xtals	TR-calcite	
634.2	195.14	mud, dk green-black, hard, dense, orange staining; small salt? xtals	I-calcite, TR-dolomite	
636.4	195.82	as 634.2; less salt?	TR-calcite, TR-dolomite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
638.5	196.46	mud, lt green, hard, dense, orange stained; salt? stringers	TR-northupite TR-calcite	
641.1	197.26	as 638.5;	TR-dolomite, TR-calcite	
642.5	197.69	mud, lt green, less dense than above; ~50% salts	M-calcite, TR-dolomite	
645.2	198.52	mud, green, crumbly; ~30% salts (below 1.5 ft halite seam)	M-calcite, TR-dolomite	
647.4	199.20	mud, dk green; ~20% salts	M-calcite, TR-pirssonite	
651.4	200.43	as 647.4;	M-calcite	
653.6	201.11	mud, grey; little salt	M-calcite I-dolomite	I-dolomite I-pirssonite
659.2	202.83	mud, green-grey; little salt	TR-calcite TR-dolomite	
661.7	203.60	as 659.2;	TR-calcite TR-dolomite	
664.8	204.55	mud, dk green; few salt? xtals	I-dolomite	
666.5	205.08	as 664.8;	TR-calcite TR-dolomite TR-northupite?	
668.4	205.66	mud, green-grey, orange-stained; few salts	I-calcite I-dolomite	
670.2	206.22	mud, dk green-orange; few salt stringers	TR-dolomite TR-northupite?	
671.5	206.62	mud, grey-black; small salt? xtals	TR-dolomite? TR-northupite?	
680.7	209.45	mud, yellow-green; little salt	M-dolomite	
684.7	210.68	mud, white, soft; no salt	M-dolomite	
690.3	212.40	mud, lt green-orange; PIRS? xtals	TR-dolomite TR-calcite	
701.2	215.75	mud, green-brown; ~60% salts?(just below coarse halite)	I-northupite I-dolomite	
706.2	217.29	mud, green; large salt? xtals	M-calcite TR-pirssonite	
710.2	218.52	as 706.2;	M-calcite TR-pirssonite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
713.4	219.51	as 706.2;	I-calcite TR-pirssonite	
717.0	220.62	mud, dk green; salt? xtals	I-calcite TR-pirssonite	
724.9	223.05	as 717.0;	I-calcite Tr-pirssonite	
727.5	223.85	mud, green, crumbly; small PIRS? xtals	I-calcite	
747.2	229.91	mud, lt grey- orange,lam w/ salts;(just above minor halite seam)	I-dolomite TR-calcite	
750.0	230.77	mud, lt green; little salt	I-dolomite	
752.5	231.54	mud, buff/black lam; salt? xtals	TR-calcite TR-northupite TR-dolomite	TR-pirssonite TR-northupite TR-dolomite
758.0	233.23	as 752.5;less salt	I-dolomite TR-calcite	
762.4	234.58	mud, grey/buff lam; little salt	M-dolomite	
766.5	235.85	mud, green/buff lam; more salt than 762.4	M-calcite TR-dolomite	I-pirssonite TR-dolomite
772.2	237.60	mud, buff/orange lam; little salt	TR-calcite TR-dolomite	
775.8	238.71	mud, green-orange,lam; minor salt	I-dolomite	
780.8	240.25	mud, green-dk orange; PIRS? xtals	M-calcite I-dolomite	
785.9	241.82	mud, green, mass;	I-dolomite	
794.2	244.37	mud, green-orange, mass;	I-dolomite	
798.7	245.75	as 780.8;	I-calcite I-dolomite	
805.0	247.69	mud, dk green-orange; minor salts	I-calcite TR-dolomite	
814.0	250.46	mud, lt green; large amounts of salts?	I-calcite TR-dolomite	
818.2	251.75	mud, green-orange, hard, dense; minor amounts of salt	M-calcite TR-dolomite	M-calcite I-pirssonite I-dolomite
823.6	253.42	as 818.2;	TR-calcite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
827.0	254.46	mud, dk green/orange lam, hard; minor salts	TR-calcite TR-dolomite	
831.7	255.91	mud, dk green/orange lam, hard, dense; little salt	TR-dolomite TR-calcite	
837.0	257.54	mud, green/buff lam; no salt ***HAND SAMPLE****	M-dolomite	M-dolomite
841.6	258.95	mud, green-orange; minor salt	I-dolomite TR-calcite	
846.6	260.49	mud, buff;mainly PIRS xtals	M-calcite TR-dolomite TR-pirsonnite	I-pirssonite I-dolomite TR-calcite
859.8	264.55	mud, green-orange, hard,dense; salt stringers	TR-calcite TR-dolomite	
864.0	265.85	mud, lt green-buff; minor salt	I-dolomite	
876.2	269.60	mud, lt green/buff lam; no salt	M-dolomite	
887.6	273.11	mud, green-orange, hard, dense; large amounts of salt	M-calcite TR-dolomite	
893.9	275.05	mud, dk green, hard, dense,orange-stained; minor salt	I-calcite	
898.9	276.58	as 893.9;	I-calcite	
903.1	277.88	mud, green-orange; large amount of salt	I-calcite I-dolomite	
908.6	279.57	mud, green-orange, hard, dense;minor salts	I-calcite TR-dolomite	
917.1	282.18	mud, grey-buff, hard, dense; salt stringers	M-calcite	
922.0	283.69	mud, grey-orange,hard dense; minor salt	M-calcite TR-dolomite	M-calcite TR-dolomite
930.0	286.15	mud, lt green-orange; minor salt	TR-calcite TR-dolomite	
935.7	287.91	as 930.0;	TR-calcite TR-dolomite	
939.9	289.20	mud, green/orange lam; hard, dense; salt xtals	TR-calcite TR-dolomite	
948.0	291.69	mud, green-orange, hard, dense; salt xtals	M-calcite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
955.0	293.85	mud, dk green, hard, dense;minor salt	I-calcite TR-dolomite	
959.0	295.08	mud, green, hard, dense; minor salt	M-calcite	
966.0	297.23	mud, green/buff lam; salt stringers	M-calcite	M-calcite
972.1	299.11	mud, dk green,mass;	TR-dolomite	
977.0	300.62	mud, green/buff lam;	I-calcite TR-dolomite	I-calcite TR-dolomite
986.0	303.38	mud, green,mass;	I-calcite	
1006.0	309.54	mud, green, mass; minor salt	I-aragonite? TR-calcite TR-dolomite?	
1012.0	311.38	mud, dk green,mass;	I-calcite	
1017.0	312.92	as 1012.0;	I-calcite I-dolomite	
1034.2	318.22	mud, green,mass; large HAL xtals	M-dolomite	
1068.0	328.62	mud, dk green, hard, dense; large HAL xtals	TR-dolomite TR-calcite	
1076.0	331.08	mud, green/orange lam; hard,dense; some salt	I-calcite	I-calcite
1080.4	332.43	mud, green, hard,dense; minor salt	TR-calcite	
1089.0	335.08	mud, green, lam;	I-calcite	
1093.0	336.31	mud, green-orange, hard, dense; large HAL xtals	M-calcite	
1099.0	338.15	mud, green, hard, dense;	TR-calcite	
1104.0	339.69	mud, grey/buff lam; minor salt	TR-calcite	
1106.5	340.46	mud, grey-buff, lam;minor salt	TR-calcite	
1110.6	341.72	mud, grey-black lam; little salt	TR-calcite	
1115.3	343.17	mud, grey-buff; **HAND SAMPLE**	TR-calcite	
1120.0	344.62	mud, grey lam; minor salt	I-calcite	
1125.5	346.31	mud, green-brown; no salt	TR-calcite	

CORE DEPTH (ft.)	CORE DEPTH (m)	FIELD DESCRIPTION	XRD LEACHED SAMPLES	XRD BULK SAMPLES
1130.5	347.85	mud, white/green lam, orange-stained; minor halite	M-calcite I-dolomite	M-calcite I-dolomite
1132.7	348.52	mud, green, lam; minor halite	M-dolomite TR-calcite	
1156.5	355.85	mud, grey/brown lam;minor salt	TR-calcite	
1210.7	372.52	mud, grey-buff, hard, dense; no salt	I-calcite	
1294.0	398.15	mud, grey-buff, hard, dense; **HAND SAMPLE**	TR-calcite	

Appendix C

Water-leaching Procedure

APPENDIX C

Water-Leaching Procedure

Materials needed:

- tweezers
- Mortar and Pestle
- 50 ml centrifuge tubes with caps
- distilled water
- centrifuge
- watch glasses
- oven
- 2-drams vials with lids

Procedure

1) Carefully inspect each sample for macroscopic salts crystals (gaylussite, pirssonite, trona, thenardite, borax, northupite, halite, etc.) Remove all macroscopic salts crystals with tweezers.

2) Grind remaining sample in mortar until it would pass 100 mesh.

3) Place approximately five grams of ground mud sample into a 50 ml centrifuge tube. Fill tube with distilled water and cap (no air pockets should be in the tube).

4) Shake centrifuge tube. Leave sample for 24 hours, shaking periodically, to dissolve soluble salts.

5) After 24 hours, check sample. If the sample is cloudy, centrifuge sample until liquid is clear. Carefully decant liquid. Re-add fresh distilled water , cap, and shake sample to rinse mud. Centrifuge sample until liquid is clear and, again, carefully decant liquid. Repeat the rinsing process at least three times or until liquid shows no signs of color. (the 'leachate' was usually a dark yellow to brown color).

6) Carefully scrape the centrifuged mud (not easy) from the bottom of the centrifuge tube. Place mud on a watch glass. [NOTE: At this point the sample was checked to make sure no tiny salt crystals were visible. If there were crystals visible (O.K., it happened..), the entire leaching process for that sample was repeated.] Place in the oven overnight at 35°C to dry the mud. Do not raise temperature beyond this!

7) Remove sample from oven. [NOTE: again, another check for salt was made. If an efflourescence was observed, the entire leaching process was repeated.] Remove dried sample from watch glass and crush in mortar until the sample will pass 200 mesh. Place sample in vial for mass spectrometric analysis.

Appendix D

EDTA Dissolution Procedure

Appendix D

EDTA Dissolution Procedure and Results

Introduction

Following Glover (1961) and Bodine and others (1973), an attempt was made to separate calcite from dolomite using a high pH aqueous solution of the complexing agent EDTA (ethylenedinitrilotetraacetic acid). The chemical processes governing the dissolution of alkaline earth carbonates by EDTA can be found in Bodine and Fernalld (1973; after Welcher, 1958).

Materials Needed

Tetrasodium EDTA dihydrate powder
50 Ml vials (centrifuge tubes used here) with caps
Ph paper or meter
shaker
watch glasses
centrifuge
distilled water
oven
containers for leached sample (2-dram vials used here)

Procedure Followed

1) A .25M (near saturation) solution of EDTA (using tetrasodium EDTA pH should be 10.5) was prepared.

2) Searles Lake sample 127.7 (a fairly equal mixture of "mongrel calcite and dolomite, and about 30 weight percent carbonate) was selected as a test sample. Four aliquots of water-leached mud sample (passing 100 mesh) at two grams each were placed into 50 ml centrifuge tubes. 50 ml of EDTA solution were added to the sample, and the centrifuge tube was capped.

3) The four samples were placed on a shaker for 1,2,4, and 8 hours respectively at room temperature (20°C). High temperature used by previous investigators were not used due to the extreme rapidity of the complexing reaction.

4) After each time interval had passed, one sample was removed from the shaker. The pH was measured (all samples had a pH between 9.5-10.5).

5) The sample was placed in a centrifuge briefly to spin down solids. The EDTA solution was carefully decanted. The sample was rinsed with distilled water several times to remove all EDTA. The mud was carefully removed from the centrifuge tube and placed on a watch glass. The watch glass was placed in a 35°C oven overnight to dry.

6) Mortar and pestle were used to crush the dried sample until it passed 200 mesh. Samples were placed in vials for XRD analysis.

7) The steps one through six were repeated until eight samples were created (A and B): two 1-hour, two 2-hour, two 4-hour, and two 8-hour samples.

8) X-Ray diffraction analysis was performed on the eight samples. The samples were scanned from 25-32 degrees 2- Cu K_x for the primary carbonate peaks. The calcite peak gradually disappeared, and the primary dolomite peak diminished slightly by 4 hours and was significantly reduced at 8 hours.

9) The eight samples were analyzed for ^{18}O and ^{13}C content at the University of New Mexico (Albuquerque) isotope lab following the methods of McCrea (1950). The UNM lab reported difficulties in the mass spectrometric analyses of these samples: samples 1A, 4A, and 8B were reported "contaminated". The stable isotope content of the remaining samples is shown graphically in Figure 1D. No acid fraction factors were assigned to the samples. The stable isotope content of the bulk water-leached sample and the dolomite (determined later) values are included in Figure 1D for illustrative purposes.

Discussion

The isotopic content of the 1-hour sample is very similar to that of the bulk sample, as would be expected. The isotopic content of the 2-hour sample approaches the dolomite values. however, it appears that the EDTA had begun to complex with the dolomite by the 4-hour sample, and by eight hours the isotopic content of the sample had essentially returned to that of the 1-hour sample. It appears that the same relative proportion of calcite/dolomite existed after eight hours of EDTA dissolution as had existed in the 1-hour sample.

The money available for the stable isotope analysis did not permit the detailed mole percent calite and dolomite determination nor exhaustive grain size analysis that would have been required to calibrate this procedure. Further, the 'dolomite endpoint' might not be achieved as the EDTA began to complex with the dolomite in the sample.

TETRASODIUM EDTA MINERAL SEPARATIONS

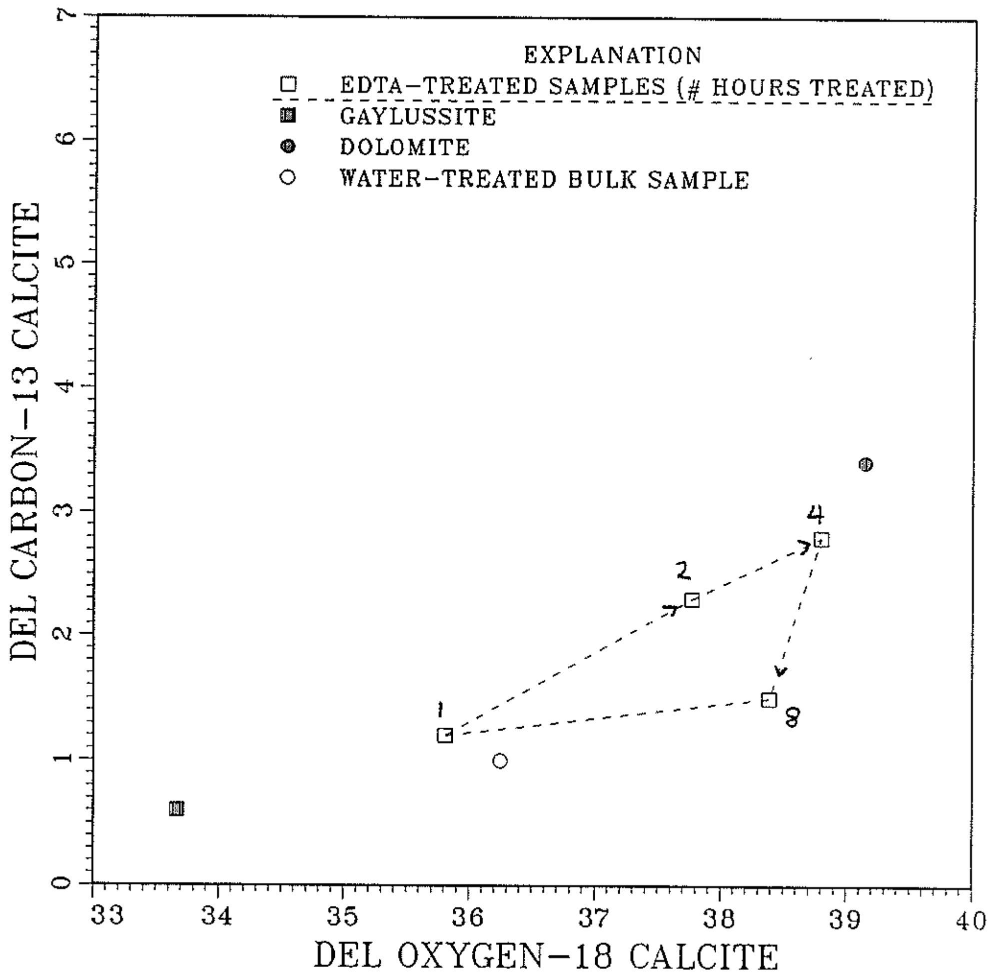


FIGURE 1D. $\delta^{13}C$ and $\delta^{18}O$ values for EDTA treated samples

Appendix E

Hierarchical Statistics Analysis of Uncertainty for ^{18}O and ^{13}C Analyses

```
c
c       PROGRAM RUNS ANALYSIS OF VARIANCE FOR DATA SET
c
        program anova
        character fname*80
        character fname2*20
        character desc*50
        dimension yhia(1000),yhiabar(1000)
        dimension yhi(1000),yhibar(1000),nhi(1000)
        dimension yh(1000),yhbar(1000),nh(1000)
        dimension min(1000),maj(1000)
        DIMENSION EHI(1000),THI(1000),RMHI(1000)

        write(6,*)'enter the input file name'
        read(5,'(a)')fname
        open(unit=99,file=fname,status='old')
        write(6,*)'enter output file name'
        read(5,'(a)')fname2
        open(unit=96,file=fname2,status='new')


        read(99,*)n,nmhi,nmh
        if(n.gt.1000 .or.nmhi.gt.1000 .or. nmh.gt.1000)then
           write(6,*)'program dimensioned to 1000',n,nmhi,nmh
           stop
        endif
        write(6,*)'input number of cycles per minor group'
        read(5,*)k

        do 100 i = 1,n
           read(99,*)yhia(i),min(i),maj(i)
           nhi(min(i)) = nhi(min(i)) + 1
           nh(maj(i)) = nh(maj(i)) + 1
100     continue

        write(96,*)
        write(96,*)
        write(96,*)
        write(96,*)
        write(96,*)
        write(96,*)
        write(6,*)'enter decription'
        read(5,'(a)')desc
        write(96,*)desc
        write(96,*)'________________________'
        write(96,*)
        write(96,*)'number of points:      ',n
        write(96,*)'number of minor groups:',nmhi
        write(96,*)'number of major groups:',nmh
        write(96,*)

C   data has been input, compute the totals for each minor and major group

        do 200 i = 1,n
           yhi(min(i)) = yhi(min(i)) + yhia(i)
           yh(maj(i)) = yh(maj(i)) + yhia(i)
           y = y + yhia(i)
200     continue
        DO 300 I = 1,NMHI*nmh
           YHIBAR(I) = YHI(I) / FLOAT(NHI(I) )
           WRITE(96,10)' MINOR GROUP #, SIZE, TOTAL, MEAN',
     1                 I,NHI(I),YHI(I),YHIBAR(I)
300     CONTINUE
        WRITE(96,*)
        DO 400 I = 1,NMH
           YHBAR(I) = YH(I) / FLOAT(NH(I) )
        WRITE(96,10)' MAJOR GROUP #, SIZE, TOTAL, MEAN',
     1                 I,NH(I),YH(I),YHBAR(I)
```

```
400     CONTINUE
        YBAR = Y/FLOAT(N)
        write(96,*)
        WRITE(96,10)' TOTAL RUN      SIZE, TOTAL, MEAN',1,N,Y,YBAR
10      FORMAT(1X,A,I5,1X,I5,1X,1PE12.4,1X,1PE12.4)
        WRITE(96,*)
        WRITE(96,*)

C     OK SO THE YH... AND NH ARRAYS HAVE BEEN FILLED
C     SO NOW COMPUTE THE A,B1,B2,C STUFF:

        A = 0.
        DO 500 I = 1,N
           A = A + YHIA(I)*YHIA(I)
500     CONTINUE
        B1 = 0.
        DO 600 I = 1,(NMHI*nmh)
           B1 = B1 + YHI(I)*YHI(I)/float(k)
600     CONTINUE
        B2 = 0.
        DO 700 I = 1,NMH
           B2 = B2 + YH(I)*YH(I)/(float(k)*float(nmhi))
700     CONTINUE
        C = Y*Y/FLOAT(N)

C      NOW WE HAVE THE A,B1,B2,C SO COMPUTE THE SSQ STUFF

        SSQW = A-B1
        SSQB = B1 - B2
        SSQH = B2 - C
        SSQT = A - C

C        IN ALLAN'S NOTATION I = NMH
C                            J = NMHI
C                            K = K
C       COMPUTE DEGREES OF FREEDOM

        IHDF=NMH-1
        IBDF=NMH*(NMHI-1)
        IWDF=NMH*NMHI*(K-1)

C      COMPUTE THE MEAN SUM SQUARES (DIVIDE SUM SQUARES BY DEGREES OF FREEDOM)
C
        ESSQW= SSQW/FLOAT(IWDF)
        ESSQB = SSQB/FLOAT(IBDF)
        ESSQH = SSQH/FLOAT(IHDF)

        WRITE(96,*)
        write(96,*)
        WRITE(96,*)'                    ____________________________________'
        write(96,*)'                                 ANOVA                  '
        WRITE(96,*)'                    ____________________________________'
        write(96,*)

C       COMPUTE THE VARIANCE OF MEAN SUM SQUARES FOR EXPECTATION CALCULATIONS

        SIGSQW=ESSQW
        SIGSQB=(ESSQB-ESSQW)/FLOAT(K)
        SIGSQH=(ESSQH-ESSQB)/(FLOAT(K*NMHI))
        WRITE(96,30)'SUM SQUARES','DEG FRDM','MEAN SUM SQUARES','VARIANCE'
30      FORMAT(22x,A11,4x,A8,4x,A16,2X,A8)
40      FORMAT(1x,A19,4x,f8.4,8x,I4,10x,f8.4,7x,f8.4)
        WRITE(96,*)
        WRITE(96,40)'BTWN SAMPLES       ',SSQH,IHDF,ESSQH,SIGSQH
        WRITE(96,*)
        WRITE(96,40)'BTWN GAS ALIQUOTS  ',SSQB,IBDF,ESSQB,SIGSQB
        WRITE(96,*)
        WRITE(96,40)'BTWN MACHINE CYCLES',SSQW,IWDF,ESSQW,SIGSQW

        END
```

SAMPLE 690.3 OXYGEN-18

number of points: 150
number of minor groups: 5
number of major groups: 5

MINOR GROUP #, SIZE, TOTAL, MEAN	1	6	1.4449E+02	2.4082E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	2	6	1.4366E+02	2.3943E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	3	6	1.4320E+02	2.3866E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	4	6	1.4308E+02	2.3846E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	5	6	1.4321E+02	2.3868E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	6	6	1.4750E+02	2.4583E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	7	6	1.4807E+02	2.4679E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	8	6	1.4814E+02	2.4690E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	9	6	1.4813E+02	2.4689E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	10	6	1.4813E+02	2.4688E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	11	6	1.5090E+02	2.5150E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	12	6	1.4941E+02	2.4901E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	13	6	1.4892E+02	2.4820E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	14	6	1.4892E+02	2.4820E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	15	6	1.4907E+02	2.4845E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	16	6	1.4845E+02	2.4742E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	17	6	1.4764E+02	2.4607E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	18	6	1.4765E+02	2.4608E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	19	6	1.4739E+02	2.4565E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	20	6	1.4747E+02	2.4578E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	21	6	1.3855E+02	2.3091E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	22	6	1.3964E+02	2.3274E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	23	6	1.3950E+02	2.3249E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	24	6	1.3935E+02	2.3225E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	25	6	1.3914E+02	2.3191E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	1	30	7.1763E+02	2.3921E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	2	30	7.3997E+02	2.4666E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	3	30	7.4721E+02	2.4907E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	4	30	7.3859E+02	2.4620E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	5	30	6.9618E+02	2.3206E+01
TOTAL RUN SIZE, TOTAL, MEAN	1	150	3.6396E+03	2.4264E+01

ANOVA

	SUM SQUARES	DEG FRDM	MEAN SUM SQUARES	VARIANCE
BTWN SAMPLES	58.1719	4	14.5430	0.4831
BTWN GAS ALIQUOTS	1.0000	20	0.0500	0.0072
BTWN MACHINE CYCLES	0.8359	125	0.0067	0.0067

RAW OXYGEN_18 CYCLE DATA: SAMPLE 690.3

CYCLE	MINOR GROUP	MAJOR GROUP
24.087	1	1
24.069	1	1
23.995	1	1
24.125	1	1
24.216	1	1
24.001	1	1
23.652	2	1
24.025	2	1
24.019	2	1
23.910	2	1
23.974	2	1
24.076	2	1
23.867	3	1
23.742	3	1
23.907	3	1
23.824	3	1
23.909	3	1
23.949	3	1
23.815	4	1
23.842	4	1
23.769	4	1
23.779	4	1
24.010	4	1
23.863	4	1
23.839	5	1
23.857	5	1
23.955	5	1
23.968	5	1
23.856	5	1
23.730	5	1
24.461	6	2
24.547	6	2
24.769	6	2
24.512	6	2
24.532	6	2
24.676	6	2
24.563	7	2
24.701	7	2
24.676	7	2
24.758	7	2
24.665	7	2
24.710	7	2
24.686	8	2
24.722	8	2
24.725	8	2
24.695	8	2
24.650	8	2
24.659	8	2
24.684	9	2
24.699	9	2
24.692	9	2
24.754	9	2
24.695	9	2
24.608	9	2
24.697	10	2
24.625	10	2
24.704	10	2
24.635	10	2
24.728	10	2
24.738	10	2
25.194	11	3
25.190	11	3
25.063	11	3
25.060	11	3
25.209	11	3
25.184	11	3

CYCLE	MINOR GROUP	MAJOR GROUP
24.848	12	3
24.823	12	3
24.866	12	3
24.977	12	3
24.947	12	3
24.946	12	3
24.728	13	3
24.911	13	3
24.786	13	3
24.866	13	3
24.771	13	3
24.857	13	3
24.841	14	3
24.838	14	3
24.775	14	3
24.767	14	3
24.869	14	3
24.830	14	3
24.867	15	3
24.865	15	3
24.846	15	3
24.760	15	3
24.795	15	3
24.935	15	3
24.800	16	4
24.779	16	4
24.771	16	4
24.690	16	4
24.629	16	4
24.780	16	4
24.610	17	4
24.574	17	4
24.642	17	4
24.615	17	4
24.733	17	4
24.465	17	4
24.561	18	4
24.618	18	4
24.654	18	4
24.603	18	4
24.512	18	4
24.699	18	4
24.520	19	4
24.518	19	4
24.688	19	4
24.563	19	4
24.493	19	4
24.609	19	4
24.573	20	4
24.576	20	4
24.659	20	4
24.604	20	4
24.577	20	4
24.476	20	4
22.855	21	5
22.988	21	5
23.179	21	5
23.160	21	5
23.159	21	5
23.206	21	5
23.287	22	5
23.102	22	5
23.209	22	5
23.331	22	5
23.288	22	5
23.425	22	5

CYCLE	MINOR GROUP	MAJOR GROUP
23.287	23	5
23.383	23	5
23.291	23	5
22.991	23	5
23.284	23	5
23.259	23	5
23.241	24	5
23.220	24	5
23.203	24	5
23.167	24	5
23.303	24	5
23.218	24	5
23.223	25	5
23.214	25	5
23.173	25	5
23.109	25	5
23.242	25	5
23.182	25	5

SAMPLE 690.3 CARBON-13

number of points:	150
number of minor groups:	5
number of major groups:	5

MINOR GROUP #, SIZE, TOTAL, MEAN	1	6	2.6780E+02	4.4633E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	2	6	2.6750E+02	4.4583E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	3	6	2.6751E+02	4.4585E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	4	6	2.6750E+02	4.4583E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	5	6	2.6793E+02	4.4655E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	6	6	2.6807E+02	4.4679E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	7	6	2.6854E+02	4.4756E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	8	6	2.6837E+02	4.4729E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	9	6	2.6890E+02	4.4817E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	10	6	2.6856E+02	4.4759E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	11	6	2.6943E+02	4.4905E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	12	6	2.6906E+02	4.4844E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	13	6	2.6917E+02	4.4862E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	14	6	2.6887E+02	4.4812E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	15	6	2.6898E+02	4.4830E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	16	6	2.6845E+02	4.4742E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	17	6	2.6838E+02	4.4730E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	18	6	2.6854E+02	4.4757E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	19	6	2.6857E+02	4.4762E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	20	6	2.6851E+02	4.4752E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	21	6	2.6615E+02	4.4358E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	22	6	2.6691E+02	4.4485E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	23	6	2.6672E+02	4.4453E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	24	6	2.6689E+02	4.4482E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	25	6	2.6669E+02	4.4448E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	1	30	1.3382E+03	4.4608E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	2	30	1.3424E+03	4.4748E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	3	30	1.3455E+03	4.4851E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	4	30	1.3425E+03	4.4749E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	5	30	1.3334E+03	4.4445E+01
TOTAL RUN SIZE, TOTAL, MEAN	1	150	6.7020E+03	4.4680E+01

ANOVA

	SUM SQUARES	DEG FRDM	MEAN SUM SQUARES	VARIANCE
BTWN SAMPLES	3.0000	4	0.7500	0.0247
BTWN GAS ALIQUOTS	0.1563	20	0.0078	0.0008
BTWN MACHINE CYCLES	0.4063	125	0.0033	0.0033

RAW CARBON-13 CYCLE DATA: SAMPLE 690.3

CYCLE	MINOR GROUP	MAJOR GROUP
44.667	1	1
44.531	1	1
44.551	1	1
44.678	1	1
44.690	1	1
44.684	1	1
44.542	2	1
44.575	2	1
44.519	2	1
44.506	2	1
44.690	2	1
44.666	2	1
44.578	3	1
44.536	3	1
44.498	3	1
44.608	3	1
44.712	3	1
44.578	3	1
44.639	4	1
44.557	4	1
44.617	4	1
44.478	4	1
44.635	4	1
44.572	4	1
44.625	5	1
44.608	5	1
44.643	5	1
44.702	5	1
44.646	5	1
44.706	5	1
44.654	6	2
44.666	6	2
44.726	6	2
44.671	6	2
44.638	6	2
44.716	6	2
44.704	7	2
44.765	7	2
44.747	7	2
44.749	7	2
44.755	7	2
44.815	7	2
44.733	8	2
44.729	8	2
44.708	8	2
44.776	8	2
44.705	8	2
44.721	8	2
44.800	9	2
44.816	9	2
44.844	9	2
44.846	9	2
44.789	9	2
44.804	9	2
44.744	10	2
44.712	10	2
44.672	10	2
44.808	10	2
44.816	10	2
44.803	10	2
44.937	11	3
44.938	11	3
44.908	11	3
44.911	11	3
44.881	11	3
44.855	11	3

CYCLE	MINOR GROUP	MAJOR GROUP
44.791	12	3
44.765	12	3
44.849	12	3
44.914	12	3
44.949	12	3
44.794	12	3
44.838	13	3
44.922	13	3
44.812	13	3
44.896	13	3
44.849	13	3
44.853	13	3
44.825	14	3
44.808	14	3
44.846	14	3
44.846	14	3
44.783	14	3
44.766	14	3
44.860	15	3
44.892	15	3
44.792	15	3
44.830	15	3
44.792	15	3
44.816	15	3
44.690	16	4
44.790	16	4
44.757	16	4
44.711	16	4
44.768	16	4
44.737	16	4
44.739	17	4
44.687	17	4
44.711	17	4
44.708	17	4
44.789	17	4
44.745	17	4
44.739	18	4
44.695	18	4
44.761	18	4
44.790	18	4
44.772	18	4
44.783	18	4
44.702	19	4
44.735	19	4
44.829	19	4
44.809	19	4
44.740	19	4
44.758	19	4
44.775	20	4
44.805	20	4
44.710	20	4
44.720	20	4
44.789	20	4
44.711	20	4
44.227	21	5
44.353	21	5
44.374	21	5
44.361	21	5
44.444	21	5
44.390	21	5
44.401	22	5
44.518	22	5
44.538	22	5
44.441	22	5
44.521	22	5
44.493	22	5

CYCLE	MINOR GROUP	MAJOR GROUP
44.353	23	5
44.492	23	5
44.535	23	5
44.449	23	5
44.483	23	5
44.405	23	5
44.432	24	5
44.420	24	5
44.549	24	5
44.605	24	5
44.459	24	5
44.429	24	5
44.426	25	5
44.522	25	5
44.398	25	5
44.374	25	5
44.485	25	5
44.483	25	5

SAMPLE 375.0 OXYGEN-18

number of points: 150
number of minor groups: 5
number of major groups: 5

MINOR GROUP #, SIZE, TOTAL, MEAN	1	6	1.0872E+02	1.8120E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	2	6	1.0946E+02	1.8243E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	3	6	1.0917E+02	1.8194E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	4	6	1.0845E+02	1.8074E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	5	6	1.0858E+02	1.8097E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	6	6	1.1005E+02	1.8341E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	7	6	1.0977E+02	1.8295E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	8	6	1.0965E+02	1.8275E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	9	6	1.0930E+02	1.8217E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	10	6	1.1006E+02	1.8344E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	11	6	1.1317E+02	1.8861E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	12	6	1.1392E+02	1.8987E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	13	6	1.1268E+02	1.8780E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	14	6	1.1264E+02	1.8774E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	15	6	1.1347E+02	1.8912E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	16	6	1.0823E+02	1.8039E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	17	6	1.0870E+02	1.8117E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	18	6	1.0996E+02	1.8327E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	19	6	1.0874E+02	1.8124E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	20	6	1.0990E+02	1.8316E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	21	6	1.1293E+02	1.8821E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	22	6	1.1407E+02	1.9011E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	23	6	1.1332E+02	1.8887E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	24	6	1.1437E+02	1.9062E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	25	6	1.1418E+02	1.9031E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	1	30	5.4437E+02	1.8146E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	2	30	5.4883E+02	1.8294E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	3	30	5.6589E+02	1.8863E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	4	30	5.4554E+02	1.8185E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	5	30	5.6887E+02	1.8962E+01
TOTAL RUN SIZE, TOTAL, MEAN	1	150	2.7735E+03	1.8490E+01

ANOVA

	SUM SQUARES	DEG FRDM	MEAN SUM SQUARES	VARIANCE
BTWN SAMPLES	18.3867	4	4.5967	0.1515
BTWN GAS ALIQUOTS	1.0430	20	0.0521	0.0030
BTWN MACHINE CYCLES	4.2969	125	0.0344	0.0344

RAW OXYGEN-18 CYCLE DATA: SAMPLE 375.0

CYCLE	MINOR GROUP	MAJOR GROUP
18.209	1	1
18.111	1	1
18.073	1	1
18.211	1	1
18.033	1	1
18.085	1	1
17.883	2	1
18.127	2	1
18.149	2	1
19.185	2	1
18.015	2	1
18.100	2	1
17.917	3	1
18.198	3	1
18.361	3	1
18.364	3	1
18.231	3	1
18.094	3	1
17.969	4	1
17.848	4	1
18.147	4	1
18.297	4	1
18.202	4	1
17.983	4	1
18.134	5	1
18.328	5	1
18.216	5	1
17.804	5	1
18.156	5	1
17.942	5	1
18.555	6	2
18.128	6	2
18.443	6	2
18.179	6	2
18.317	6	2
18.424	6	2
18.089	7	2
18.356	7	2
18.332	7	2
18.389	7	2
18.341	7	2
18.262	7	2
18.104	8	2
18.276	8	2
18.301	8	2
18.424	8	2
18.352	8	2
18.194	8	2
17.988	9	2
18.349	9	2
18.258	9	2
18.325	9	2
18.330	9	2
18.049	9	2
18.317	10	2
18.254	10	2
18.205	10	2
18.444	10	2
18.307	10	2
18.536	10	2
18.463	11	3
18.849	11	3
18.718	11	3
19.181	11	3
18.929	11	3
19.026	11	3

CYCLE	MINOR GROUP	MAJOR GROUP
18.859	12	3
18.759	12	3
19.057	12	3
19.130	12	3
18.959	12	3
19.159	12	3
18.811	13	3
18.727	13	3
18.985	13	3
18.954	13	3
18.555	13	3
18.649	13	3
18.844	14	3
18.819	14	3
18.744	14	3
18.750	14	3
18.768	14	3
18.718	14	3
18.952	15	3
18.922	15	3
18.835	15	3
18.866	15	3
18.907	15	3
18.991	15	3
17.900	16	4
17.868	16	4
18.235	16	4
18.126	16	4
17.883	16	4
18.221	16	4
17.962	17	4
17.952	17	4
18.165	17	4
18.382	17	4
18.139	17	4
18.101	17	4
18.069	18	4
18.226	18	4
18.636	18	4
18.429	18	4
18.414	18	4
18.190	18	4
17.966	19	4
18.070	19	4
18.440	19	4
18.014	19	4
18.015	19	4
18.238	19	4
18.176	20	4
18.083	20	4
18.373	20	4
18.603	20	4
18.512	20	4
18.151	20	4
18.683	21	5
18.796	21	5
19.017	21	5
18.727	21	5
18.751	21	5
18.952	21	5
18.895	22	5
19.014	22	5
19.148	22	5
19.189	22	5
18.808	22	5
19.015	22	5

CYCLE	MINOR GROUP	MAJOR GROUP
18.562	23	5
18.877	23	5
18.914	23	5
18.827	23	5
18.913	23	5
19.226	23	5
18.876	24	5
19.187	24	5
19.057	24	5
18.921	24	5
19.170	24	5
19.158	24	5
19.330	25	5
19.061	25	5
18.906	25	5
19.231	25	5
18.746	25	5
18.910	25	5

SAMPLE 375.0 CARBON-13

number of points: 150
number of minor groups: 5
number of major groups: 5

MINOR GROUP #, SIZE, TOTAL, MEAN	1	6	2.6720E+02	4.4533E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	2	6	2.6723E+02	4.4538E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	3	6	2.6732E+02	4.4553E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	4	6	2.6719E+02	4.4532E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	5	6	2.6734E+02	4.4557E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	6	6	2.6840E+02	4.4734E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	7	6	2.6818E+02	4.4696E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	8	6	2.6804E+02	4.4674E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	9	6	2.6830E+02	4.4717E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	10	6	2.6826E+02	4.4709E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	11	6	2.6906E+02	4.4844E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	12	6	2.6979E+02	4.4964E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	13	6	2.6929E+02	4.4882E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	14	6	2.6945E+02	4.4908E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	15	6	2.6958E+02	4.4930E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	16	6	2.6799E+02	4.4665E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	17	6	2.6917E+02	4.4862E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	18	6	2.6844E+02	4.4740E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	19	6	2.6862E+02	4.4770E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	20	6	2.6903E+02	4.4839E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	21	6	2.6998E+02	4.4997E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	22	6	2.6968E+02	4.4947E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	23	6	2.7016E+02	4.5027E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	24	6	2.7000E+02	4.5000E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	25	6	2.7042E+02	4.5070E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	1	30	1.3363E+03	4.4543E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	2	30	1.3412E+03	4.4706E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	3	30	1.3472E+03	4.4906E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	4	30	1.3433E+03	4.4775E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	5	30	1.3502E+03	4.5008E+01
TOTAL RUN SIZE, TOTAL, MEAN	1	150	6.7181E+03	4.4787E+01

ANOVA

	SUM SQUARES	DEG FRDM	MEAN SUM SQUARES	VARIANCE
BTWN SAMPLES	3.7813	4	0.9453	0.0312
BTWN GAS ALIQUOTS	0.1875	20	0.0094	-0.0011
BTWN MACHINE CYCLES	2.0313	125	0.0162	0.0162

RAW CARBON-13 CYCLE DATA: SAMPLE 375.0

CYCLE	MINOR GROUP	MAJOR GROUP
44.462	1	1
44.607	1	1
44.542	1	1
44.465	1	1
44.504	1	1
44.621	1	1
44.449	2	1
44.655	2	1
44.417	2	1
44.583	2	1
44.497	2	1
44.627	2	1
44.526	3	1
44.588	3	1
44.567	3	1
44.580	3	1
44.541	3	1
44.515	3	1
44.603	4	1
44.435	4	1
44.448	4	1
44.566	4	1
44.627	4	1
44.513	4	1
44.487	5	1
44.603	5	1
44.563	5	1
44.421	5	1
44.606	5	1
44.660	5	1
44.904	6	2
44.659	6	2
44.876	6	2
44.588	6	2
44.651	6	2
44.727	6	2
44.651	7	2
44.591	7	2
44.745	7	2
44.714	7	2
44.661	7	2
44.814	7	2
44.679	8	2
44.573	8	2
44.698	8	2
44.651	8	2
44.769	8	2
44.674	8	2
44.572	9	2
44.651	9	2
44.776	9	2
44.676	9	2
44.800	9	2
44.825	9	2
44.566	10	2
44.638	10	2
44.647	10	2
44.740	10	2
44.781	10	2
44.884	10	2
44.955	11	3
44.777	11	3
44.603	11	3
44.975	11	3
44.854	11	3
44.897	11	3

CYCLE	MINOR GROUP	MAJOR GROUP
45.105	12	3
44.927	12	3
44.831	12	3
44.964	12	3
44.905	12	3
45.053	12	3
44.766	13	3
44.851	13	3
44.955	13	3
44.972	13	3
44.959	13	3
44.787	13	3
44.857	14	3
45.029	14	3
45.007	14	3
44.840	14	3
44.774	14	3
44.941	14	3
44.990	15	3
44.949	15	3
44.821	15	3
45.025	15	3
44.994	15	3
44.802	15	3
44.486	16	4
44.525	16	4
44.570	16	4
44.672	16	4
45.005	16	4
44.733	16	4
44.839	17	4
44.887	17	4
44.920	17	4
45.018	17	4
44.802	17	4
44.704	17	4
44.806	18	4
44.802	18	4
44.997	18	4
44.784	18	4
44.612	18	4
44.440	18	4
44.599	19	4
44.783	19	4
44.747	19	4
45.033	19	4
44.826	19	4
44.633	19	4
44.936	20	4
44.783	20	4
44.793	20	4
44.873	20	4
44.949	20	4
44.698	20	4
44.814	21	5
45.220	21	5
44.953	21	5
44.790	21	5
45.008	21	5
45.194	21	5
45.018	22	5
45.041	22	5
44.753	22	5
44.870	22	5
44.927	22	5
45.072	22	5

CYCLE	MINOR GROUP	MAJOR GROUP
44.865	23	5
44.779	23	5
45.200	23	5
45.077	23	5
45.088	23	5
45.155	23	5
44.964	24	5
45.082	24	5
45.037	24	5
44.994	24	5
44.978	24	5
44.946	24	5
45.060	25	5
44.969	25	5
44.875	25	5
45.485	25	5
45.243	25	5
44.786	25	5

SAMPLE 127.7- OXYGEN-18

number of points: 150
number of minor groups: 5
number of major groups: 5

MINOR GROUP #, SIZE, TOTAL, MEAN	1	6	1.7868E+02	2.9780E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	2	6	1.7832E+02	2.9719E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	3	6	1.7861E+02	2.9768E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	4	6	1.7869E+02	2.9782E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	5	6	1.7874E+02	2.9791E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	6	6	1.7879E+02	2.9798E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	7	6	1.7864E+02	2.9773E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	8	6	1.7855E+02	2.9758E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	9	6	1.7844E+02	2.9740E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	10	6	1.7843E+02	2.9738E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	11	6	1.8324E+02	3.0540E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	12	6	1.8350E+02	3.0583E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	13	6	1.8339E+02	3.0565E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	14	6	1.8310E+02	3.0517E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	15	6	1.8324E+02	3.0539E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	16	6	1.7972E+02	2.9954E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	17	6	1.8004E+02	3.0006E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	18	6	1.8010E+02	3.0017E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	19	6	1.7962E+02	2.9937E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	20	6	1.8000E+02	3.0000E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	21	6	1.7953E+02	2.9922E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	22	6	1.7956E+02	2.9926E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	23	6	1.7898E+02	2.9830E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	24	6	1.7892E+02	2.9819E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	25	6	1.7907E+02	2.9845E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	1	30	8.9304E+02	2.9768E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	2	30	8.9284E+02	2.9761E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	3	30	9.1647E+02	3.0549E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	4	30	8.9949E+02	2.9983E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	5	30	8.9606E+02	2.9869E+01
TOTAL RUN SIZE, TOTAL, MEAN	1	150	4.4979E+03	2.9986E+01

ANOVA

	SUM SQUARES	DEG FRDM	MEAN SUM SQUARES	VARIANCE
BTWN SAMPLES	12.9375	4	3.2344	0.1076
BTWN GAS ALIQUOTS	0.1563	20	0.0078	0.0005
BTWN MACHINE CYCLES	0.5781	125	0.0046	0.0046

RAW OXYGEN-18 CYCLE DATA: SAMPLE 127.7

CYCLE	MINOR GROUP	MAJOR GROUP
29.818	1	1
29.858	1	1
29.864	1	1
29.700	1	1
29.745	1	1
29.697	1	1
29.863	2	1
29.752	2	1
29.618	2	1
29.587	2	1
29.776	2	1
29.721	2	1
29.667	3	1
29.790	3	1
29.830	3	1
29.760	3	1
29.816	3	1
29.747	3	1
29.746	4	1
29.820	4	1
29.872	4	1
29.761	4	1
29.759	4	1
29.734	4	1
29.841	5	1
29.866	5	1
29.822	5	1
29.743	5	1
29.792	5	1
29.680	5	1
29.822	6	2
29.751	6	2
29.853	6	2
29.819	6	2
29.720	6	2
29.820	6	2
29.742	7	2
29.749	7	2
29.774	7	2
29.815	7	2
29.775	7	2
29.786	7	2
29.720	8	2
29.818	8	2
29.736	8	2
29.752	8	2
29.705	8	2
29.818	8	2
29.841	9	2
29.703	9	2
29.740	9	2
29.746	9	2
29.716	9	2
29.693	9	2
29.809	10	2
29.750	10	2
29.717	10	2
29.693	10	2
29.699	10	2
29.762	10	2
30.444	11	3
30.522	11	3
30.521	11	3
30.605	11	3
30.570	11	3
30.578	11	3

CYCLE	MINOR GROUP	MAJOR GROUP
30.636	12	3
30.540	12	3
30.683	12	3
30.668	12	3
30.441	12	3
30.530	12	3
30.495	13	3
30.584	13	3
30.519	13	3
30.618	13	3
30.609	13	3
30.566	13	3
30.617	14	3
30.460	14	3
30.473	14	3
30.492	14	3
30.539	14	3
30.521	14	3
30.411	15	3
30.395	15	3
30.648	15	3
30.587	15	3
30.592	15	3
30.603	15	3
29.976	16	4
29.931	16	4
29.953	16	4
29.901	16	4
29.992	16	4
29.971	16	4
29.997	17	4
30.019	17	4
30.001	17	4
29.988	17	4
30.032	17	4
29.998	17	4
30.057	18	4
30.084	18	4
30.017	18	4
30.023	18	4
29.897	18	4
30.026	18	4
29.902	19	4
29.853	19	4
30.067	19	4
29.958	19	4
29.868	19	4
29.975	19	4
29.876	20	4
30.016	20	4
30.074	20	4
29.970	20	4
30.062	20	4
30.002	20	4
29.848	21	5
29.870	21	5
29.947	21	5
29.984	21	5
29.970	21	5
29.911	21	5
29.816	22	5
29.842	22	5
29.973	22	5
29.996	22	5
29.986	22	5
29.943	22	5
29.766	23	5

CYCLE	MINOR GROUP	MAJOR GROUP
29.868	23	5
29.824	23	5
29.874	23	5
29.817	23	5
29.833	23	5
29.819	24	5
29.746	24	5
29.726	24	5
29.945	24	5
29.816	24	5
29.863	24	5
29.849	25	5
29.962	25	5
29.886	25	5
29.885	25	5
29.823	25	5
29.668	25	5

SAMPLE 127.7- CARBON-13

number of total points: 150
number of minor groups/major groups: 5
number of major groups: 5

MINOR GROUP #, SIZE, TOTAL, MEAN	1	6	2.6201E+02	4.3668E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	2	6	2.6203E+02	4.3672E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	3	6	2.6172E+02	4.3620E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	4	6	2.6225E+02	4.3708E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	5	6	2.6232E+02	4.3719E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	6	6	2.6214E+02	4.3689E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	7	6	2.6225E+02	4.3708E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	8	6	2.6232E+02	4.3720E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	9	6	2.6209E+02	4.3682E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	10	6	2.6212E+02	4.3687E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	11	6	2.6093E+02	4.3488E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	12	6	2.6089E+02	4.3481E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	13	6	2.6080E+02	4.3467E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	14	6	2.6094E+02	4.3489E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	15	6	2.6093E+02	4.3489E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	16	6	2.6221E+02	4.3702E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	17	6	2.6239E+02	4.3732E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	18	6	2.6222E+02	4.3703E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	19	6	2.6201E+02	4.3669E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	20	6	2.6204E+02	4.3673E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	21	6	2.6203E+02	4.3672E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	22	6	2.6218E+02	4.3696E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	23	6	2.6201E+02	4.3668E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	24	6	2.6181E+02	4.3635E+01
MINOR GROUP #, SIZE, TOTAL, MEAN	25	6	2.6191E+02	4.3652E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	1	30	1.3103E+03	4.3677E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	2	30	1.3109E+03	4.3697E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	3	30	1.3045E+03	4.3483E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	4	30	1.3109E+03	4.3696E+01
MAJOR GROUP #, SIZE, TOTAL, MEAN	5	30	1.3099E+03	4.3665E+01
TOTAL RUN SIZE, TOTAL, MEAN	1	150	6.5465E+03	4.3644E+01

ANOVA

	SUM SQUARES	DEG FRDM	MEAN SUM SQUARES	VARIANCE
BTWN SAMPLES	1.0938	4	0.2734	0.0090
BTWN GAS ALIQUOTS	0.0938	20	0.0047	0.0007
BTWN MACHINE CYCLES	0.0938	125	0.0008	0.0008

RAW CARBON-13 CYCLE DATA: SAMPLE 127.7

CYCLE	MINOR GROUP	MAJOR GROUP
43.690	1	1
43.683	1	1
43.629	1	1
43.607	1	1
43.699	1	1
43.700	1	1
43.672	2	1
43.691	2	1
43.673	2	1
43.671	2	1
43.691	2	1
43.634	2	1
43.595	3	1
43.546	3	1
43.671	3	1
43.610	3	1
43.653	3	1
43.643	3	1
43.723	4	1
43.686	4	1
43.725	4	1
43.669	4	1
43.757	4	1
43.690	4	1
43.735	5	1
43.748	5	1
43.688	5	1
43.684	5	1
43.746	5	1
43.714	5	1
43.703	6	2
43.678	6	2
43.689	6	2
43.735	6	2
43.689	6	2
43.641	6	2
43.748	7	2
43.704	7	2
43.626	7	2
43.686	7	2
43.763	7	2
43.720	7	2
43.770	8	2
43.717	8	2
43.700	8	2
43.668	8	2
43.711	8	2
43.756	8	2
43.770	9	2
43.621	9	2
43.673	9	2
43.667	9	2
43.721	9	2
43.637	9	2
43.716	10	2
43.756	10	2
43.732	10	2
43.678	10	2
43.586	10	2
43.656	10	2
43.415	11	3
43.478	11	3
43.509	11	3
43.546	11	3
43.481	11	3
43.498	11	3

CYCLE	MINOR GROUP	MAJOR GROUP
43.550	12	3
43.527	12	3
43.431	12	3
43.468	12	3
43.433	12	3
43.479	12	3
43.426	13	3
43.387	13	3
43.543	13	3
43.495	13	3
43.450	13	3
43.500	13	3
43.493	14	3
43.503	14	3
43.562	14	3
43.473	14	3
43.419	14	3
43.485	14	3
43.464	15	3
43.460	15	3
43.557	15	3
43.472	15	3
43.411	15	3
43.571	15	3
43.668	16	4
43.712	16	4
43.675	16	4
43.684	16	4
43.712	16	4
43.762	16	4
43.758	17	4
43.750	17	4
43.665	17	4
43.750	17	4
43.732	17	4
43.739	17	4
43.709	18	4
43.717	18	4
43.660	18	4
43.705	18	4
43.739	18	4
43.688	18	4
43.633	19	4
43.715	19	4
43.691	19	4
43.634	19	4
43.654	19	4
43.684	19	4
43.645	20	4
43.664	20	4
43.695	20	4
43.601	20	4
43.726	20	4
43.708	20	4
43.585	21	5
43.686	21	5
43.715	21	5
43.682	21	5
43.648	21	5
43.718	21	5
43.696	22	5
43.712	22	5
43.734	22	5
43.682	22	5
43.675	22	5
43.678	22	5
43.681	23	5

CYCLE	MINOR GROUP	MAJOR GROUP
43.669	23	5
43.681	23	5
43.636	23	5
43.658	23	5
43.681	23	5
43.657	24	5
43.696	24	5
43.627	24	5
43.588	24	5
43.634	24	5
43.607	24	5
43.663	25	5
43.686	25	5
43.659	25	5
43.597	25	5
43.609	25	5
43.700	25	5

Appendix F

Searles Lake core KM-3 Isotopic Results and Isotopic Correction Factors

APPENDIX F

SEARLES LAKE CORE KM-3: DEPTH, AGE, O-18 & C-13 RESULTS and CORRECTIONS
SAMPLES MARKED WITH 'XX' NOT USED IN RECONSTRUCTION DUE TO MINERALOGY

Note: Dates according to Jannik (1990) chronology

CORE DEPTH (ft)	CORE DEPTH (m)	C-13 DOL. (PDB)	O-18 DOL. (SMOW)	C-13 CORRECTIONS (per mil)	O-18 CORRECTIONS (per mil)	C-13 FINAL (PDB)	O-18 FINAL (SMOW)	AGE (Ma)
65.6	19.99	0.004	30.018	0.800	2.800	0.804	32.818	0.0102
66.1	20.15	2.383	36.988	-----	1.000	2.383	37.988	0.0106
66.6	20.30	2.905	34.463	-----	-----	2.905	34.463	0.0109
68.1	20.76	0.331	33.558	-----	1.000	0.331	34.558	0.0121
68.6	20.91	1.657	34.889	-----	-----	1.657	34.889	0.0125
69.1	21.06	0.677	35.569	-----	-----	0.677	35.569	0.0129
69.6	21.21	1.047	32.623	-----	-----	1.047	32.623	0.0133
70.1	21.37	1.357	34.228	-----	-----	1.357	34.228	0.0137
70.6	21.52	0.965	32.802	-----	-----	0.965	32.802	0.0141
71.0	21.64	1.961	32.964	-----	-----	1.961	32.964	0.0144
71.6	21.82	2.188	32.704	0.800	2.800	2.988	35.504	0.0149
72.1	21.98	2.701	33.525	0.800	2.800	3.501	36.325	0.0153
72.6	22.13	2.804	34.929	0.800	2.800	3.604	37.729	0.0156
73.1	22.28	3.122	35.847	-----	-----	3.122	35.847	0.0160
73.6	22.43	3.386	36.617	-----	-----	3.386	36.617	0.0164
74.1	22.59	3.424	34.055	0.800	2.800	4.224	36.855	0.0168
74.6	22.74	3.845	36.239	-----	-----	3.845	36.239	0.0172
75.3	22.95	4.491	37.373	0.800	2.800	5.291	40.173	0.0178
75.8	23.10	3.416	32.154	-----	-----	3.416	32.154	0.0182
80.0	24.38	3.515	37.715	-----	-----	3.515	37.715	0.0214
81.0	24.69	4.370	39.140	-----	-----	4.370	39.140	0.0222
81.6	24.87	3.719	35.356	-----	-----	3.719	35.356	0.0227
81.9	24.96	1.719	31.829	-----	-----	1.719	31.829	0.0229
89.5	27.28	3.537	32.878	0.800	2.800	4.337	35.678	0.0236
90.7	27.65	3.764	35.804	0.800	2.800	4.564	38.604	0.0247
91.5	27.89	3.890	36.372	0.800	2.800	4.690	39.172	0.0253
95.5	29.11	2.617	28.411	0.800	2.800	3.417	31.211	0.0260
96.6	29.44	1.830	26.935	0.800	2.800	2.630	29.735	0.0269
111.7	34.05	3.152	32.440	0.800	2.800	3.952	35.240	0.0290
113.3	34.53	3.799	33.907	0.800	2.800	4.599	36.707	0.0298
114.5	34.90	4.754	32.632	0.800	2.800	5.554	35.432	0.0306
116.4	35.48	4.245	33.476	0.800	2.800	5.045	35.276	0.0317
118.5	36.12	3.861	35.465	0.800	2.800	4.661	38.265	0.0319
120.0	36.58	4.081	35.993	0.800	2.800	4.881	38.793	0.0326
124.9	38.07	4.006	38.129	-----	1.000	4.006	39.129	0.0335
125.7	38.31	4.706	32.652	0.800	2.800	5.506	35.452	0.0356
126.3	38.50	4.070	38.909	-----	-----	4.070	38.909	0.0372
126.9	38.68	4.354	40.755	-----	-----	4.354	40.755	0.0388
127.7	38.92	3.412	38.345	-----	-----	3.412	38.345	0.0409
128.7	39.23	2.841	35.474	-----	-----	2.841	35.474	0.0435
128.9	39.29	3.775	39.432	-----	-----	3.775	39.432	0.0440
129.8	39.56	3.980	36.587	0.800	2.800	4.780	39.387	0.0464
130.1	39.65	3.412	39.934	-----	-----	3.412	39.934	0.0471
130.4	39.75	3.962	40.348	-----	-----	3.962	40.348	0.0480
131.5	40.08	3.575	37.927	-----	-----	3.575	37.927	0.0508
132.3	40.33	1.831	37.379	-----	-----	1.831	37.379	0.0530
134.4	40.97	3.659	36.075	-----	1.000	3.659	37.075	0.0585
136.4	41.57	3.407	37.320	-----	1.000	3.407	38.320	0.0637
137.5	41.91	4.278	37.353	-----	1.000	4.278	38.353	0.0659
139.0	42.37	2.226	36.486	-----	-----	2.226	36.486	0.0672
140.5	42.82	3.124	40.365	-----	-----	3.124	40.365	0.0685
141.8	43.22	2.885	37.364	-----	-----	2.885	37.364	0.0696
142.8	43.53	3.078	40.448	-----	-----	3.078	40.448	0.0705
151.1	46.06	4.981	39.621	-----	-----	4.981	39.621	0.0779
153.0	46.63	3.571	33.188	0.800	2.800	4.371	35.988	0.0796
154.1	46.97	3.589	33.052	0.800	2.800	4.389	35.852	0.0806
158.6	48.34	6.557	41.891	-----	-----	6.557	41.891	0.0846
164.0	49.99	6.237	41.768	-----	-----	6.237	41.768	0.0894
170.0	51.82	4.521	40.515	-----	-----	4.521	40.515	0.0952

CORE DEPTH (ft)	CORE DEPTH (m)	C-13 DOL. (PDB)	O-18 DOL. (SMOW)	C-13 CORRECTIONS (per mil)	O-18	C-13 FINAL (PDB)	O-18 FINAL (SMOW)	AGE (Ma)
178.5	54.41	4.568	37.689	-----	-----	4.568	37.689	0.1034
183.0	55.78	4.749	37.390	-----	-----	4.749	37.390	0.1072
187.4	57.12	5.592	38.764	-----	-----	5.592	38.764	0.1117
189.4	57.73	4.347	34.780	-----	1.000	4.347	35.780	0.1138
191.9	58.49	2.984	30.941	0.800	2.800	3.784	33.741	0.1163
194.9	59.41	5.319	40.123	-----	-----	5.319	40.123	0.1194
199.9	60.93	5.999	41.715	-----	-----	5.999	41.715	0.1247
205.0	62.48	5.098	38.463	-----	-----	5.098	38.463	0.1303
211.1	64.34	3.507	33.198	0.800	2.800	4.307	35.998	0.1370
218.0	66.45	5.956	40.727	-----	-----	5.956	40.727	0.1446
221.0	67.36	5.763	40.267	-----	-----	5.763	40.267	0.1479
224.1	68.31	6.425	41.175	-----	-----	6.425	41.175	0.1513
226.6	69.07	6.566	42.131	-----	-----	6.566	42.131	0.1540
256.1	78.06	5.590	38.712	-----	-----	5.590	38.712	0.1806
257.0	78.33	5.282	36.223	-----	-----	5.282	36.223	0.1814
277.5	84.58	4.921	34.702	-----	-----	4.921	34.702	0.2006
288.1	87.81	6.406	23.423	0.800	2.800	7.206	26.223	0.2122
298.1	90.86	2.369	24.906	0.800	2.800	3.169	27.706	0.2231
308.0	93.88	3.768	30.120	0.800	2.800	4.568	32.920	0.2303
XX 326.5	99.52	4.465	26.841	0.800	2.800	5.265	29.641	0.2629
XX 349.9	106.65	4.730	32.428	-----	-----	4.730	32.428	0.2710
XX 359.3	109.51	5.381	29.988	-----	-----	5.381	29.988	0.2735
369.5	112.62	4.890	23.163	0.800	2.800	5.690	25.963	0.2788
370.5	112.93	3.558	23.247	0.800	2.800	4.358	26.047	0.2804
371.8	113.32	4.073	21.541	0.800	2.800	4.873	24.342	0.2825
373.3	113.78	4.732	31.709	0.800	2.800	5.532	34.509	0.2849
375.0	114.30	4.965	27.420	0.800	2.800	5.765	30.220	0.2893
446.1	135.97	4.102	30.735	0.800	2.800	4.902	33.535	0.3552
447.5	136.40	3.892	28.274	0.800	2.800	4.692	31.074	0.3620
448.5	136.70	3.560	28.514	0.800	2.800	4.360	31.314	0.3668
449.4	136.98	3.599	26.006	0.800	2.800	4.399	28.806	0.3711
453.0	138.07	6.623	35.706	-----	-----	6.623	35.706	0.3760
XX 455.5	138.84	5.429	37.563	-----	-----	5.429	37.563	0.3810
XX 457.0	139.29	3.624	29.094	-----	-----	3.624	29.094	0.3828
459.5	140.06	4.951	26.749	0.800	2.800	5.751	29.544	0.3852
476.0	145.09	4.457	31.139	0.800	2.800	5.257	33.939	0.3905
481.0	146.61	3.899	32.837	0.800	2.800	4.699	35.637	0.3930
483.4	147.34	4.741	30.302	0.800	2.800	5.541	33.102	0.3985
484.4	147.65	4.372	30.112	0.800	2.800	5.172	32.912	0.4008
485.7	148.04	4.883	25.332	0.800	2.800	5.683	28.132	0.4038
486.0	148.13	4.848	28.408	0.800	2.800	5.648	31.208	0.4045
487.8	148.68	4.377	28.848	0.800	2.800	5.177	31.648	0.4087
489.0	149.05	3.761	31.085	0.800	2.800	4.561	33.885	0.4115
XX 490.0	149.35	4.079	31.244	-----	-----	4.079	31.000	0.4138
491.2	149.72	3.755	31.014	0.800	2.800	4.555	33.814	0.4165
492.0	149.96	2.985	30.480	0.800	2.800	3.785	33.280	0.4184
493.2	150.33	3.429	30.686	0.800	2.800	4.229	33.486	0.4212
493.9	150.54	2.711	35.253	0.800	2.800	3.511	38.053	0.4228
502.1	153.04	3.311	30.935	0.800	2.800	4.111	33.735	0.4301
546.1	166.45	2.765	31.910	0.800	2.800	3.565	34.710	0.5982
547.4	166.85	2.214	32.187	0.800	2.800	3.014	34.987	0.6004
547.7	166.94	1.945	32.628	0.800	2.800	2.745	35.428	0.6009
548.5	167.18	1.455	35.075	0.800	2.800	2.255	37.875	0.6022
549.5	167.49	2.285	33.221	0.800	2.800	3.085	36.021	0.6039
550.5	167.79	1.591	33.215	-----	1.000	1.591	33.415	0.6056
551.7	168.16	3.607	31.163	0.800	2.800	4.407	33.963	0.6076
552.6	168.43	1.878	28.600	0.800	2.800	2.678	31.400	0.6091
553.8	168.80	3.271	31.311	0.800	2.800	4.071	34.111	0.6111
555.0	169.16	1.634	32.290	-----	1.000	1.634	33.290	0.6132
556.0	169.47	3.003	32.705	-----	1.000	3.003	33.705	0.6149
557.1	169.80	4.053	38.224	-----	-----	4.053	38.224	0.6168
558.2	170.14	1.280	27.901	-----	-----	1.280	27.901	0.6187
559.2	170.44	1.964	32.186	-----	-----	1.964	32.186	0.6205
560.1	170.72	3.577	32.874	-----	-----	3.577	32.874	0.6220
561.0	170.99	2.147	35.478	-----	-----	2.147	35.478	0.6236

	CORE DEPTH (ft)	CORE DEPTH (m)	C-13 DOL. (PDB)	O-18 DOL. (SMOW)	C-13 CORRECTIONS (per mil)	O-18	C-13 FINAL (PDB)	O-18 FINAL (SMOW)	AGE (Ma)
	562.2	171.36	1.121	34.479	-----	1.000	1.121	35.479	0.6257
	563.2	171.66	1.296	30.667	-----	-----	1.296	30.667	0.6274
	564.3	172.00	2.333	32.120	-----	-----	2.333	32.120	0.6293
	565.3	172.30	2.461	31.986	-----	-----	2.461	31.986	0.6310
	567.0	172.82	2.027	33.316	-----	-----	2.027	33.316	0.6340
	568.7	173.34	3.073	35.171	-----	-----	3.073	35.171	0.6369
	569.9	173.71	6.209	40.770	-----	-----	6.209	40.770	0.6390
	571.2	174.10	3.738	34.551	-----	-----	3.738	34.551	0.6412
	572.5	174.50	4.834	39.142	-----	-----	4.834	39.142	0.6435
	573.8	174.89	4.646	37.815	-----	-----	4.646	37.815	0.6457
	574.9	175.23	5.332	39.691	-----	-----	5.332	39.691	0.6476
	575.4	175.38	4.182	35.143	-----	-----	4.182	35.143	0.6485
	576.4	175.69	4.677	37.685	-----	-----	4.677	37.685	0.6502
	577.6	176.05	4.320	36.589	0.800	2.800	5.120	39.389	0.6523
	578.6	176.36	4.971	30.339	-----	-----	4.971	30.339	0.6540
	579.5	176.63	4.037	32.522	-----	-----	4.037	32.522	0.6556
	580.7	177.00	4.475	34.149	-----	-----	4.475	34.149	0.6577
	582.2	177.45	4.201	33.392	0.800	2.800	5.001	36.192	0.6603
XX	583.3	177.79	4.681	31.669	-----	-----	4.681	31.669	0.6622
XX	584.2	178.06	3.562	21.455	-----	-----	3.562	21.455	0.6637
	587.8	179.16	2.644	32.359	0.800	2.800	3.444	35.159	0.6697
	590.9	180.11	5.838	41.693	-----	-----	5.838	41.693	0.6794
XX	592.6	180.62	4.184	33.944	-----	-----	4.184	33.944	0.6848
	594.8	181.30	5.488	42.246	-----	-----	5.488	42.246	0.6917
	598.5	182.42	4.162	30.279	-----	-----	4.162	30.279	0.7034
XX	601.5	183.34	6.636	23.732	-----	-----	6.636	23.732	0.7128
XX	603.0	183.79	5.326	36.755	-----	-----	5.326	36.755	0.7175
XX	605.0	184.40	4.765	27.429	-----	-----	4.765	27.429	0.7238
XX	607.9	185.29	5.768	34.523	-----	-----	5.768	34.523	0.7339
XX	609.9	185.90	6.254	39.206	-----	-----	6.254	39.206	0.7422
XX	611.5	186.39	6.349	27.093	-----	-----	6.349	27.093	0.7489
XX	615.0	187.45	6.030	31.304	-----	-----	6.030	31.304	0.7604
	624.0	190.20	4.279	28.077	0.800	2.800	5.079	30.877	0.7826
	631.8	192.57	3.304	31.200	0.800	2.800	4.104	34.000	0.8037
	634.2	193.30	4.220	25.842	-----	-----	4.220	25.842	0.8099
	636.4	193.98	4.679	26.805	-----	-----	4.679	26.805	0.8156
XX	638.5	194.62	4.470	28.589	-----	-----	4.470	28.589	0.8210
	641.1	195.41	5.353	30.612	-----	-----	5.353	30.612	0.8278
	642.5	195.83	4.143	32.740	-----	1.000	4.143	33.740	0.8314
	645.2	196.66	4.076	32.534	-----	1.000	4.076	33.534	0.8350
	647.4	197.33	2.675	35.727	0.800	2.800	3.475	38.527	0.8402
	651.4	198.55	3.440	36.050	0.800	2.800	4.240	38.850	0.8496
	653.6	199.22	4.500	35.557	-----	-----	4.500	35.557	0.8548
	659.2	200.92	4.736	33.899	-----	-----	4.736	33.899	0.8681
	661.7	201.69	4.498	31.425	-----	-----	4.498	31.425	0.8740
	664.8	202.63	5.707	38.055	-----	-----	5.707	38.055	0.8813
	666.5	203.15	3.960	35.178	-----	-----	3.960	35.178	0.8854
	668.4	203.73	5.562	39.513	-----	-----	5.562	39.513	0.8899
	670.2	204.28	6.449	33.316	-----	-----	6.449	33.316	0.8941
	671.5	204.67	3.988	31.171	-----	-----	3.988	31.171	0.8976
	680.7	207.48	5.768	38.904	-----	-----	5.768	38.904	0.9042
	684.7	208.70	6.016	39.346	-----	-----	6.016	39.346	0.9141
	690.3	210.40	4.673	32.509	-----	-----	4.673	32.509	0.9278
XX	701.2	213.73	5.930	38.156	-----	-----	5.930	38.156	0.9352
	706.2	215.25	3.374	34.371	0.800	2.800	4.174	37.171	0.9409
	710.2	216.47	2.598	33.392	0.800	2.800	3.398	36.192	0.9454
	713.4	217.44	3.035	36.275	0.800	2.800	3.835	39.075	0.9490
	717.0	218.54	2.861	28.766	0.800	2.800	3.661	32.566	0.9530
	724.9	220.95	3.403	29.145	0.800	2.800	4.203	31.945	0.9684
	727.5	221.74	3.625	24.522	0.800	2.800	4.425	27.322	0.9718
	747.2	227.75	6.008	34.406	-----	-----	6.008	34.406	0.9919
	750.0	228.60	5.971	41.628	-----	-----	5.971	41.628	0.9948
XX	752.5	229.36	4.160	24.314	-----	-----	4.160	24.314	0.9973

CORE DEPTH (ft)	CORE DEPTH (m)	C-13 DOL. (PDB)	O-18 DOL. (SMOW)	C-13 CORRECTIONS (per mil)	O-18	C-13 FINAL (PDB)	O-18 FINAL (SMOW)	AGE (Ma)
758.0	231.04	6.272	39.973	-----	-----	6.272	39.973	1.0029
762.4	232.38	6.394	40.565	-----	-----	6.394	40.565	1.0074
766.5	233.63	3.581	35.114	-----	1.000	3.581	36.114	1.0116
772.2	235.37	6.234	38.277	-----	-----	6.234	38.277	1.0174
775.8	236.46	5.773	38.988	-----	-----	5.773	38.988	1.0211
780.8	237.99	4.861	30.802	-----	-----	4.861	30.802	1.0261
785.9	239.54	5.964	38.269	-----	-----	5.964	38.269	1.0313
794.2	242.07	6.178	40.195	-----	-----	6.178	40.195	1.0398
798.7	243.44	6.063	38.766	-----	-----	6.063	38.766	1.0444
805.0	245.36	4.262	32.830	-----	-----	4.262	32.830	1.0508
814.0	248.11	4.546	33.949	-----	-----	4.546	33.949	1.0600
818.2	249.39	5.270	33.377	-----	1.000	5.270	34.377	1.0637
823.6	251.03	4.384	32.283	0.800	2.800	5.184	35.083	1.0718
827.0	252.07	5.173	33.282	-----	-----	5.173	33.282	1.0771
831.7	253.50	5.348	36.596	-----	-----	5.348	36.596	1.0843
837.0	255.12	6.809	37.489	-----	-----	6.809	37.489	1.0924
841.6	256.52	6.166	37.373	-----	-----	6.166	37.373	1.0995
846.6	258.04	4.465	35.381	-----	1.000	4.465	36.381	1.1072
859.8	262.07	4.366	33.866	-----	-----	4.366	33.866	1.1274
864.0	263.35	6.830	40.501	-----	-----	6.830	40.501	1.1339
876.2	267.07	6.735	38.187	-----	-----	6.735	38.187	1.1526
XX 880.9	268.50	NO CARBONATE		-----	-----	-----	-----	1.1598
887.6	270.54	4.786	33.328	-----	1.000	4.786	34.328	1.1670
893.9	272.46	4.384	31.279	0.800	2.800	5.184	34.079	1.1757
898.9	273.99	4.473	29.276	0.800	2.800	5.273	32.076	1.1840
903.1	275.27	6.496	36.371	-----	-----	6.496	36.371	1.1910
XX 908.6	276.94	4.099	33.032	-----	-----	4.099	33.032	1.2002
917.1	279.53	5.022	34.559	0.800	2.800	5.822	37.359	1.2126
922.0	281.03	4.839	34.751	-----	1.000	4.839	35.751	1.2198
930.0	283.46	4.408	34.554	-----	-----	4.408	34.554	1.2315
935.7	285.20	2.905	35.058	-----	-----	2.905	35.058	1.2398
939.9	286.48	2.759	32.046	-----	-----	2.759	32.046	1.2459
948.0	288.95	2.330	34.153	0.800	2.800	3.130	36.953	1.2578
955.0	291.08	3.307	35.125	-----	-----	3.307	35.125	1.2680
959.0	292.30	1.909	35.548	0.800	2.800	2.709	38.348	1.2693
966.0	294.44	0.290	32.647	0.800	2.800	1.090	35.447	1.2730
972.1	296.30	2.091	29.356	-----	-----	2.091	29.356	1.2909
977.0	297.79	0.439	33.098	-----	-----	0.439	33.098	1.3053
986.0	300.53	-0.045	34.680	0.800	2.800	0.755	37.480	1.3218
1006.0	306.63	3.369	33.397	0.800	2.800	4.169	36.197	1.3312
1012.0	308.46	-0.076	31.653	0.800	2.800	0.724	34.453	1.3350
1017.0	309.98	4.825	31.282	-----	-----	4.825	31.282	1.3403
1034.2	315.22	5.640	36.016	-----	-----	5.640	36.016	1.3585
1068.0	325.53	-9.314	23.058	-----	-----	-9.314	23.058	1.4080
1076.0	327.97	-1.436	29.713	0.800	2.800	-0.636	32.513	1.4101
1080.4	329.31	-2.507	27.552	0.800	2.800	-1.707	30.352	1.4244
1089.0	331.93	-0.672	31.240	0.800	2.800	0.128	34.040	1.4366
1093.0	333.15	4.686	40.200	0.800	2.800	5.486	43.000	1.4500
1099.0	334.98	-5.207	28.305	0.800	2.800	-4.407	31.105	1.4535
1104.0	336.50	-5.748	26.033	0.800	2.800	-4.948	28.833	1.4712
1106.5	337.26	-4.479	25.626	0.800	2.800	-3.679	28.426	1.4800
1110.6	338.51	-6.709	28.633	0.800	2.800	-5.909	31.433	1.4846
1115.3	339.94	-4.257	24.794	0.800	2.800	-3.457	27.594	1.4991
1120.0	341.38	-3.085	27.424	0.800	2.800	-2.285	30.224	1.5112
1125.5	343.05	-0.470	29.556	0.800	2.800	0.330	32.356	1.5284
1130.5	344.58	3.095	33.892	-----	-----	3.095	33.892	1.5441
1132.7	345.25	3.498	34.063	-----	-----	3.498	34.063	1.5510
1156.5	352.50	-0.116	29.908	0.800	2.800	0.684	32.708	1.5762
1210.7	369.02	0.422	32.659	0.800	2.800	1.222	35.459	1.6230
1294.0	394.41	-1.131	28.458	0.800	2.800	-0.331	31.258	1.8127

Additional Data from KM-3

Depth (m)	$\delta^{13}C$ (‰)	$\delta^{18}O$ (‰)	Corrected $\delta^{18}O$ (‰)	Age (ka)
183.8	-	39.1	37.9	717.5
184.5	-	35.4	34.2	723.8
185.3	3.8	36.5	35.3	733.9
185.9	3.2	34.6	33.4	742.2
186.2	4.1	36.9	35.7	748.9
187.5	3.6	36.8	35.6	760.4

Additional Data from Core LDW-6

Note: no mineralogical corrections were required in this interval

Depth (m)	$\delta^{13}C$ (‰)	$\delta^{18}O$ (‰)	Age (ka)
69.0	6.0	40.8	161.3
69.6	7.7	42.2	163
70.0			
70.3	6.9	38.6	164.6
70.5	5.4	37.6	165.2
70.8	5.9	38.2	166.0
71.0	6.3	39.5	166.9
71.5	5.9	38.0	168.0
71.5	5.7	37.4	168.3
72.3	5.4	37.0	170.2
72.6	3.1	30.2	171.3
73.3	6.3	38.8	173.0
73.6	6.3	38.9	174.1
78.6	5.8	31.5	188.1
78.8	6.3	29.9	188.3
79.1	5.8	28.8	189.5
79.5	5.6	24.0	190.3
79.6	5.5	26.1	190.8
80.6	4.3	28.6	193.6
81.4	4.4	29.0	195.6
81.6	4.8	31.6	196.4
82.0	2.0	27.9	197.3
82.6	4.5	29.7	199.2
83.4	6.0	35.9	201.1
83.5	4.3	31.5	201.7
83.8	4.2	37.7	202.3
83.9	4.6	31.5	202.6
84.1	4.5	33.1	203.4
84.4	6.7	29.5	203.9
84.5	6.0	27.2	204.5
89.5	5.1	35.4	218.4
89.8	5.5	37.6	219.0
90.0	4.4	33.0	219.8
90.4	3.6	32.3	220.7
90.6	4.1	31.0	221.5
91.0	4.3	32.5	222.3
91.1	4.5	33.6	222.9
91.5	4.8	35.8	223.7
91.9	2.8	30.2	224.8
92.4	3.5	27.6	226.3
92.9	5.6	36.2	227.9
93.0	3.2	28.8	228.2
93.5	3.2		229.6
93.8	4.7	32.7	230.1
95.0	4.0	34.6	233.8
95.3	4.2	31.1	234.3
95.5	5.6	29.9	234.9
95.8	4.0	32.4	235.7
96.0	5.9	37.5	236.6
96.4	5.8	38.4	237.4
96.8	6.4	40.1	238.5
97.0	6.4		239.3
97.4	5.7	28.5	240.2

97.6	6.7	28.8	241.0
98.1	5.5	26.0	242.4
98.4	4.9	33.8	243.0
98.6	6.6	22.0	243.8
99.0	5.7	27.6	244.6
99.1	5.0	32.2	245.2
100.1	5.4	39.3	248.0

****

APPENDIX F

****

Note: Combined samples from KM-3 and LDW-(Chronology accordi[illegible] to this report.

CORE DEPTH (m)	CORE DEPTH (ft)	C-13 FINAL (PDB)	O-18 FINAL (SMOW)	AGE (Ma)
19.99	65.6	0.804	32.818	0.0102
20.15	66.1	2.383	37.988	0.0106
20.30	66.6	2.905	34.463	0.0109
20.76	68.1	0.331	34.558	0.0121
20.91	68.6	1.657	34.889	0.0125
21.06	69.1	0.677	35.569	0.0129
21.21	69.6	1.047	32.623	0.0133
21.37	70.1	1.357	34.228	0.0137
21.52	70.6	0.965	32.802	0.0141
21.64	71.0	1.961	32.964	0.0144
21.82	71.6	2.988	35.504	0.0149
21.98	72.1	3.501	36.325	0.0153
22.13	72.6	3.604	37.729	0.0156
22.28	73.1	3.122	35.847	0.0160
22.43	73.6	3.386	36.617	0.0164
22.59	74.1	4.224	36.855	0.0168
22.74	74.6	3.845	36.239	0.0172
22.95	75.3	5.291	40.173	0.0178
23.10	75.8	3.416	32.154	0.0182
24.38	80.0	3.515	37.715	0.0214
24.69	81.0	4.370	39.140	0.0222
24.87	81.6	3.719	35.356	0.0227
24.96	81.9	1.719	31.829	0.0229
27.28	89.5	4.337	35.678	0.0236
27.65	90.7	4.564	38.604	0.0247
27.89	91.5	4.690	39.172	0.0253
29.11	95.5	3.417	31.211	0.0260
29.44	96.6	2.630	29.735	0.0269
34.05	111.7	3.952	35.240	0.0290
34.53	113.3	4.599	36.707	0.0298
34.90	114.5	5.554	35.432	0.0306
35.48	116.4	5.045	35.276	0.0317
36.12	118.5	4.661	38.265	0.0319
36.58	120.0	4.881	38.793	0.0326
38.07	124.9	4.006	39.129	0.0335
38.31	125.7	5.506	35.452	0.0356
38.50	126.3	4.070	38.909	0.0372
38.68	126.9	4.354	40.755	0.0388
38.92	127.7	3.412	38.345	0.0409
39.23	128.7	2.841	35.474	0.0435
39.29	128.9	3.775	39.432	0.0440
39.56	129.8	4.780	39.387	0.0464
39.65	130.1	3.412	39.934	0.0471
39.75	130.4	3.962	40.348	0.0480
40.08	131.5	3.575	37.927	0.0508
40.33	132.3	1.831	37.379	0.0530

40.97	134.4	3.659	37.075	0.0585
41.57	136.4	3.407	38.320	0.0637
41.91	137.5	4.278	38.353	0.0659
42.37	139.0	2.226	36.486	0.0672
42.82	140.5	3.124	40.365	0.0685
43.22	141.8	2.885	37.364	0.0696
43.53	142.8	3.078	40.448	0.0705
46.06	151.1	4.981	39.621	0.0779
46.63	153.0	4.371	35.988	0.0796
46.97	154.1	4.389	35.852	0.0806
48.34	158.6	6.557	41.891	0.0846
49.99	164.0	6.237	41.768	0.0894
51.82	170.0	4.521	40.515	0.0952
54.41	178.5	4.568	37.689	0.1034
55.78	183.0	4.749	37.390	0.1072
57.12	187.4	5.592	38.764	0.1117
57.73	189.4	4.347	35.780	0.1138
58.49	191.9	3.784	33.741	0.1163
59.41	194.9	5.319	40.123	0.1194
60.93	199.9	5.999	41.715	0.1247
62.48	205.0	5.098	38.463	0.1303
64.34	211.1	4.307	35.998	0.1370
66.45	218.0	5.956	40.727	0.1446
67.36	221.0	5.763	40.267	0.1479
68.31	224.1	6.425	41.175	0.1513
69.00	226.4	6.084	40.800	0.1613
69.60	228.3	7.725	42.200	0.1630
70.30	230.6	6.960	38.600	0.1646
70.50	231.3	5.496	37.600	0.1652
70.80	232.3	5.959	38.200	0.1660
71.00	232.9	6.394	39.500	0.1669
71.50	234.6	5.952	38.000	0.1680
71.50	234.6	5.711	37.400	0.1683
72.30	237.2	5.419	37.000	0.1702
72.60	238.2	3.185	30.200	0.1713
73.30	240.5	6.379	38.800	0.1730
73.60	241.5	6.359	38.900	0.1741
78.60	257.9	5.807	31.500	0.1881
78.80	258.5	6.318	29.900	0.1883
79.10	259.5	5.821	28.800	0.1895
79.50	260.8	5.690	24.000	0.1903
79.60	261.2	5.562	26.100	0.1908
80.60	264.4	4.356	28.600	0.1936
81.40	267.1	4.447	29.000	0.1956
81.60	267.7	4.807	31.600	0.1964
82.00	269.0	2.058	27.900	0.1973
82.60	271.0	4.586	29.700	0.1992
83.40	273.6	6.039	35.900	0.2011
83.50	274.0	4.339	31.500	0.2017
83.80	274.9	4.243	37.700	0.2023
83.90	275.3	4.626	31.500	0.2026
84.10	275.9	4.595	33.100	0.2034
84.40	276.9	6.745	29.500	0.2039
84.50	277.2	6.043	28.030	0.2045
89.50	293.6	5.189	35.400	0.2184
89.80	294.6	5.550	37.600	0.2190
90.00	295.3	4.401	33.000	0.2198
90.40	296.6	3.621	32.300	0.2207
90.60	297.2	4.165	31.000	0.2215
91.00	298.6	4.332	32.500	0.2223

91.10	298.9	4.586	33.600	0.2229
91.50	300.2	4.871	35.800	0.2237
91.90	301.5	2.886	30.200	0.2248
92.40	303.1	3.515	28.360	0.2263
93.00	305.1	5.680	36.200	0.2279
93.50	306.8	3.238	29.490	0.2296
93.80	307.7	4.718	32.700	0.2301
95.00	311.7	4.094	34.600	0.2338
95.30	312.7	4.288	31.100	0.2343
95.50	313.3	5.683	29.900	0.2349
95.80	314.3	4.098	32.400	0.2357
96.00	315.0	5.948	37.500	0.2366
96.40	316.3	5.807	38.400	0.2374
96.80	317.6	6.488	40.100	0.2385
97.40	319.6	5.710	28.500	0.2402
97.60	320.2	6.761	28.800	0.2410
98.10	321.9	5.561	26.000	0.2424
98.40	322.8	4.993	33.800	0.2430
98.60	323.5	6.671	22.000	0.2438
99.00	324.8	5.757	27.600	0.2446
99.10	325.1	5.032	32.200	0.2452
100.10	328.4	5.444	39.300	0.2480
112.62	369.5	5.690	25.963	0.2788
112.93	370.5	4.358	26.047	0.2804
113.32	371.8	4.873	24.342	0.2825
113.78	373.3	5.532	34.509	0.2849
114.30	375.0	5.765	30.220	0.2893
135.97	446.1	4.902	33.535	0.3552
136.40	447.5	4.692	31.074	0.3620
136.70	448.5	4.360	31.314	0.3668
136.98	449.4	4.399	28.806	0.3711
138.07	453.0	6.623	35.706	0.3760
140.06	459.5	5.751	29.544	0.3852
145.09	476.0	5.257	33.939	0.4135
146.61	481.0	4.699	35.637	0.4160
147.34	483.4	5.541	33.102	0.4215
147.65	484.4	5.172	32.912	0.4238
148.04	485.7	5.683	28.132	0.4268
148.13	486.0	5.648	31.208	0.4275
148.68	487.8	5.177	31.648	0.4317
149.05	489.0	4.561	33.885	0.4345
149.72	491.2	4.555	33.814	0.4395
149.96	492.0	3.785	33.280	0.4414
150.33	493.2	4.229	33.486	0.4442
150.54	493.9	3.511	38.053	0.4458
153.04	502.1	4.111	33.735	0.4531
166.45	546.1	3.565	34.710	0.5982
166.85	547.4	3.014	34.987	0.6004
166.94	547.7	2.745	35.428	0.6009
167.18	548.5	2.255	37.875	0.6022
167.49	549.5	3.085	36.021	0.6039
167.79	550.5	1.591	33.415	0.6056
168.16	551.7	4.407	33.963	0.6076
168.43	552.6	2.678	31.400	0.6091
168.80	553.8	4.071	34.111	0.6111
169.16	555.0	1.634	33.290	0.6132
169.47	556.0	3.003	33.705	0.6149
169.80	557.1	4.053	38.224	0.6168
170.14	558.2	1.280	27.901	0.6187
170.44	559.2	1.964	32.186	0.6205

170.72	560.1	3.577	32.874	0.6220
170.99	561.0	2.147	35.478	0.6236
171.36	562.2	1.121	35.479	0.6257
171.66	563.2	1.296	30.667	0.6274
172.00	564.3	2.333	32.120	0.6293
172.30	565.3	2.461	31.986	0.6310
172.82	567.0	2.027	33.316	0.6340
173.34	568.7	3.073	35.171	0.6369
173.71	569.9	6.209	40.770	0.6390
174.10	571.2	3.738	34.551	0.6412
174.50	572.5	4.834	39.142	0.6435
174.89	573.8	4.646	37.815	0.6457
175.23	574.9	5.332	39.691	0.6476
175.38	575.4	4.182	35.143	0.6485
175.69	576.4	4.677	37.685	0.6502
176.05	577.6	5.120	39.389	0.6523
176.36	578.6	4.971	30.339	0.6540
176.63	579.5	4.037	32.522	0.6556
177.00	580.7	4.475	34.149	0.6577
177.45	582.2	5.001	36.192	0.6603
179.16	587.8	3.444	35.159	0.6697
180.11	590.9	5.838	41.693	0.6794
181.30	594.8	5.488	42.246	0.6917
182.42	598.5	4.162	30.279	0.7034
183.80	603.0	4.830	37.900	0.7175
185.30	607.9	3.837	35.300	0.7339
185.90	609.9	3.221	33.400	0.7422
186.20	610.9	3.651	35.700	0.7489
187.50	615.2	4.078	35.600	0.7604
190.20	624.0	5.079	30.877	0.7826
192.57	631.8	4.104	34.000	0.8037
193.30	634.2	4.220	25.842	0.8099
193.98	636.4	4.679	26.805	0.8156
195.41	641.1	5.353	30.612	0.8278
195.83	642.5	4.143	33.740	0.8314
196.66	645.2	4.076	33.534	0.8350
197.33	647.4	3.475	38.527	0.8402
198.55	651.4	4.240	38.850	0.8496
199.22	653.6	4.500	35.557	0.8548
200.92	659.2	4.736	33.899	0.8681
201.69	661.7	4.498	31.425	0.8740
202.63	664.8	5.707	38.055	0.8813
203.15	666.5	3.960	35.178	0.8854
203.73	668.4	5.562	39.513	0.8899
204.28	670.2	6.449	33.316	0.8941
204.67	671.5	3.988	31.171	0.8976
207.48	680.7	5.768	38.904	0.9042
208.70	684.7	6.016	39.346	0.9141
210.40	690.3	4.673	32.509	0.9278
215.25	706.2	4.174	37.171	0.9409
216.47	710.2	3.398	36.192	0.9454
217.44	713.4	3.835	39.075	0.9490
218.54	717.0	3.661	32.566	0.9530
220.95	724.9	4.203	31.945	0.9684
221.74	727.5	4.425	27.322	0.9708
227.75	747.2	6.008	34.406	0.9801
228.60	750.0	5.971	41.628	0.9814
231.04	758.0	6.272	39.973	0.9852
232.38	762.4	6.394	40.565	0.9873
233.63	766.5	3.581	36.114	0.9892

235.37	772.2	6.234	38.277	0.9919
236.46	775.8	5.773	38.988	0.9936
237.99	780.8	4.861	30.802	0.9959
239.54	785.9	5.964	38.269	0.9983
242.07	794.2	6.178	40.195	1.0022
243.44	798.7	6.063	38.766	1.0043
245.36	805.0	4.262	32.830	1.0073
248.11	814.0	4.546	33.949	1.0115
249.39	818.2	5.270	34.377	1.0132
251.03	823.6	5.184	35.083	1.0170
252.07	827.0	5.173	33.282	1.0194
253.50	831.7	5.348	36.596	1.0227
255.12	837.0	6.809	37.489	1.0265
256.52	841.6	6.166	37.373	1.0297
258.04	846.6	4.465	36.381	1.0333
262.07	859.8	4.366	33.866	1.0426
263.35	864.0	6.830	40.501	1.0456
267.07	876.2	6.735	38.187	1.0542
270.54	887.6	4.786	34.328	1.0609
272.46	893.9	5.184	34.079	1.0649
273.99	898.9	5.273	32.076	1.0687
275.27	903.1	6.496	36.371	1.0719
279.53	917.1	5.822	37.359	1.0819
281.03	922.0	4.839	35.751	1.0852
283.46	930.0	4.408	34.554	1.0906
285.20	935.7	2.905	35.058	1.0945
286.48	939.9	2.759	32.046	1.0973
288.95	948.0	3.130	36.953	1.1028
291.08	955.0	3.307	35.125	1.1101
292.30	959.0	2.709	38.348	1.1111
294.44	966.0	1.090	35.447	1.1138
296.30	972.1	2.091	29.356	1.1267
297.79	977.0	0.439	33.098	1.1372
300.53	986.0	0.755	37.480	1.1491
306.63	1006.0	4.169	36.197	1.1559
308.46	1012.0	0.724	34.453	1.1587
309.98	1017.0	4.825	31.282	1.1625
315.22	1034.2	5.640	36.016	1.1757
325.53	1068.0	-9.314	23.058	1.2116
327.97	1076.0	-0.636	32.513	1.2131
329.31	1080.4	-1.707	30.352	1.2235
331.93	1089.0	0.128	34.040	1.2323
333.15	1093.0	5.486	43.000	1.2420
334.98	1099.0	-4.407	31.105	1.2446
336.50	1104.0	-4.948	28.833	1.2574
337.26	1106.5	-3.679	28.426	1.2638
338.51	1110.6	-5.909	31.433	1.2671
339.94	1115.3	-3.457	27.594	1.2776
341.38	1120.0	-2.285	30.224	1.2864
343.05	1125.5	0.330	32.356	1.2988
344.58	1130.5	3.095	33.892	1.3102
345.25	1132.7	3.498	34.063	1.3152
352.50	1156.5	0.684	32.708	1.3335
369.02	1210.7	1.222	35.459	1.3674
394.41	1294.0	-0.331	31.258	1.5049

Appendix G

Searles Lake Core KM-3 Isotopic Results for Bulk Leached Samples Run at University of Missouri-Columbia

```
*****************************************************************************
*                                                                           *
*                                APPENDIX G                                 *
*                                                                           *
*   SEARLES LAKE CORE KM-3: DEPTH, AGE, O-18 & C-13 RESULTS FOR BULK        *
*       LEACHED SAMPLES RUN AT UNIVERSITY OF MISSOURI-COLUMBIA.             *
*                                                                           *
*****************************************************************************
```

Note: Dates according to Jannik (1990) chronology

CORE DEPTH (ft)	CORE DEPTH (m)	C-13 DOL (PDB)	O-18 DOL (SMOW)	AGE (Ma)
96.6	29.44	2.38	29.59	.0269
124.9	38.07	4.06	34.99	.0335
125.7	38.31	4.04	31.161	.0356
126.3	38.50	2.52	33.34	.0372
126.9	38.68	3.93	38.92	.0388
127.7	38.92	2.964	36.033	.0409
128.7	39.29	2.83	34.22	.0440
130.1	39.65	2.51	28.35	.0471
130.4	39.75	3.6	37.65	.0480
132.3	40.33	1.75	33.21	.0530
134.4	40.97	3.40	34.44	.0585
136.4	41.57	2.63	35.71	.0637
137.5	41.91	4.05	34.61	.0659
139.0	42.37	1.52	34.15	.0672
140.5	42.82	2.79	38.084	.0685
141.8	43.22	3.77	35.93	.0696
142.8	43.53	2.82	38.97	.0705
481.0	146.61	3.34	31.97	.3930
481.0	146.61	3.36	31.36	.3930
492.0	149.96	1.88	29.21	.4148
493.9	150.54	2.14	32.24	.4228
502.1	153.04	2.204	28.63	.4301
546.1	166.45	2.56	30.94	.5982
547.4	166.85	2.00	30.14	.6004
547.7	166.94	1.364	30.88	.6009
548.5	167.18	1.62	31.95	.6022
548.5	167.18	1.36	32.11	.6022
549.5	167.49	1.63	30.76	.6039
550.5	167.79	1.518	29.558	.6056
551.7	168.16	3.36	30.15	.6076
552.6	168.43	2.54	29.08	.6091
553.8	168.80	2.85	29.29	.6111
555.0	169.16	1.76	30.61	.6132
556.0	169.47	2.88	28.56	.6149
557.1	169.80	3.71	35.12	.6168
558.2	170.14	1.07	27.13	.6187
559.2	170.44	1.34	21.52	.6205
560.1	170.72	3.32	30.79	.6220
561.0	170.99	1.13	30.81	.6236
562.2	171.36	2.31	33.41	.6257
562.2	171.36	2.58	33.14	.6257
563.2	171.66	1.04	28.27	.6274
564.3	172.00	1.56	28.19	.6293
564.3	172.00	1.475	28.924	.6293
565.3	172.30	2.67	30.98	.6310
567.0	172.82	1.73	29.70	.6340
568.7	173.34	2.90	31.64	.6369
569.9	173.71	7.19	41.13	.6390
571.2	174.10	1.88	28.45	.6412
572.5	174.50	4.94	38.40	.6435
574.9	175.23	4.51	36.31	.6476
575.4	175.38	3.19	30.94	.6485
576.4	175.69	4.58	36.02	.6502
577.6	176.05	3.68	33.66	.6523
578.6	176.36	4.19	27.37	.6540
578.6	176.36	4.46	37.68	.6540
580.7	177.00	3.95	30.20	.6577

CORE DEPTH (ft)	CORE DEPTH (m)	C-13 DOL (PDB)	O-18 DOL (SMOW)	AGE (Ma)
582.2	177.45	3.62	31.19	.6603
583.3	177.79	6.05	32.22	.6622
584.2	178.06	2.56	19.57	.6637
587.8	179.16	2.36	28.75	.6697
590.9	180.11	5.38	39.7	.6794
590.9	180.11	5.73	40.35	.6794
592.6	180.62	3.426	30.436	.6848
636.4	193.98	1.28	23.00	.8156
636.4	193.98	2.82	25.15	.8156
638.5	194.62	3.49	27.60	.8210
641.1	195.41	4.21	28.59	.8278
642.5	195.83	2.94	31.28	.8314
651.4	198.55	3.32	32.73	.8496
653.6	199.22	3.17	30.02	.8548
659.2	200.92	4.06	29.87	.8681
661.7	201.69	3.72	29.62	.8740
666.5	203.15	4.36	31.58	.8854
668.4	203.73	4.52	35.82	.8899
671.5	204.67	3.64	28.63	.8976
701.2	213.73	4.51	34.32	.9352
706.2	215.25	1.43	28.98	.9409
710.2	216.47	1.87	30.40	.9454
713.4	217.44	2.95	42.30	.9490
750.0	228.60	5.80	40.00	.9948
766.5	233.63	2.25	30.48	1.0116
772.2	235.37	5.17	35.01	1.0174
798.7	243.44	4.74	34.43	1.0444
1006.0	306.63	3.84	33.11	1.3312
1012.0	308.46	0.78	32.05	1.3350
1017.0	309.98	4.17	29.86	1.3403
1076.0	327.97	-0.95	30.24	1.4101
1080.4	329.31	-1.91	29.03	1.4244
1089.0	331.93	0.63	33.13	1.4366
1093.0	333.15	4.75	38.95	1.4500
1099.0	334.98	-5.14	27.06	1.4535

Appendix H

Analytical-Solution Lake Isotope Evolution Model and Results

```
C******************************************************************************
C*                                                                            *
C*    THIS PROGRAM COMPUTES FRED'S ANALYTICAL SOLUTIONS FOR CLOSED-BASIN     *
C*    LAKE VOLUMES AND ISOTOPIC COMPOSITION.  YOU WILL BE ASKED TO INPUT     *
C*    ALL PARAMETERS- PAY CAREFUL ATTENTION TO UNITS!                        *
C*                                                                            *
C******************************************************************************
        PROGRAM CBLAKES
        CHARACTER FNAME*50,ANS*1,ANS2*1
C
        WRITE(6,*)'THIS PROGRAM COMPUTES FREDS ANALYTICAL
     1 SOLUTIONS FOR'
        WRITE(6,*)'CLOSED-BASIN  LAKE VOLUMES AND ISOTOPIC COMPOSITION.'
        WRITE(6,*)'YOU WILL BE ASKED TO INPUT  ALL PARAMETERS- PAY
     1 CAREFUL'
        WRITE(6,*)' ATTENTION TO UNITS!'
C
C
C
        WRITE(6,*)'ENTER OUTPUT FILE NAME'
        READ(5,'(A)')FNAME
        OPEN(UNIT=95,FILE=FNAME,STATUS='NEW',CARRIAGECONTROL='LIST')

   10   WRITE(6,*)'INPUT TYPE OF EQUATIONS USED (R,I,E)'
        WRITE(6,*)'    R=    RAMP INFLOW EQUATIONS'
        WRITE(6,*)'    I=    INSTANTANEOUS INFLOW EQUATIONS'
        WRITE(6,*)'    E=    EXIT PROGRAM'
        READ(5,'(A)')ANS
        IF((ANS.EQ.'R').OR.(ANS.EQ.'r'))THEN
                WRITE(6,*)'STATE SITUATION (I,D,E)'
                WRITE(6,*)'   I=  INCREASING OR DECREASING INFLOW'
                WRITE(6,*)'   D=  DESSICATION:INITIAL AND FINAL INFLOW=0'
                WRITE(6,*)'   E=  EXIT RAMP EQUATIONS'
                READ(5,'(A)')ANS2
                IF((ANS2.EQ.'I').OR.(ANS2.EQ.'i'))THEN
                        CALL RAMP
                ELSEIF((ANS2.EQ.'D').OR.(ANS2.EQ.'d'))THEN
                        CALL DESSICATE
                ELSEIF((ANS2.EQ.'E').OR.(ANS2.EQ.'e'))THEN
                        GO TO 10
                ENDIF
        ELSEIF((ANS.EQ.'I').OR.(ANS.EQ.'i'))THEN
                WRITE(6,*)'STATE SITUATION (I,D,E)'
                WRITE(6,*)'   I=  INCREASING OR DECREASING INFLOW'
                WRITE(6,*)'   D=  DESSICATION:INITIAL AND FINAL INFLOW=0'
                WRITE(6,*)'   E=  EXIT INSTANTANEOUS INFLOW EQUATIONS'
                READ(5,'(A)')ANS2
                IF((ANS2.EQ.'I').OR.(ANS2.EQ.'i'))THEN
                        CALL INSTANT
                ELSEIF((ANS2.EQ.'D').OR.(ANS2.EQ.'d'))THEN
                        CALL DESSICATE
                ELSEIF((ANS2.EQ.'E').OR.(ANS2.EQ.'e'))THEN
                        GO TO 10
                ENDIF
        ELSEIF((ANS.EQ.'E').OR.(ANS.EQ.'e'))THEN
                GO TO 100
  100   ENDIF
        STOP
        END
C
C******************************************************************************
C
        SUBROUTINE RAMP
c       COMMON /STUFF/ TEMP,ELVS,EVAP,AREA,VOL,DK,H,DELO,VO,DELC,CAREPS
c       COMMON /RAMPSTUFF/ QIO,QF,LRAMP,BETA
c       COMMON /BOTH/ DELI
cC
c       CHARACTER TITLE*60,ANS3*1
c       DIMENSION RVOL(0:1000),DLAKE(0:1000),DELOF(0:1000)
c       DIMENSION CARB(0:1000)
```

```
C       COMING TO A THEATRE NEAR YOU...
        RETURN
        END
C
C*******************************************************************************
C*******************************************************************************
        SUBROUTINE INSTANT
        COMMON /STUFF/ TEMP,ELVS,EVAP,AREA,VOL,DK,H,DELO,VO,DELC,CAREPS
        COMMON /QINSTANT/ QI
        COMMON /BOTH/ DELI
C
        CHARACTER TITLE*60,ANS2*1
        DIMENSION RVOL(0:3000),DLAKE(0:3000),DELOF(0:3000)
        DIMENSION CARB(0:3000)
C
        WRITE(6,*)'DESCRIBE SITUATION (60 CHARACTERS OR LESS)'
        READ(5,'(A)')TITLE
        WRITE(95,*)'************************************************************'
        WRITE(95,*)
        WRITE(95,*)TITLE
        WRITE(95,*)
        WRITE(95,*)'************************************************************'

        CALL PARAMS
C
C
C       *****  DETERMINE IF OVERFLOW WILL OCCUR WITH INFLOW  ********
C
 12     FORMAT(1x,A5,7x,A6,7x,A12,4x,A8)
 13     FORMAT(1x,I5,3x,1e13.7,5x,f11.7,9x,f8.5)

        V=QI/((1-H)*DK)
        IF(VOL.EQ.VO)THEN
        KOFTIM=LOG((VOL-V)/(VO-V))/(-(1-H)*DK)
                WRITE(5,*)'LAKE WILL OVERFLOW'
        WRITE(95,*)
        WRITE(95,12)'TIME ','VOLUME','DEL O-18 H2O','   O-18 CARB'
        WRITE(95,12)'(yrs)',' (m^3)','    (SMOW)    ','     (SMOW)     '
        WRITE(95,*)
                WRITE(95,*)
                WRITE(95,*)'************ OVERFLOW *****************'
                WRITE(95,*)
C
C               COMPUTE ISOTOPIC COMPOSITION OF OVERFLOWING LAKE
C
                QO=QI-DK*VOL*(1-H)
                KTIM=0
                P1=DK/(ELVS+1) + QO/VOL
                P2=QI*DELI/VOL + DK*(ELVS/(ELVS+1) + H*DELC)
                DO 250 J=0,500,10
                        RVOL(J)=(VO-V)*EXP(-(1-H)*DK*KTIM)+V
                        DELOF(J)=(P2/P1 + (DELO-P2/P1)
     1                          *EXP((-P1)*KTIM))*1000.
                        CARB(J)=CAREPS + DELOF(J)
                        WRITE(95,13)KTIM,VOL,DELOF(J),CARB(J)
                        KTIM=KTIM+10
                        IF(RVOL(J).LT.VOL)THEN
                           DELBSW=DELOF(J)/1000.
                                KBSW=2000-KTIM+10
                                GO TO 251

                        ENDIF
 250            CONTINUE
C
C               COMPUTE LAKE VOLUME AND ISO COMP. LAKE BENEATH SPILLWAY
C
 251            WRITE(95,*)
                WRITE(95,*)'******** LAKE LEVEL BENEATH SPILLWAY *******'
                WRITE(95,*)
                KTIM=0
                        A=(ELVS+1)/DK
```

```
                  B=(ELVS+DELC*H*(ELVS+1))/(1-(ELVS+1)*(1-H))
                  D2=QI*(DELI*A+B*(VO-QI*A))
            DO 260 J=0,KBSW,100
                  RVOL(J)=(VO-V)*EXP(-(1-H)*DK*KTIM) + V
                  C=EXP(-(DK*KTIM/(ELVS+1)))
                  DLAKE(J)=((QI*A*(DELI-B)+(VO*(DELBSW-B)-QI*A
     1                  *(DELI-B))*C)/RVOL(J)+B)*1000.
                  CARB(J)=CAREPS + DLAKE(J)
                  WRITE(95,13)KTIM,RVOL(J),DLAKE(J),CARB(J)
                  KTIM=KTIM+100
                  IF(KTIM.GE.KBSW)THEN
                       RVOL(KBSW)=(VO-V)*EXP(-(1-H)*DK*KBSW) + V
                       C=EXP(-(DK*KBSW/(ELVS+1)))
                       DLAKE(KBSW)=((QI*A*(DELI-B)+(VO*(DELBSW-B)
     1                   -QI*A*(DELI-B))*C)/RVOL(KBSW)+B)*1000.
                       CARB(KBSW)=CAREPS+DLAKE(KBSW)
                       WRITE(95,13)KBSW,RVOL(KBSW),DLAKE(KBSW),
     1                    CARB(KBSW)
                  ENDIF
 260        CONTINUE
C
C           COMPUTE STEADY STATE DEL LAKE
C
            FV=QI/((1-H)*DK)
            SS1=QI*DELI*(ELVS+1)/(DK*FV) + ELVS + H*DELC*(ELVS+1)
            SS=SS1*1000.
            DOLSS=CAREPS + SS
            WRITE(95,*)'           STEADY STATE VALUES:NON-OVERFLOW'
            WRITE(95,*)
            WRITE(95,*)'                              LAKE VOLUME',FV
            WRITE(95,*)'                     STEADY STATE DEL H2O',SS
            WRITE(95,*)'                   STEADY STATE CARBONATE',DOLSS
        ELSEIF(V.GT.VOL)THEN
        KOFTIM=LOG((VOL-V)/(VO-V))/(-(1-H)*DK)
            WRITE(5,*)'LAKE WILL OVERFLOW'
C
C           COMPUTE LAKE VOLUME AND ISO COMPOSITION UNTIL OVERFLOW
C
            WRITE(95,*)
        WRITE(95,12)'TIME ','VOLUME','DEL O-18 H2O','  O-18 CARB'
        WRITE(95,12)'(yrs)',' (m^3)','   (SMOW)   ','    (SMOW)    '
            WRITE(95,*)
                  KTIM=0
                  A=(ELVS+1)/DK
                  B=(ELVS+DELC*H*(ELVS+1))/(1-(ELVS+1)*(1-H))
            DO 150 J=0,KOFTIM+1,10
                  RVOL(J)=(VO-V)*EXP(-(1-H)*DK*KTIM) + V
                  C=EXP(-(DK*KTIM/(ELVS+1)))
                  IF((VO.EQ.0).AND.(KTIM.EQ.0))THEN
                  DLAKE(0)=DELI*1000.
                  CARB(0)=CAREPS+DLAKE(0)
                  ELSE
                  DLAKE(J)=((QI*A*(DELI-B)+(VO*(DELO-B)-QI*A
     1                  *(DELI-B))*C)/RVOL(J)+B)*1000.
                  CARB(J)=CAREPS+DLAKE(J)
                  ENDIF
                  WRITE(95,13)KTIM,RVOL(J),DLAKE(J),CARB(J)
                  KTIM=KTIM+10
                  IF(KTIM.GE.KOFTIM+1)THEN
                       RVOL(KOFTIM+1)=(VO-V)*
     1                    EXP(-(1-H)*DK*(KOFTIM+1))+V
                       C=EXP(-(DK*(KOFTIM+1)/(ELVS+1)))
                       DLAKE(KOFTIM+1)=((QI*A*(DELI-B)+(VO*(DELO
     1                    -B)-QI*A*(DELI-B))*C)/RVOL(KOFTIM+1)+B)
     1                    *1000.
                       CARB(KOFTIM+1)=CAREPS+DLAKE(KOFTIM+1)
                       WRITE(95,13)KOFTIM,RVOL(KOFTIM+1),
     1                      DLAKE(KOFTIM+1),CARB(KOFTIM+1)
                       DELOV=(DLAKE(KOFTIM+1))/1000.
                  ENDIF
```

```
 150          CONTINUE
C
C             COMPUTE ISOTOPIC COMPOSITION OF OVERFLOWING LAKE
C
              WRITE(95,*)
              WRITE(95,*)' ***************  OVERFLOW  ********************'
              WRITE(95,*)
              QO=QI-DK*VOL*(1-H)
              KTIM=0
              P1=DK/(ELVS+1) + QO/VOL
              P2=QI*DELI/VOL + DK*(ELVS/(ELVS+1) + H*DELC)
              DO 160 J=0,500,50
                     DELOF(J)=(P2/P1 + (DELOV-P2/P1)
     1                      *EXP((-P1)*KTIM))*1000.
                     CARB(J)=CAREPS+DELOF(J)
                     WRITE(95,13)KTIM,VOL,DELOF(J),CARB(J)
                     KTIM=KTIM+50
 160          CONTINUE
C
C             COMPUTE STEADY STATE DEL LAKE-OVERFLOW
C
              SS1=QI*DELI + DK*VOL*((ELVS/(ELVS+1)) + H*DELC)
              SS2=QI + DK*VOL*(H-1+(1/(ELVS+1)))
              SSDELO=(SS1/SS2)*1000.
              DOL=CAREPS+SSDELO
              WRITE(95,*)'            STEADY STATE VALUES-OVERFLOW'
              WRITE(95,*)
              WRITE(95,*)'                   OVERFLOW TIME',KOFTIM
              WRITE(95,*)'                     LAKE VOLUME',VOL
              WRITE(95,*)'            STEADY STATE DEL H2O',SSDELO
              WRITE(95,*)'          STEADY STATE CARBONATE',DOL
              WRITE(95,*)'                    OUTFLOW RATE',QO
C
C       COMPUTE LAKE VOLUME FOR NON-OVERFLOWING LAKE'
C
        ELSE
        WRITE(95,*)
        WRITE(95,12)'TIME ','VOLUME','DEL O-18 H2O','  O-18 CARB'
        WRITE(95,12)'(yrs)',' (m^3)','   (SMOW)   ','   (SMOW)    '
        WRITE(95,*)
              KTIM=0
                     A=(ELVS+1)/DK
                     B=(ELVS+DELC*H*(ELVS+1))/(1-(ELVS+1)*(1-H))
              DO 170 J=0,2500,100
                     RVOL(J)=(VO-V)*EXP(-(1-H)*DK*KTIM) + V
                     C=EXP(-(DK*KTIM/(ELVS+1)))
                     IF((VO.EQ.0).AND.(KTIM.EQ.0))THEN
                     DLAKE(J)=DELI*1000.
                     CARB(J)=CAREPS+DLAKE(J)
                     ELSE
                     DLAKE(J)=((QI*A*(DELI-B) + (VO*(DELO-B)-
     1                  QI*A*(DELI-B))*C)/RVOL(J) + B)*1000.
                     CARB(J)=CAREPS+DLAKE(J)
                     ENDIF
                     WRITE(95,13)KTIM,RVOL(J),DLAKE(J),CARB(J)
                     KTIM=KTIM+100
                     IF(KTIM.GT.2500)THEN
                     ENDIF
 170          CONTINUE
C
C             COMPUTE STEADY STATE DEL LAKE
C

              FV=QI/((1-H)*DK)
              SS1=QI*DELI*(ELVS+1)/(DK*FV) + ELVS + H*DELC*(ELVS+1)
              SSDELO=SS1*1000.
              DOL=CAREPS+SSDELO
              WRITE(95,*)'        STEADY STATE VALUES:NON-OVERFLOW'
              WRITE(95,*)
```

```
            WRITE(95,*)'                               LAKE VOLUME',FV
            WRITE(95,*)'                     STEADY STATE DEL H2O',SSDELO
            WRITE(95,*)'                   STEADY STATE CARBONATE',DOL
        ENDIF
        RETURN
        END
C*******************************************************************************
C*******************************************************************************
        SUBROUTINE DESSICATE
        COMMON /STUFF/ TEMP,ELVS,EVAP,AREA,VOL,DK,H,DELO,VO,DELC,CAREPS
C
        CHARACTER TITLE*60
        DIMENSION RVOL(0:2100),DLAKE(0:2100),CARB(0:2100)
C
        WRITE(6,*)'DESCRIBE SITUATION (60 CHARACTERS OR LESS)'
        READ(5,'(A)')TITLE
        WRITE(95,*)'************************************************************'
        WRITE(95,*)
        WRITE(95,*)TITLE
        WRITE(95,*)
        WRITE(95,*)'************************************************************'
C
        CALL PARAMS
C
C               COMPUTE VOLUME AND ISO COMP OF DESSICATING LAKE
C
 12     FORMAT(1x,A5,7x,A6,7x,A12,4x,A8)
 13     FORMAT(1x,I5,3x,1e13.7,5x,f11.7,9x,f8.5)
        KTIM=0
        A=(ELVS+DELC*H*(ELVS+1))/(1-(ELVS+1)*(1-H))
        WRITE(95,*)
        WRITE(95,12)'TIME ','VOLUME','DEL O-18 H2O',' O-18 CARB'
        WRITE(95,12)'(YRS)',' (M^3)','   (SMOW)   ','   (SMOW)   '
        WRITE(95,*)
        DO 180 J=0,2000,100
                RVOL(J)=VO*EXP(-(1-H)*DK*KTIM)
                DLAKE(J)=((1/RVOL(J))*(VO*(DELO-A)*EXP(-DK*KTIM/(ELVS+1
     1                  )))+A)*1000.
                CARB(J)=CAREPS+DLAKE(J)
                WRITE(95,13)KTIM,RVOL(J),DLAKE(J),CARB(J)
                KTIM=KTIM+100
 180    CONTINUE
        RETURN
        END
C*******************************************************************************
C*******************************************************************************
C
        SUBROUTINE PARAMS
C
        COMMON /STUFF/ TEMP,ELVS,EVAP,AREA,VOL,DK,H,DELO,VO,DELC,CAREPS
        COMMON /RAMPSTUFF/ QIO,QF,LRAMP,BETA
        COMMON /QINSTANT/ QI
        COMMON /BOTH/ DELI
C
        CHARACTER ANS*1,ANS3*1,ANS4*1,ANS5*1
C
C
C       INPUT TEMPERATURE AND COMPUTE ELV ACCORDING TO BOTTINGA AND CRAIG(1969)
C
        WRITE(6,*)'ENTER TEMPERATURE IN DEGREES CELCIUS'
        READ(5,*)TEMP
        write(95,*)'TEMPERATURE IN DEGREES CELCIUS IS',TEMP
        TEMP=TEMP + 273.4
        ELV=1.534*(1e6/TEMP**2)-3.206*(1e3/TEMP)+2.644
        WRITE(95,*)'ELV =',ELV
        WRITE(6,*)'IS THIS RUN ASSUMING DELC=DELI?(Y,N)'
        READ(5,'(A)')ANS5
                IF((ANS5.EQ.'Y').OR.(ANS5.EQ.'y'))THEN
        WRITE(6,*)'ENTER ISOTOPIC COMP. OF BACK CONDENS. FLUX(PER MIL)'
        READ(5,*)DELC
                ELSEIF((ANS5.EQ.'N').OR.(ANS5.EQ.'n'))THEN
```

```
      WRITE(6,*)'ENTER ISOTOPIC COMPOSITION OF ATMOSPHERE'
      READ(5,*)DELA
      DELC=ELV*(1+(DELA/1000))+DELA
             ENDIF
      WRITE(95,*)'ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS ',DELC
      DELC=DELC/1000.
      WRITE(6,*)'ENTER RELATIVE HUMIDITY'
      READ(5,*)H
      WRITE(95,*)'RELATIVE HUMIDITY IS',H
      ALPHA=EXP(ELV/1000)
      ELVS=((1.0/ALPHA)*(1.0 - 0.0069)/(1.0-0.0069*H))-1
      ELVS=ELVS*(-1)
      WRITE(95,*)'ELV-STAR =',ELVS
C
      WRITE(6,*)' COMPUTE CARBONATE VALUES ALONG WITH WATER? (Y,N)'
      READ(5,'(A)')ANS3
      IF((ANS3.EQ.'Y').OR.(ANS3.EQ.'y'))THEN
              WRITE(6,*)'CALCITE OR DOLOMITE? (C,D)'
              READ(5,'(A)')ANS4
              IF((ANS4.EQ.'D').OR.(ANS4.EQ.'d'))THEN
              WRITE(95,*)'COMPUTED CARBONATE VALUES FOR DOLOMITE'
C
C             COMPUTE DOLOMITE ELV ACCORDING TO NORTHROP & CLAYTON (1966)
C
                      CAREPS=3.20*(1.E6)/(TEMP**2) - 1.50
C
              ELSEIF((ANS4.EQ.'C').OR.(ANS4.EQ.'c'))THEN
              WRITE(95,*)'COMPUTED CARBONATE VALUES FOR CALCITE'
C
C     COMPUTE CALCITE EPSILON ACCORDING TO O'NEIL,CLAYTON & MAYEDA(1969)
C
                      CAREPS=2.78*(1.E6)/(TEMP**2)-2.89
              ENDIF
      ELSEIF((ANS3.EQ.'N').OR.(ANS3.EQ.'n'))THEN
      ENDIF
C
C             INPUT OTHER INITIAL PARAMETERS
C
      WRITE(6,*)'ENTER EVAPORATION RATE (m/yr)'
      read(5,*)EVAP
      WRITE(95,*)'EVAPORATION RATE IS (m/yr) ',EVAP
C
C             INPUT LAKE GEOMETRY
C
      WRITE(6,*)'ENTER MAX. LAKE SURFACE AREA(EST. HIGH LAKE STAND)(m^2)'
      read(5,*)AREA
      WRITE(95,*)'ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2)',AREA
      WRITE(6,*)'ENTER MAX. LAKE VOLUME (EST. HIGH LAKE STAND)(m^3)'
      READ(5,*)VOL
      WRITE(95,*)'ESTIMATED MAXIMUM LAKE VOLUME IS (m^3)',VOL
C
C             COMPUTE K
C
      DK=(AREA/VOL)*EVAP
      WRITE(95,*)'K= ',DK
C
C             ENTER YET MORE INITIAL PARAMETERS
C
      WRITE(6,*)'ENTER INITIAL LAKE ISO COMPOSITION (PER MIL)'
      READ(5,*)DELO
      WRITE(95,*)'INITIAL LAKE ISO COMPOSITION IS(PER MIL)',DELO
      DELO=DELO/1000.
      WRITE(6,*)'ENTER INITIAL LAKE VOLUME(FLUID IN LAKE)(m^3)'
      READ(5,*)VO
      WRITE(95,*)'INITIAL LAKE VOLUME (m^3) IS',VO
C
C
C
      WRITE(6,*)'INPUT TYPE EQUATIONS USED ( R,I,D )'
      WRITE(6,*)'      R =  RAMP EQUATIONS'
```

```
      WRITE(6,*)'      I =  INSTANTANEOUS INFLOW'
      WRITE(6,*)'      D =  DESSICATION:INITIAL & FINAL INFLOW=0'
      READ(5,'(A)')ANS
      IF((ANS.EQ.'R').OR.(ANS.EQ.'r'))THEN
              WRITE(6,*)'ENTER INITIAL INFLOW RATE (m^3/yr)'
              read(5,*)QIO
              WRITE(95,*)'INITIAL INFLOW RATE IS (m^3/yr)',QIO
              WRITE(6,*)'ENTER FINAL INFLOW RATE (m^3/yr)'
              READ(5,*)QF
              WRITE(95,*)'FINAL INFLOW RATE IS (m^3/yr)',QF
              WRITE(6,*)'ENTER TIME RAMP (YEARS)'
              READ(5,*)LRAMP
              WRITE(95,*)'TIME RAMP IN YEARS IS',LRAMP
C
C             COMPUTE BETA
C
              BETA=(QF-QIO)/LRAMP
              WRITE(95,*)'BETA =',BETA
              WRITE(6,*)'INPUT ISOTOPIC COMP.
     6                        OF INFLOW WATER(PER MIL)'
              READ(5,*)DELI
              WRITE(95,*)'ISOTOPIC COMP OF INFLOW WATER IS',DELI
              DELI=DELI/1000.
      ELSEIF((ANS.EQ.'I').OR.(ANS.EQ.'i'))THEN
              WRITE(6,*)'ENTER  INFLOW RATE (m^3/yr)'
              read(5,*)QI
              WRITE(95,*)' INFLOW RATE IS (m^3/yr)',QI
              WRITE(6,*)'INPUT ISOTOPIC COMP.OF INFLOW WATER(PER MIL)'
              READ(5,*)DELI
              WRITE(95,*)'ISOTOPIC COMP OF INFLOW WATER IS',DELI
              DELI=DELI/1000.
      ELSEIF((ANS.EQ.'D').OR.(ANS.EQ.'d'))THEN
      ENDIF
      RETURN
      END
C*****************************************************************************
```

****

INFILL OWENS FROM DRY LAKE; DELI=DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 15.50000
ELV = 9.926092
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -16.25000
RELATIVE HUMIDITY IS 0.2780000
ELV-STAR = 1.4819145E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.177000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 6.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 3.0020000E+10
K= 2.7209796E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) -16.25000
INITIAL LAKE VOLUME (m^3) IS 0.0000000E+00
INFLOW RATE IS (m^3/yr) 3.7000000E+09
ISOTOPIC COMP OF INFLOW WATER IS -16.25000

TIME (yrs)	VOLUME (m^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
0	0.0000000E+00	-16.2500000	20.59025
8	0.3052221E+11	-14.5784540	22.26180

*************** OVERFLOW *******************

0	0.3002000E+11	-14.5784540	22.26180
50	0.3002000E+11	-13.2548151	23.58543
100	0.3002000E+11	-13.2528658	23.58738
150	0.3002000E+11	-13.2528629	23.58739
200	0.3002000E+11	-13.2528629	23.58739
250	0.3002000E+11	-13.2528629	23.58739
300	0.3002000E+11	-13.2528629	23.58739
350	0.3002000E+11	-13.2528629	23.58739
400	0.3002000E+11	-13.2528629	23.58739
450	0.3002000E+11	-13.2528629	23.58739
500	0.3002000E+11	-13.2528629	23.58739

STEADY STATE VALUES-OVERFLOW

OVERFLOW TIME	8
LAKE VOLUME	3.0020000E+10
STEADY STATE DEL H2O	-13.25286
STEADY STATE CARBONATE	23.58739
OUTFLOW RATE	3.1102428E+09

```
***********************************************************

INFILL CHINA LAKE FROM DRY LAKE; DELI=DELC

***********************************************************
TEMPERATURE IN DEGREES CELCIUS IS   18.40000
ELV =   9.672875
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS    -13.25286
RELATIVE HUMIDITY IS  0.2320000
ELV-STAR =  1.4882922E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr)     1.413000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2)  1.5500000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3)  6.9600000E+08
K=   0.3146767
INITIAL LAKE ISO COMPOSITION IS(PER MIL)  -13.25286
INITIAL LAKE VOLUME (m^3) IS  0.0000000E+00
INFLOW RATE IS (m^3/yr)  3.1100001E+09
ISOTOPIC COMP OF INFLOW WATER IS  -13.25286

TIME         VOLUME          DEL O-18 H2O          O-18 CARB
(yrs)         (m^3)             (SMOW)               (SMOW)

    0    0.0000000E+00      -13.2528601          22.82910
    0    0.2762729E+10      -11.1157265          24.96623

 ***************  OVERFLOW  ********************

    0    0.6960000E+09      -11.1157265          24.96623
   50    0.6960000E+09      -12.2491808          23.83278
  100    0.6960000E+09      -12.2491808          23.83278
  150    0.6960000E+09      -12.2491808          23.83278
  200    0.6960000E+09      -12.2491808          23.83278
  250    0.6960000E+09      -12.2491808          23.83278
  300    0.6960000E+09      -12.2491808          23.83278
  350    0.6960000E+09      -12.2491808          23.83278
  400    0.6960000E+09      -12.2491808          23.83278
  450    0.6960000E+09      -12.2491808          23.83278
  500    0.6960000E+09      -12.2491808          23.83278
              STEADY STATE VALUES-OVERFLOW

                     OVERFLOW TIME           0
                       LAKE VOLUME   6.9600000E+08
              STEADY STATE DEL H2O   -12.24918
            STEADY STATE CARBONATE    23.83278
                      OUTFLOW RATE   2.9417966E+09
```

****

INFILL SEARLES LAKE FROM DRY LAKE; DELI=DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 19.10000
ELV = 9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -12.24918
RELATIVE HUMIDITY IS 0.2210000
ELV-STAR = 1.4898777E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 8.5279998E+10
K= 1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) -12.24918
INITIAL LAKE VOLUME (m^3) IS 0.0000000E+00
INFLOW RATE IS (m^3/yr) 2.9417999E+09
ISOTOPIC COMP OF INFLOW WATER IS -12.24918

TIME (yrs)	VOLUME (m^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
0	0.0000000E+00	-12.2491798	23.65311
10	0.2727811E+11	-10.8798704	25.02242
20	0.5068638E+11	-9.6160879	26.28621
30	0.7077383E+11	-8.4522257	27.45007
38	0.8640402E+11	-7.4856467	28.41665
*************** OVERFLOW *******************			
0	0.8528000E+11	-7.4856467	28.41665
50	0.8528000E+11	-5.2432985	30.65900
100	0.8528000E+11	-4.9169765	30.98532
150	0.8528000E+11	-4.8694882	31.03281
200	0.8528000E+11	-4.8625770	31.03972
250	0.8528000E+11	-4.8615713	31.04072
300	0.8528000E+11	-4.8614249	31.04087
350	0.8528000E+11	-4.8614039	31.04089
400	0.8528000E+11	-4.8614006	31.04089
450	0.8528000E+11	-4.8614001	31.04089
500	0.8528000E+11	-4.8614001	31.04089

STEADY STATE VALUES-OVERFLOW

OVERFLOW TIME	38
LAKE VOLUME	8.5279998E+10
STEADY STATE DEL H2O	-4.861401
STEADY STATE CARBONATE	31.04089
OUTFLOW RATE	1.6370606E+09

****

SEARLES OVERFLOW TO BENEATH SPILLWAY; DELI=DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 19.10000
ELV = 9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -12.24918
RELATIVE HUMIDITY IS 0.2210000
ELV-STAR = 1.4898777E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 8.5279998E+10
K= 1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) -4.861401
INITIAL LAKE VOLUME (m^3) IS 8.5279998E+10
INFLOW RATE IS (m^3/yr) 1.0000000E+09
ISOTOPIC COMP OF INFLOW WATER IS -12.24918

TIME (yrs)	VOLUME (m^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
*********** OVERFLOW ****************			
0	0.8528000E+11	-4.8614011	31.04089
10	0.8528000E+11	-3.3052216	32.59707
******** LAKE LEVEL BENEATH SPILLWAY *******			
0	0.8528000E+11	-3.3052227	32.59707
100	0.6967496E+11	2.5928617	38.49516
200	0.6629573E+11	2.7469881	38.64928
300	0.6556398E+11	2.5650747	38.46737
400	0.6540551E+11	2.4936721	38.39597
500	0.6537120E+11	2.4736226	38.37592
600	0.6536377E+11	2.4686158	38.37091
700	0.6536216E+11	2.4674423	38.36974
800	0.6536181E+11	2.4671741	38.36947
900	0.6536174E+11	2.4671144	38.36941
1000	0.6536172E+11	2.4670997	38.36939
1100	0.6536172E+11	2.4670959	38.36939
1200	0.6536172E+11	2.4670959	38.36939
1300	0.6536172E+11	2.4670959	38.36939
1400	0.6536172E+11	2.4670959	38.36939
1490	0.6536172E+11	2.4670959	38.36939

STEADY STATE VALUES:NON-OVERFLOW

LAKE VOLUME 6.5361715E+10
STEADY STATE DEL H2O 2.467098
STEADY STATE CARBONATE 38.36939

****

SEARLES HIGH LAKE LEVEL TO LOW LEVEL; DELI=DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 19.10000
ELV = 9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -12.24918
RELATIVE HUMIDITY IS 0.2210000
ELV-STAR = 1.4898777E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 8.5279998E+10
K= 1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) 2.467098
INITIAL LAKE VOLUME (m^3) IS 6.5361719E+10
INFLOW RATE IS (m^3/yr) 3.0000000E+08
ISOTOPIC COMP OF INFLOW WATER IS -12.24918

TIME (yrs)	VOLUME (m^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
0	0.6536172E+11	2.4670959	38.36939
100	0.2951623E+11	8.6809998	44.58329
200	0.2175400E+11	5.5103006	41.41259
300	0.2007311E+11	3.3718011	39.27409
400	0.1970912E+11	2.6946478	38.59694
500	0.1963030E+11	2.5206320	38.42293
600	0.1961323E+11	2.4792888	38.38158
700	0.1960954E+11	2.4698229	38.37212
800	0.1960874E+11	2.4676993	38.37000
900	0.1960856E+11	2.4672225	38.36952
1000	0.1960852E+11	2.4671257	38.36942
1100	0.1960852E+11	2.4671032	38.36940
1200	0.1960851E+11	2.4670959	38.36939
1300	0.1960851E+11	2.4670959	38.36939
1400	0.1960851E+11	2.4670959	38.36939
1500	0.1960851E+11	2.4670959	38.36939
1600	0.1960851E+11	2.4670959	38.36939
1700	0.1960851E+11	2.4670959	38.36939
1800	0.1960851E+11	2.4670959	38.36939
1900	0.1960851E+11	2.4670959	38.36939
2000	0.1960851E+11	2.4670959	38.36939

STEADY STATE VALUES:NON-OVERFLOW

LAKE VOLUME 1.9608515E+10
STEADY STATE DEL H2O 2.467099
STEADY STATE CARBONATE 38.36939

****

DESSICATE SEARLES LAKE; DELI=DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 19.10000
ELV = 9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -12.24918
RELATIVE HUMIDITY IS 0.2950000
ELV-STAR = 1.4394820E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 8.5279998E+10
K= 1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) 2.467098
INITIAL LAKE VOLUME (m^3) IS 1.9608515E+10

TIME (YRS)	VOLUME (M^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
0	0.1960851E+11	2.4670997	38.36939
50	0.9812506E+10	10.9500923	46.85239
100	0.4910380E+10	17.3886890	53.29099
150	0.2457255E+10	22.2755814	58.17788
200	0.1229661E+10	25.9847355	61.88703
250	0.6153478E+09	28.7999821	64.70228
300	0.3079328E+09	30.9367580	66.83905
350	0.1540960E+09	32.5585709	68.46087
400	0.7711277E+08	33.7895241	69.69182
450	0.3858883E+08	34.7238235	70.62612
500	0.1931064E+08	35.4329491	71.33524
550	0.9663446E+07	35.9711800	71.87347
600	0.4835787E+07	36.3796997	72.28200
650	0.2419927E+07	36.6897621	72.59206
700	0.1210982E+07	36.9250984	72.82739
750	0.6060003E+06	37.1037254	73.00602
800	0.3032550E+06	37.2392998	73.14159
850	0.1517550E+06	37.3421974	73.24449
900	0.7594140E+05	37.4202995	73.32259
950	0.3800263E+05	37.4795799	73.38187
1000	0.1901730E+05	37.5245743	73.42687

```
**********************************************************

INFILL OWENS FROM DRY LAKE; DELI > DELC

**********************************************************
TEMPERATURE IN DEGREES CELCIUS IS   15.50000
ELV =   9.926092
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS   -21.68460
RELATIVE HUMIDITY IS  0.2780000
ELV-STAR =  1.4819145E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr)    1.177000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2)  6.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3)  3.0020000E+10
K=   2.7209796E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL)  -16.25000
INITIAL LAKE VOLUME (m^3) IS  0.0000000E+00
INFLOW RATE IS (m^3/yr)  3.7000000E+09
ISOTOPIC COMP OF INFLOW WATER IS  -16.25000

TIME         VOLUME        DEL O-18 H2O        O-18 CARB
(yrs)        (m^3)          (SMOW)              (SMOW)

    0   0.0000000E+00     -16.2500000          20.59025
    8   0.3052221E+11     -14.7542467          22.08600

 ***************  OVERFLOW  *******************

    0   0.3002000E+11     -14.7542467          22.08600
   50   0.3002000E+11     -13.5698185          23.27043
  100   0.3002000E+11     -13.5680752          23.27217
  150   0.3002000E+11     -13.5680723          23.27218
  200   0.3002000E+11     -13.5680723          23.27218
  250   0.3002000E+11     -13.5680723          23.27218
  300   0.3002000E+11     -13.5680723          23.27218
  350   0.3002000E+11     -13.5680723          23.27218
  400   0.3002000E+11     -13.5680723          23.27218
  450   0.3002000E+11     -13.5680723          23.27218
  500   0.3002000E+11     -13.5680723          23.27218
              STEADY STATE VALUES-OVERFLOW

                   OVERFLOW TIME          8
                     LAKE VOLUME    3.0020000E+10
              STEADY STATE DEL H2O  -13.56807
            STEADY STATE CARBONATE   23.27218
                    OUTFLOW RATE    3.1102428E+09
```

```
*********************************************************

INFILL CHINA LAKE FROM DRY LAKE; DELI > DELC

*********************************************************
TEMPERATURE IN DEGREES CELCIUS IS   18.40000
ELV =    9.672875
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS   -13.75154
RELATIVE HUMIDITY IS  0.2320000
ELV-STAR =  1.4882922E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr)    1.413000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2)  1.5500000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3)  6.9600000E+08
K=   0.3146767
INITIAL LAKE ISO COMPOSITION IS(PER MIL)  -13.56807
INITIAL LAKE VOLUME (m^3) IS  0.0000000E+00
INFLOW RATE IS (m^3/yr)  3.1100001E+09
ISOTOPIC COMP OF INFLOW WATER IS  -13.56807

TIME        VOLUME          DEL O-18 H2O          O-18 CARB
(yrs)        (m^3)            (SMOW)               (SMOW)

   0    0.0000000E+00      -13.5680704           22.51389
   0    0.2762729E+10      -11.4379120           24.64405

 ***************  OVERFLOW  *******************

   0    0.6960000E+09      -11.4379120           24.64405
  50    0.6960000E+09      -12.5676632           23.51430
 100    0.6960000E+09      -12.5676632           23.51430
 150    0.6960000E+09      -12.5676632           23.51430
 200    0.6960000E+09      -12.5676632           23.51430
 250    0.6960000E+09      -12.5676632           23.51430
 300    0.6960000E+09      -12.5676632           23.51430
 350    0.6960000E+09      -12.5676632           23.51430
 400    0.6960000E+09      -12.5676632           23.51430
 450    0.6960000E+09      -12.5676632           23.51430
 500    0.6960000E+09      -12.5676632           23.51430
              STEADY STATE VALUES-OVERFLOW

                     OVERFLOW TIME         0
                       LAKE VOLUME    6.9600000E+08
              STEADY STATE DEL H2O    -12.56766
            STEADY STATE CARBONATE     23.51430
                      OUTFLOW RATE    2.9417966E+09
```

****

INFILL SEARLES LAKE FROM DRY LAKE; DELI > DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 19.10000
ELV = 9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -13.80998
RELATIVE HUMIDITY IS 0.2210000
ELV-STAR = 1.4898777E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 8.5279998E+10
K= 1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) -12.56766
INITIAL LAKE VOLUME (m^3) IS 0.0000000E+00
INFLOW RATE IS (m^3/yr) 2.9417966E+09
ISOTOPIC COMP OF INFLOW WATER IS -12.56766

TIME (yrs)	VOLUME (m^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
0	0.0000000E+00	-12.5676603	23.33463
10	0.2727808E+11	-11.2246838	24.67761
20	0.5068633E+11	-9.9852495	25.91705
30	0.7077375E+11	-8.8438129	27.05848
38	0.8640392E+11	-7.8958535	28.00644

*************** OVERFLOW *******************

0	0.8528000E+11	-7.8958535	28.00644
50	0.8528000E+11	-5.6966844	30.20561
100	0.8528000E+11	-5.3766456	30.52565
150	0.8528000E+11	-5.3300710	30.57222
200	0.8528000E+11	-5.3232932	30.57900
250	0.8528000E+11	-5.3223071	30.57999
300	0.8528000E+11	-5.3221631	30.58013
350	0.8528000E+11	-5.3221426	30.58015
400	0.8528000E+11	-5.3221393	30.58015
450	0.8528000E+11	-5.3221388	30.58016
500	0.8528000E+11	-5.3221388	30.58016

STEADY STATE VALUES-OVERFLOW

OVERFLOW TIME	38
LAKE VOLUME	8.5279998E+10
STEADY STATE DEL H2O	-5.322138
STEADY STATE CARBONATE	30.58016
OUTFLOW RATE	1.6370573E+09

```
*********************************************************

SEARLES OVERFLOW TO BENEATH SPILLWAY; DELI > DELC

*********************************************************
TEMPERATURE IN DEGREES CELCIUS IS   19.10000
ELV =    9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS   -13.80998
RELATIVE HUMIDITY IS  0.2210000
ELV-STAR =  1.4898777E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr)    1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2)  9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3)  8.5279998E+10
K=   1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL)  -5.322138
INITIAL LAKE VOLUME (m^3) IS  8.5279998E+10
INFLOW RATE IS (m^3/yr)  1.0000000E+09
ISOTOPIC COMP OF INFLOW WATER IS  -12.56766

TIME          VOLUME          DEL O-18 H2O          O-18 CARB
(yrs)          (m^3)            (SMOW)               (SMOW)


*********** OVERFLOW ****************

    0    0.8528000E+11       -5.3221383          30.58016
   10    0.8528000E+11       -3.7959268          32.10637

******** LAKE LEVEL BENEATH SPILLWAY *******

    0    0.8528000E+11       -3.7959256          32.10637
  100    0.6967496E+11        1.9885786          37.89087
  200    0.6629573E+11        2.1397359          38.04203
  300    0.6556398E+11        1.9613206          37.86361
  400    0.6540551E+11        1.8913001          37.79359
  500    0.6537120E+11        1.8716305          37.77393
  600    0.6536377E+11        1.8667244          37.76902
  700    0.6536216E+11        1.8655733          37.76787
  800    0.6536181E+11        1.8653088          37.76760
  900    0.6536174E+11        1.8652492          37.76754
 1000    0.6536172E+11        1.8652343          37.76753
 1100    0.6536172E+11        1.8652306          37.76752
 1200    0.6536172E+11        1.8652306          37.76752
 1300    0.6536172E+11        1.8652306          37.76752
 1400    0.6536172E+11        1.8652306          37.76752
 1490    0.6536172E+11        1.8652306          37.76752
           STEADY STATE VALUES:NON-OVERFLOW

                         LAKE VOLUME  6.5361715E+10
                 STEADY STATE DEL H2O  1.865230
               STEADY STATE CARBONATE  37.76752
```

****

SEARLES HIGH LAKE LEVEL TO LOW LEVEL; DELI > DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 19.10000
ELV = 9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -13.80998
RELATIVE HUMIDITY IS 0.2210000
ELV-STAR = 1.4898777E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 8.5279998E+10
K= 1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) 1.865231
INITIAL LAKE VOLUME (m^3) IS 6.5361715E+10
INFLOW RATE IS (m^3/yr) 3.0000000E+08
ISOTOPIC COMP OF INFLOW WATER IS -12.56766

TIME (yrs)	VOLUME (m^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
0	0.6536172E+11	1.8652267	37.76752
100	0.2951623E+11	7.9594703	43.86176
200	0.2175400E+11	4.8498325	40.75213
300	0.2007311E+11	2.7525201	38.65482
400	0.1970912E+11	2.0884016	37.99070
500	0.1963030E+11	1.9177384	37.82003
600	0.1961323E+11	1.8771924	37.77949
700	0.1960954E+11	1.8679053	37.77020
800	0.1960874E+11	1.8658266	37.76812
900	0.1960856E+11	1.8653609	37.76765
1000	0.1960852E+11	1.8652604	37.76756
1100	0.1960852E+11	1.8652380	37.76753
1200	0.1960851E+11	1.8652343	37.76753
1300	0.1960851E+11	1.8652343	37.76753
1400	0.1960851E+11	1.8652343	37.76753
1500	0.1960851E+11	1.8652343	37.76753
1600	0.1960851E+11	1.8652343	37.76753
1700	0.1960851E+11	1.8652343	37.76753
1800	0.1960851E+11	1.8652343	37.76753
1900	0.1960851E+11	1.8652343	37.76753
2000	0.1960851E+11	1.8652343	37.76753

STEADY STATE VALUES:NON-OVERFLOW

LAKE VOLUME 1.9608515E+10
STEADY STATE DEL H2O 1.865231
STEADY STATE CARBONATE 37.76752

****

DESSICATE SEARLES LAKE; DELI > DELC

****

TEMPERATURE IN DEGREES CELCIUS IS 19.10000
ELV = 9.613042
ISOTOPIC COMP. OF BACK CONDENSATION FLUX IS -13.80998
RELATIVE HUMIDITY IS 0.2950000
ELV-STAR = 1.4394820E-02
COMPUTED CARBONATE VALUES FOR DOLOMITE
EVAPORATION RATE IS (m/yr) 1.685000
ESTIMATED MAXIMUM LAKE SURFACE AREA IS(m^2) 9.9400000E+08
ESTIMATED MAXIMUM LAKE VOLUME IS (m^3) 8.5279998E+10
K= 1.9639893E-02
INITIAL LAKE ISO COMPOSITION IS(PER MIL) 1.865231
INITIAL LAKE VOLUME (m^3) IS 1.9600001E+10

TIME (YRS)	VOLUME (M^3)	DEL O-18 H2O (SMOW)	O-18 CARB (SMOW)
0	0.1960000E+11	1.8652306	37.76752
50	0.9808245E+10	10.0981112	46.00040
100	0.4908248E+10	16.3468742	52.24917
150	0.2456189E+10	21.0896816	56.99198
200	0.1229128E+10	24.6894760	60.59177
250	0.6150807E+09	27.4217167	63.32401
300	0.3077991E+09	29.4954910	65.39779
350	0.1540291E+09	31.0694866	66.97178
400	0.7707929E+08	32.2641487	68.16644
450	0.3857208E+08	33.1708984	69.07320
500	0.1930226E+08	33.8591194	69.76141
550	0.9659251E+07	34.3814774	70.28378
600	0.4833687E+07	34.7779503	70.68024
650	0.2418876E+07	35.0788765	70.98117
700	0.1210457E+07	35.3072739	71.20957
750	0.6057372E+06	35.4806290	71.38293
800	0.3031233E+06	35.6122055	71.51450
850	0.1516891E+06	35.7120743	71.61437
900	0.7590842E+05	35.7878723	71.69017
950	0.3798613E+05	35.8454056	71.74770
1000	0.1900904E+05	35.8890686	71.79137

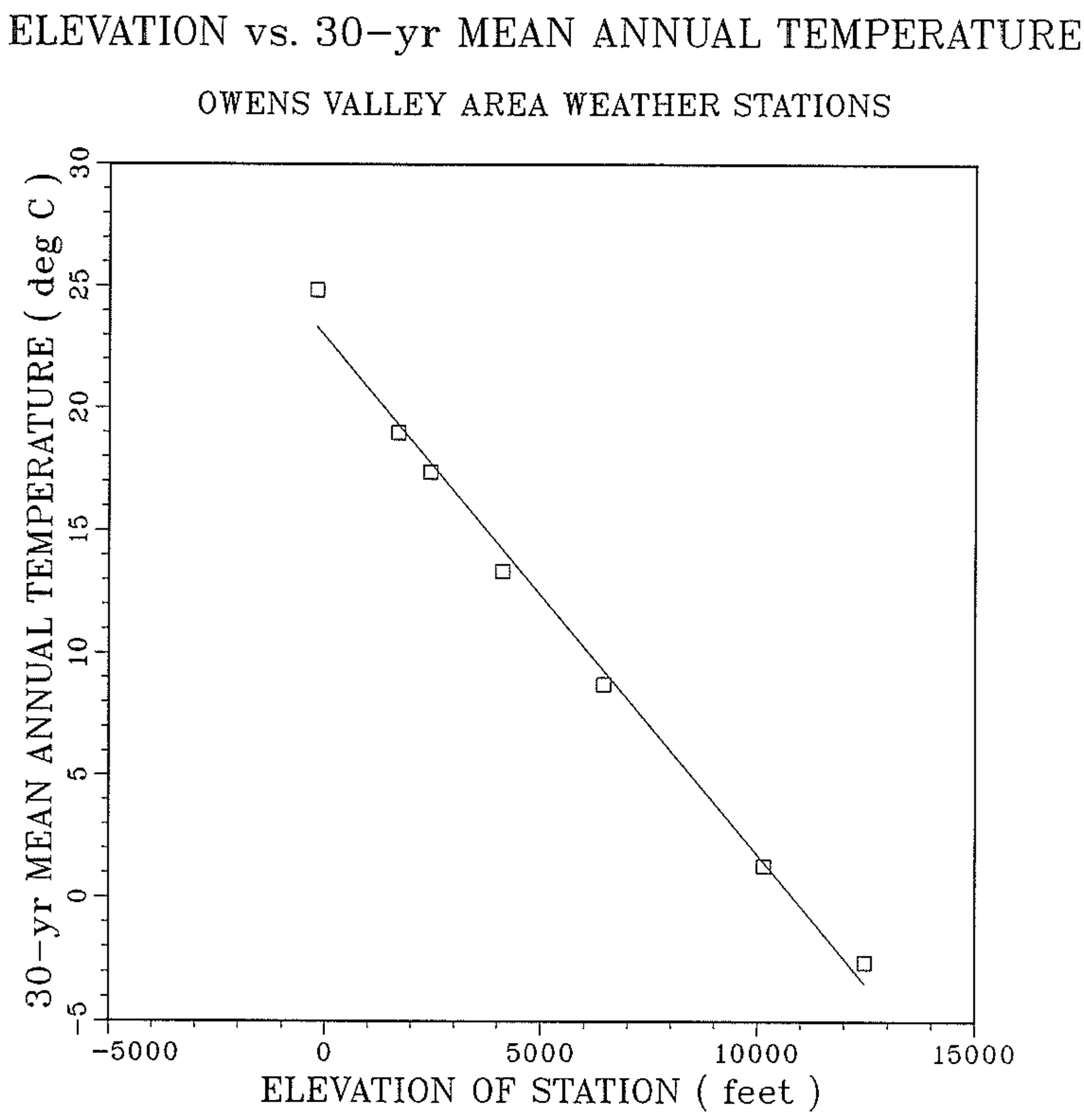
Appendix I

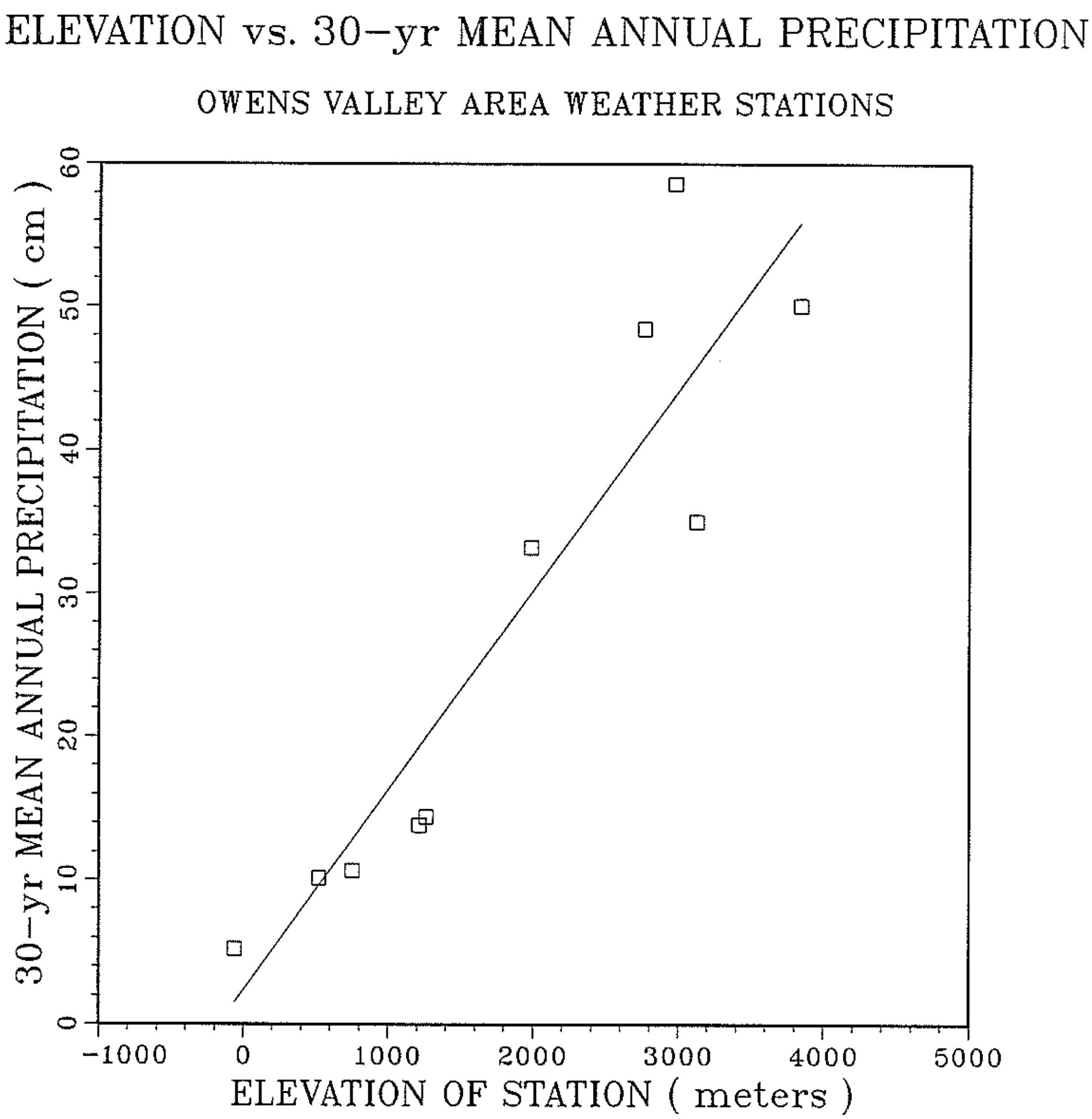
Climatological Data for Southeastern California Weather Stations

```
******************************************************************************
*                                                                            *
*   This file contains climatological data for SE California weather         *
*   stations east of the Sierra Nevada divide.                               *
*                                                                            *
*   Data from White Mountain stations are from info compiled by UC-Davis     *
*   for the BARCROFT(white Mountain2)  1953-1973 and CROOKED CREEK           *
*   (white mountain1) 1949-1973 reseach stations.                            *
*                                                                            *
*   Relative humidity data for Bishop is from:                               *
*   Ruffner,J.A. (ed.?),1985,CLimates of the States:Gale Research Co.        *
*       Detroit, Michigan.  Vol. 1                                           *
*   (compilation from NOAA climatological data QE 983 .C56 1985 v.1)         *
*                                                                            *
*   All other elevation,temp and precip data are from "Monthly normals of    *
*   Temperature, Precipitation, and Heating and Cooling Degree Days          *
*   1951-1980 no.81 (California):NOAA,National Climatic Center, Asheville,   *
*   North Carolina,1982."                                                    *
*                                                                            *
*   Evaporation data (following Smith,G.I. and Street-Perrott,F.A.) are      *
*   interpolations between "evaporation from lake surfaces-Plate 3"          *
*   IN Meyers,J.S.,1962, Evaporation from the 17 Western States, USGS        *
*   Professional Paper 272-D: USGS etc....                                   *
*                                                                            *
*                                                                            *
******************************************************************************
```

STATION NAME /LOCATION	ELEVATION (METERS)	TEMP DEG C 30*YR MEAN	ANNUAL PRECIP 30yr MEAN (cm)	REL HUMIDITY 30yr av annual
DEATH VALLEY	-59.69	24.83	5.16	----
TRONA	521.54	19.0	10.03	----
INYOKERN	750.77	17.39	10.57	----
INDEPENDENCE	1215.39	-----	13.69	----
BISHOP	1264.0	13.33	----	0.29
MONO LAKE	1984.62	8.72	----	----
WHITE MOUNTAIN1	3123.08	1.28	34.70	0.529
WHITE MOUNTAIN2	3836.92	-2.44	49.61	0.558

LOCATION	ELEVATION (METERS)	EVAPORATION FROM LAKE SURFACES ANNUAL GROSS (m/yr)
LAKE MANLY	-86	2.13
LAKE PANAMINT	317	1.93
SEARLES LAKE	493	1.78
CHINA LAKE	657	1.52
OWENS LAKE	1081	1.33
HAIWEE RESEV	1177	1.025
INDEPENDENCE	1215.39	1.025
BISHOP	1264.0	0.77


ELEVATION vs. 30-yr MEAN ANNUAL TEMPERATURE
OWENS VALLEY AREA WEATHER STATIONS
30-yr MEAN ANNUAL TEMPERATURE (deg C)
-5
0
5
10
15
20
25
30
-5000
0
5000
10000
15000
ELEVATION OF STATION (feet)



ELEVATION vs. 30-yr MEAN ANNUAL PRECIPITATION
OWENS VALLEY AREA WEATHER STATIONS
30-yr MEAN ANNUAL PRECIPITATION (cm)
0
10
20
30
40
50
60
-1000
0
1000
2000
3000
4000
5000
ELEVATION OF STATION (meters)

FOR YOUR CHOICE: "LINE" Y=A+B*X
"A"= 2.3442E+00
"B" = 1.3947E-02

THE REGRESSION COEFFICIENT "R" = 0.84895

THE CALCULATED VALUES:

X	Y	YCALC	% DIFF
-5.9692E+01	5.2050E+00	1.5117E+00	70.96
5.2154E+02	1.0128E+01	9.6180E+00	5.04
7.5077E+02	1.0667E+01	1.2815E+01	-20.14
1.2154E+03	1.3821E+01	1.9295E+01	-39.61
1.2640E+03	1.4385E+01	1.9973E+01	-38.85
1.9846E+03	3.3231E+01	3.0023E+01	9.65
2.7600E+03	4.8436E+01	4.0837E+01	15.69
2.9677E+03	5.8590E+01	4.3734E+01	25.36
3.1231E+03	3.5026E+01	4.5901E+01	-31.05
3.8369E+03	5.0077E+01	5.5857E+01	-11.54

THE NONLINEAR CORRELATION COEFFICIENT: R**2= 0.84895211
the ave. % diff= 26.78702736

```
FOR YOUR CHOICE:  "LINE"   Y=A+B*X
 "A"=       3.2587E+00
 "B" =      1.1224E-02

THE REGRESSION COEFFICIENT "R" =      0.97240
```

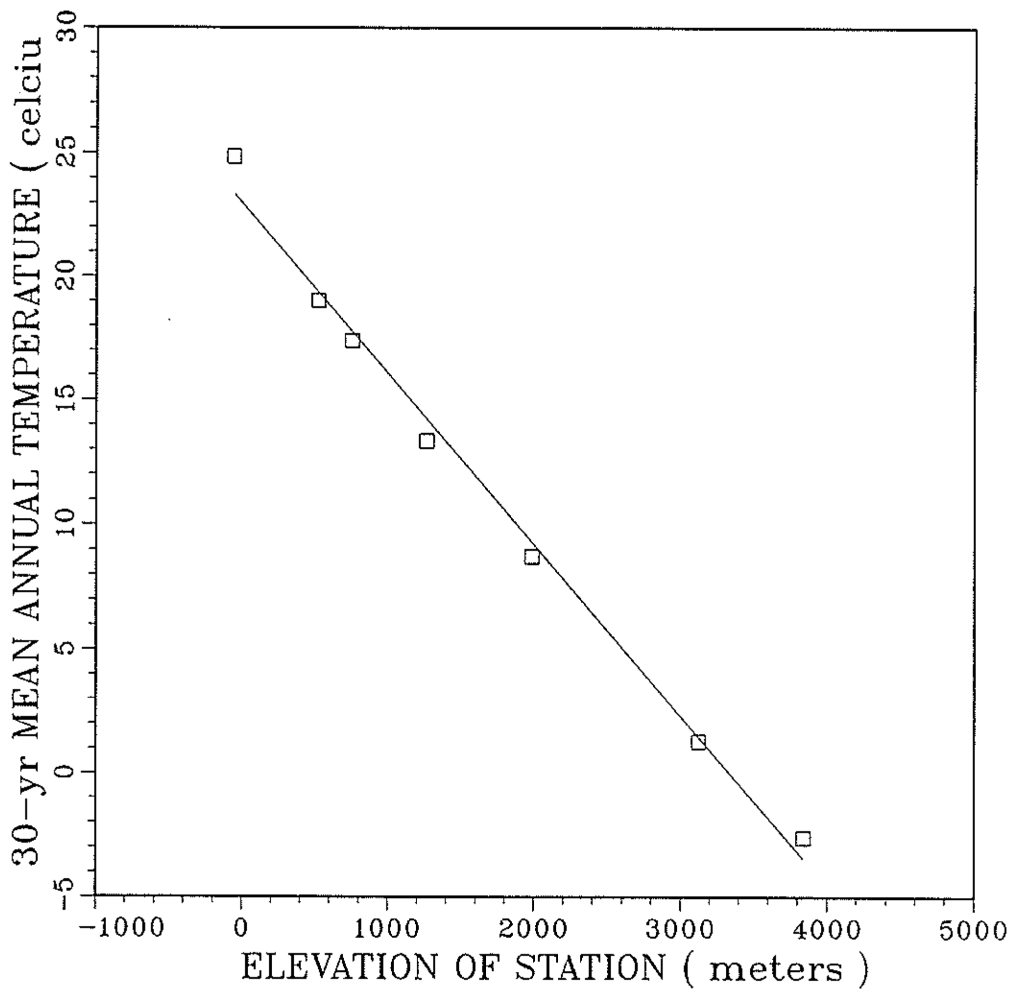
THE CALCULATED VALUES:

X	Y	YCALC	% DIFF
-5.9692E+01	5.2050E+00	2.5887E+00	50.26
5.2154E+02	1.0128E+01	9.1125E+00	10.03
7.5077E+02	1.0667E+01	1.1685E+01	-9.55
1.2154E+03	1.3821E+01	1.6900E+01	-22.28
3.1231E+03	3.5026E+01	3.8312E+01	-9.38
3.8369E+03	5.0077E+01	4.6325E+01	7.49

```
THE NONLINEAR CORRELATION COEFFICIENT: R**2=  0.97240067
the ave. % diff= 18.16574287
```

ELEVATION vs. 30-yr MEAN ANNUAL TEMPERATURE

OWENS VALLEY AREA WEATHER STATIONS



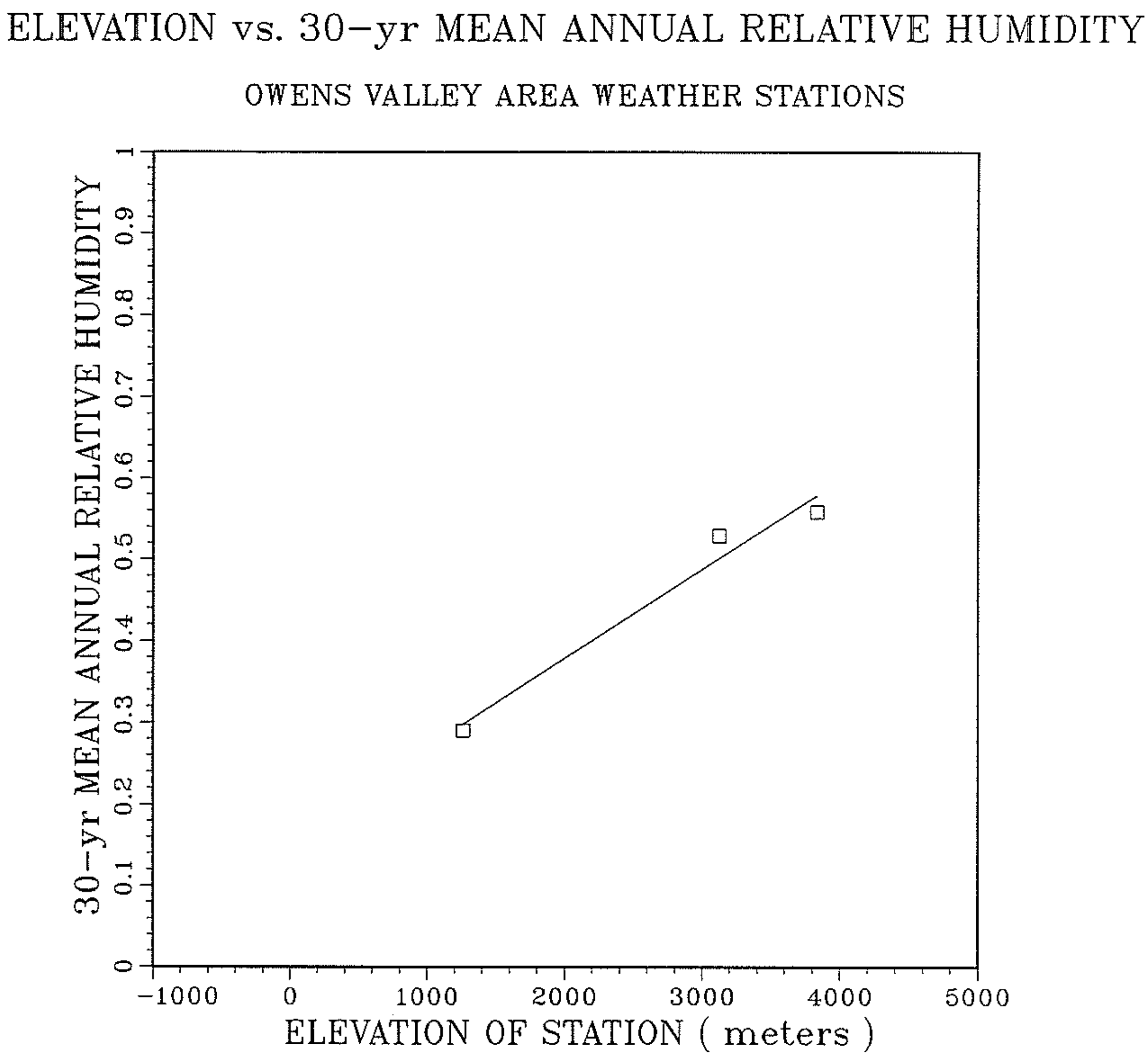
```
FOR YOUR CHOICE:  "LINE"   Y=A+B*X
 "A"=       2.2934E+01
 "B" =     -6.8820E-03

THE REGRESSION COEFFICIENT "R" =      0.99251
```

THE CALCULATED VALUES:

X	Y	YCALC	% DIFF
-5.9692E+01	2.4830E+01	2.3345E+01	5.98
5.2154E+02	1.9000E+01	1.9345E+01	-1.82
7.5077E+02	1.7390E+01	1.7768E+01	-2.17
1.2640E+03	1.3330E+01	1.4236E+01	-6.79
1.9846E+03	8.7200E+00	9.2763E+00	-6.38
3.1231E+03	1.2800E+00	1.4414E+00	-12.61
3.8369E+03	-2.6100E+00	-3.4713E+00	-33.00

```
THE NONLINEAR CORRELATION COEFFICIENT: R**2=  0.99251443
the ave. % diff=  9.82137108
```


ELEVATION vs. 30−yr MEAN ANNUAL RELATIVE HUMIDITY
OWENS VALLEY AREA WEATHER STATIONS
30−yr MEAN ANNUAL RELATIVE HUMIDITY
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
−1000
0
1000
2000
3000
4000
5000
ELEVATION OF STATION (meters)

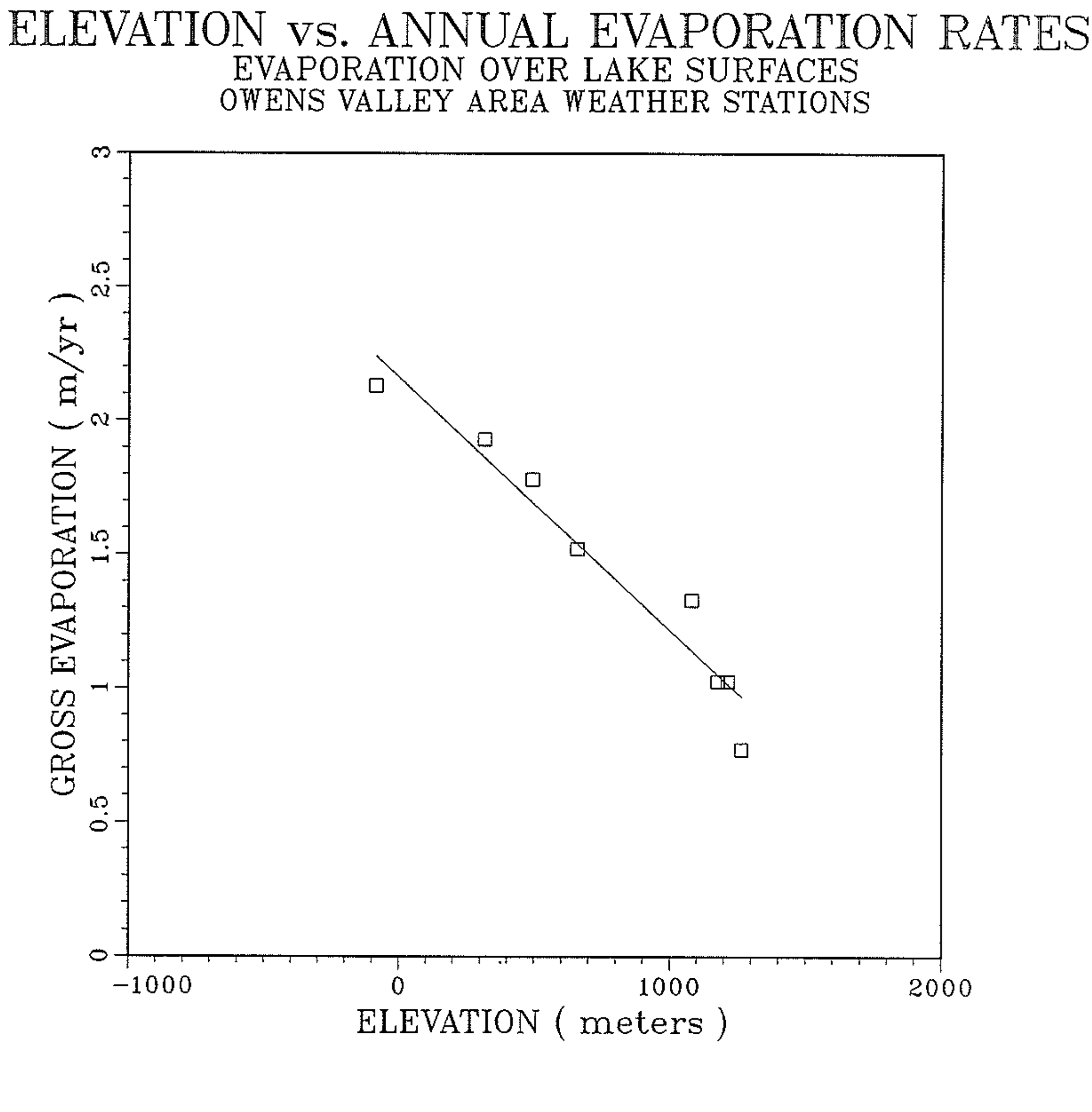
FOR YOUR CHOICE: "LINE" Y=A+B*X
"A"= 1.6001E-01
"B" = 1.0907E-04

THE REGRESSION COEFFICIENT "R" = 0.97026

THE CALCULATED VALUES:

X	Y	YCALC	% DIFF
1.2640E+03	2.9000E-01	2.9787E-01	-2.71
3.1231E+03	5.2900E-01	5.0064E-01	5.36
3.8369E+03	5.5800E-01	5.7849E-01	-3.67

THE NONLINEAR CORRELATION COEFFICIENT: R**2= 0.97026372
the ave. % diff= 3.91607213


ELEVATION vs. ANNUAL EVAPORATION RATES
EVAPORATION OVER LAKE SURFACES
OWENS VALLEY AREA WEATHER STATIONS
GROSS EVAPORATION (m/yr)
0
0.5
1
1.5
2
2.5
3
-1000
0
1000
2000
ELEVATION (meters)

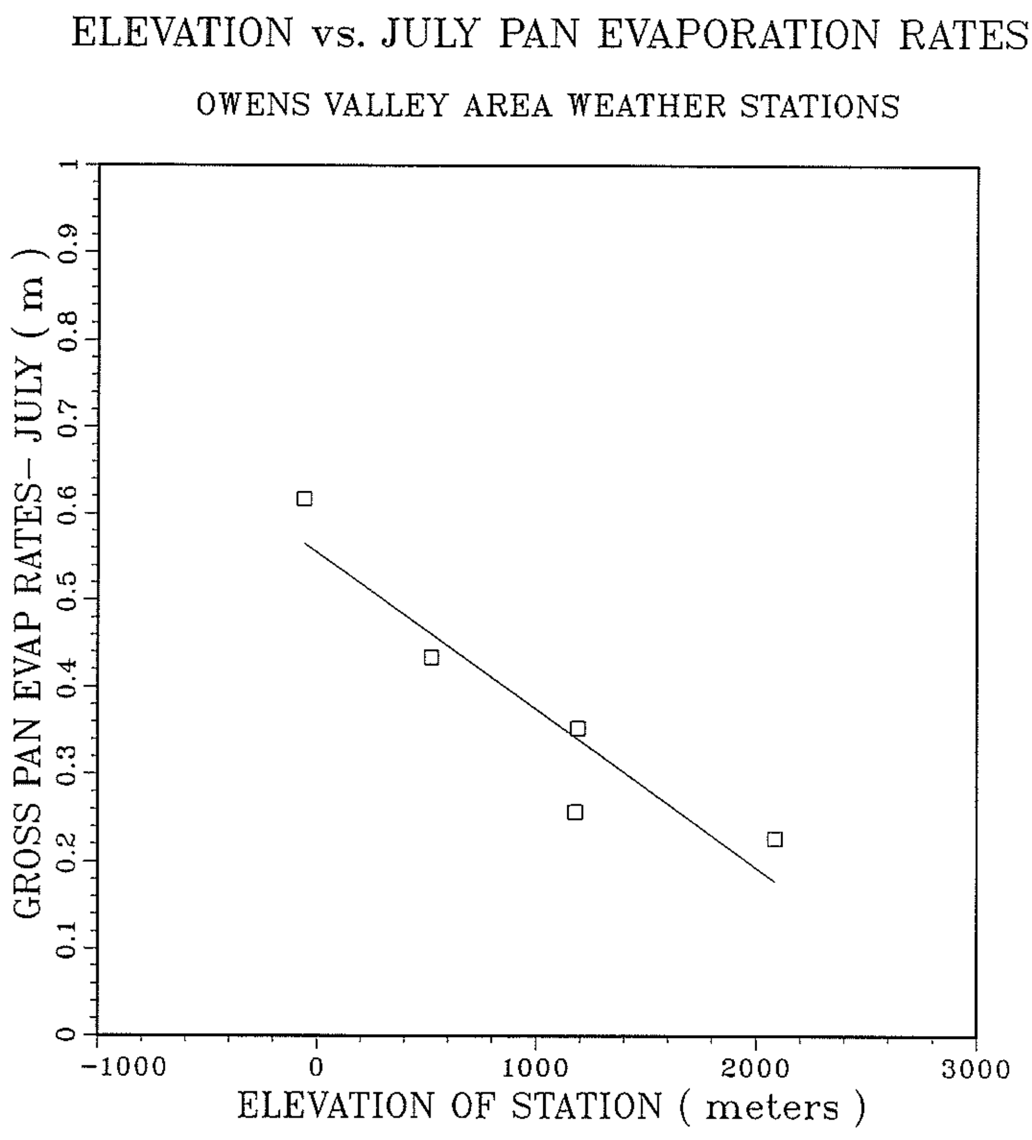
FOR YOUR CHOICE: "LINE" Y=A+B*X
"A"= 2.1607E+00
"B" = -9.4398E-04

THE REGRESSION COEFFICIENT "R" = 0.93882

THE CALCULATED VALUES:

X	Y	YCALC	% DIFF
-8.6000E+01	2.1300E+00	2.2418E+00	-5.25
3.1700E+02	1.9300E+00	1.8614E+00	3.55
4.9300E+02	1.7800E+00	1.6953E+00	4.76
6.5700E+02	1.5200E+00	1.5405E+00	-1.35
1.0810E+03	1.3300E+00	1.1402E+00	14.27
1.1770E+03	1.0250E+00	1.0496E+00	-2.40
1.2150E+03	1.0250E+00	1.0137E+00	1.10
1.2640E+03	7.7000E-01	9.6747E-01	-25.64

THE NONLINEAR CORRELATION COEFFICIENT: R**2= 0.93882328
the ave. % diff= 7.29056263


ELEVATION vs. JULY PAN EVAPORATION RATES
OWENS VALLEY AREA WEATHER STATIONS
GROSS PAN EVAP RATES– JULY (m)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
−1000
0
1000
2000
3000
ELEVATION OF STATION (meters)

FOR YOUR CHOICE: "LINE" Y=A+B*X
"A"= 5.5555E-01
"B" = -1.8106E-04

THE REGRESSION COEFFICIENT "R" = 0.86592

THE CALCULATED VALUES:

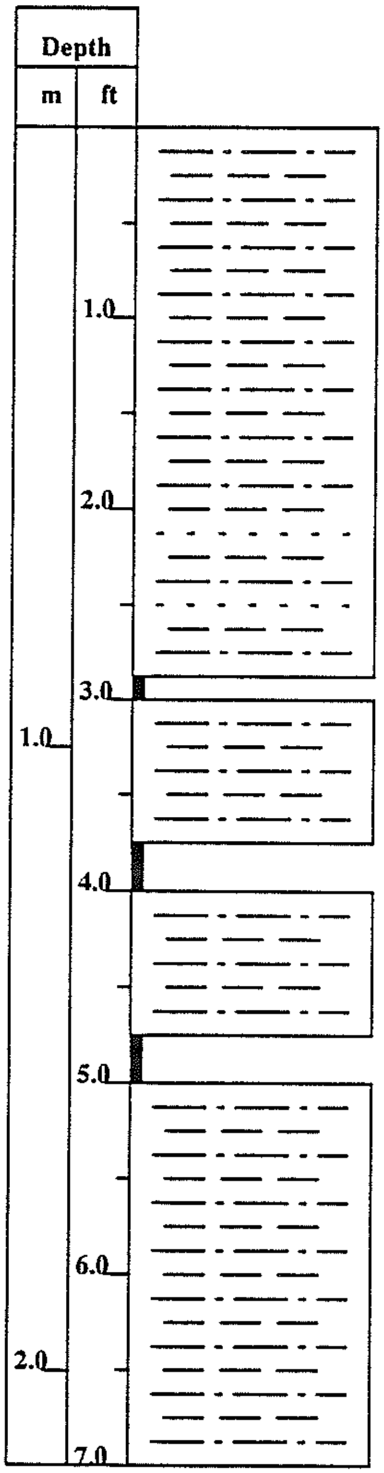
X	Y	YCALC	% DIFF
-5.9690E+01	6.1700E-01	5.6635E-01	8.21
5.2154E+02	4.3400E-01	4.6112E-01	-6.25
1.1769E+03	2.5700E-01	3.4246E-01	-33.25
1.1892E+03	3.5300E-01	3.4023E-01	3.62
2.0861E+03	2.2700E-01	1.7784E-01	21.66

THE NONLINEAR CORRELATION COEFFICIENT: R**2= 0.86591488
the ave. % diff= 14.59685802

Appendix J

Lithologic Log of San Agustin Core SAC-3/4

Plains of San Agustin, New Mexico
Lithologic Log -- Core SAC-3/4



Lithostratigraphy

0.0–1.0 ft: Clayey and silty surficial sediments with plant and root debris throughout; calcareous below 0.67 ft; light olive gray (5Y6/1).

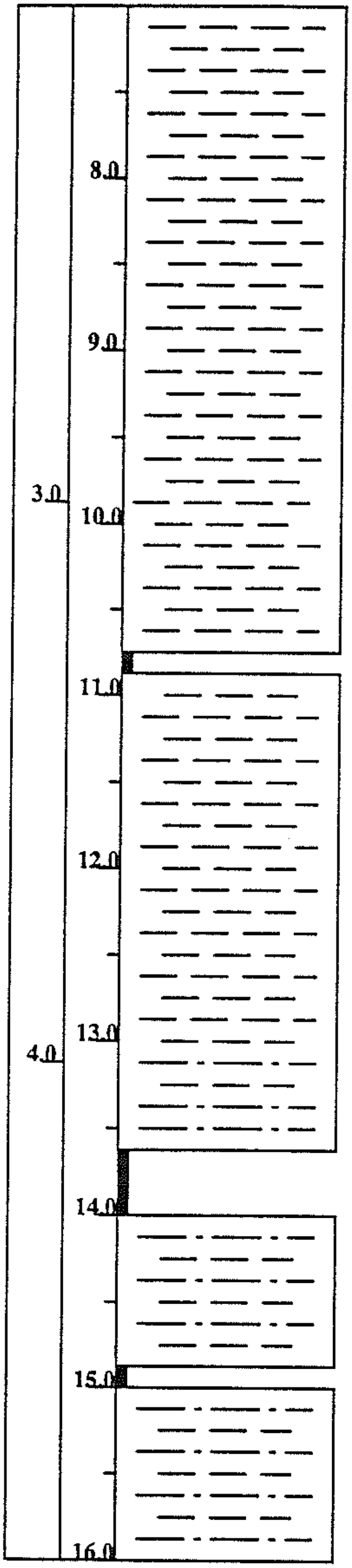
1.0–2.17 ft: Silty clay with plant roots abundant throughout; very calcareous; light olive gray (5Y6/1) to olive gray (5Y4/1).

2.17–2.92 ft: Sandy, silty clay with plant roots common throughout; very calcareous; light olive gray (5Y6/1) to dark yellowish brown (10YR4/2).

3.0–3.83 ft: Silty clay with iron-stained grains; very calcareous; dark yellowish brown (10YR4/2) to dusky yellowish brown (10YR2/2). XRD sample taken at 3.79 ft.

4.0–4.75 ft: Silty clay with organic remains and iron-stained grains; very calcareous; dark yellowish brown (10YR4/2).

5.0–7.0 ft: Silty clay, same as above.



7.0–8.0 ft: Fine clay; very calcareous; dark yellowish brown (10YR4/2). Laminations of yellowish gray (5Y8/1) clay at 7.83 ft. XRD samples taken at: (1) 7.38 ft indicate presence of illite (1/10), smectite (5/10), kaolinite (1/10) and mixed layer illite/smectite (2/10 with illite < 5%) and calcite; and (2) 7.92 ft indicate presence of illite (3/10), smectite (6/10) and kaolinite (1/10) and calcite.

8.0–9.0 ft: Fine calcareous clay, same as above, with yellowish gray laminations at 8.48 and 8.71 ft.

9.0–10.0 ft: Fine calcareous clay, same as above, with organic residue common throughout; yellowish gray lamination at 9.71 ft.

10.0–10.75 ft: Fine calcareous clay, same as above, with abundant organic residue; light olive gray (5Y6/1) and yellowish gray (5Y8/1) varve-like microlaminations from 10.58 to 10.63 ft. XRD sample taken at 10.58 ft indicates presence of illite (2/10), smectite (2/10), kaolinite (1/10) and mixed layer illite/smectite (5/10 with illite < 5%) and calcite.

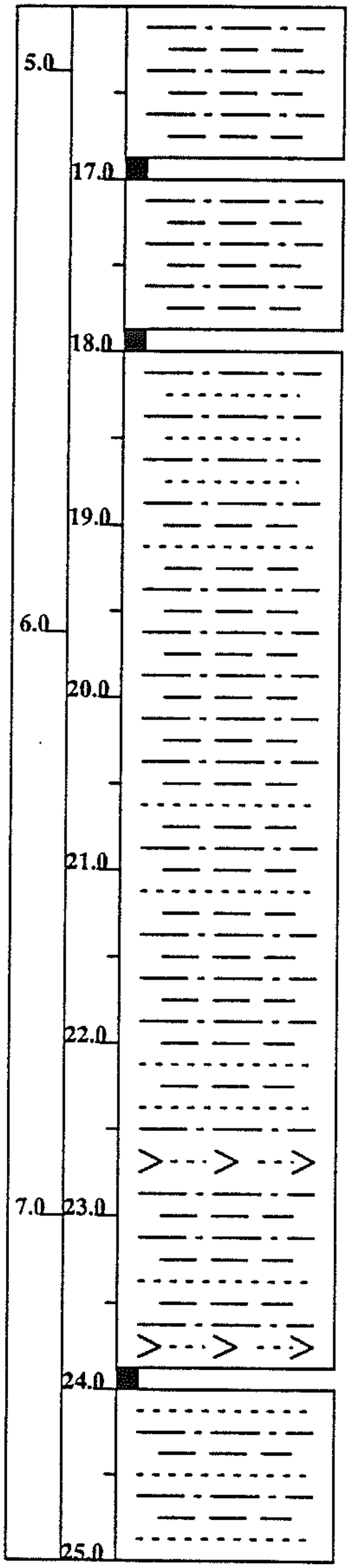
10.92–12.0 ft: Fine calcareous clay, same as above; no organic residue; yellowish gray clay layer at 11.17 ft.

12.0–13.0 ft: Fine clay; moderately calcareous; light olive gray (5Y5/2). Yellowish gray laminations at 12.0 ft.

13.0–13.67 ft: Silty clay, very calcareous; mottled light olive gray (5Y5/2) and moderate brown (5YR4/4). Yellowish gray clay laminations at 13.0 ft, 13.67 ft, and in laminated zones throughout. XRD sample taken at 13.58 ft.

14.0–14.83 ft: Silty clay; calcareous; pale yellowish brown (10YR7/2) to dark yellowish brown (10YR4/2–3/2), with light olive gray (5Y5/2) and pinkish gray (5YR8/1) clay in discontinuous zones throughout. Moderate brown (5YR4/4) iron oxidation zones throughout.

15.0–16.0 ft: Silty clay; very calcareous; dark yellowish brown (10YR4/2) with pinkish gray (5YR7/1) zones and moderate brown oxidized zones throughout.



16.0—16.83 ft: Silty clay; calcareous; interlayered light olive gray (5Y5/2–4/2) and yellowish gray (5Y7/1); minor oxidation.

17.0–17.92 ft: Silty clay with interbedded fine clay; calcareous; yellowish gray (5Y7/1) to light olive gray (5Y5/2–4/2)

18.0–18.83 ft: Silt and very fine sand with iron-coated grains; calcareous.

18.83–20.17 ft: Silty clay; calcareous; dark yellowish brown (10YR4/2) to olive gray (5Y4/1) with yellowish gray (5Y7/1) clay in discontinuous zones to 19.75 ft. Layer of very fine sand from 19.08 to 19.13 ft.

20.17–20.42 ft: Clayey silt with very fine sand (20%), gypsum crystals, and iron-coated grains; calcareous; pale yellowish brown (10YR5/2).

20.42–21.25 ft: Silty clay with very fine sand (10%); calcareous; dark yellowish brown (10YR4/2) with moderate brown (5YR4/4) oxidation zones.

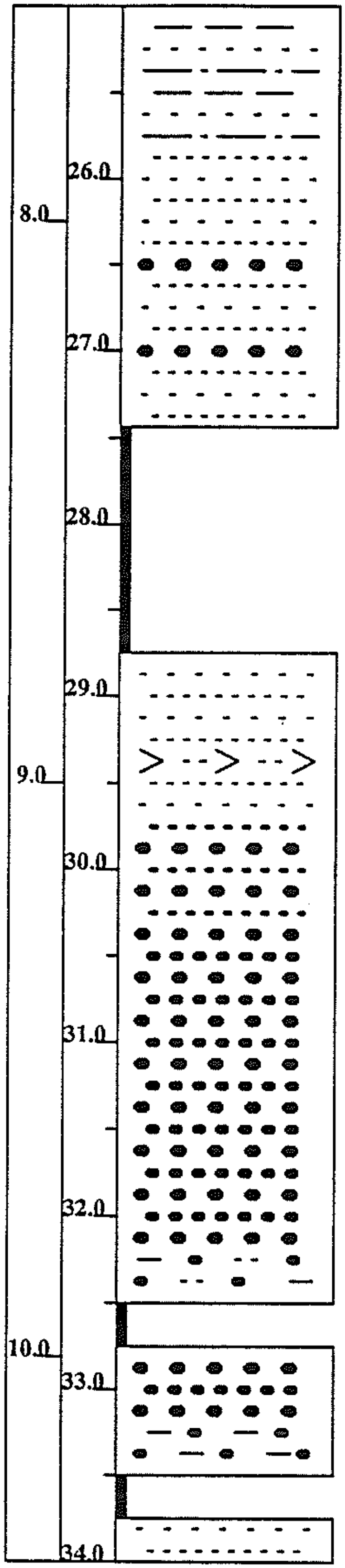
21.25–22.0 ft: Silty clay; calcareous; dark yellowish brown (10YR4/2).

22.0–22.50 ft: Very fine sand with minor clay (20%); light olive gray (5Y6/1).

22.50–22.75 ft: Silty clay with trace of very fine sand; calcareous; dark yellowish brown (10YR3/2); large zone of gypsum at 22.67 to 22.75 ft.

22.75–23.92 ft: Clayey silt and silty clay with very fine sand; light olive gray (5Y6/1) to olive gray (5Y4/1); large zone of gypsum from 23.75 to 23.83 ft; occasional vertebrate bones.

24.0–25.0 ft: Very fine sand and silt, with minor clay; calcareous; light olive gray (5Y6/1).



25.0–26.0 ft: Very fine sand and silt with minor clay; calcareous; light olive gray (5Y6/1).

26.0–26.5 ft: Moderately well sorted, medium to coarse, salt and pepper sand (abrupt contact with overlying very fine sand at 26.0 ft); slightly calcareous; light olive gray (5Y6/1). Sand coarsens downward to gravel layer at 26.5 ft.

26.5–27.0 ft: Poorly sorted, fine to coarse sand with occasional large (to 1 cm diameter) lithic granules. Gravel layer at 27.0 ft.

27.0–27.42 ft: Fine sand with minor coarse sand and occasional large (to 3 mm diameter) lithic granules.

28.75–29.92 ft: Fine to medium, salt and pepper sand with small lithic granules and selenite gypsum crystals; slightly calcareous; yellowish gray (5Y8/1) to light olive gray (5Y5/2). Sand coarsens downward to gravel and sand at 29.92 ft.

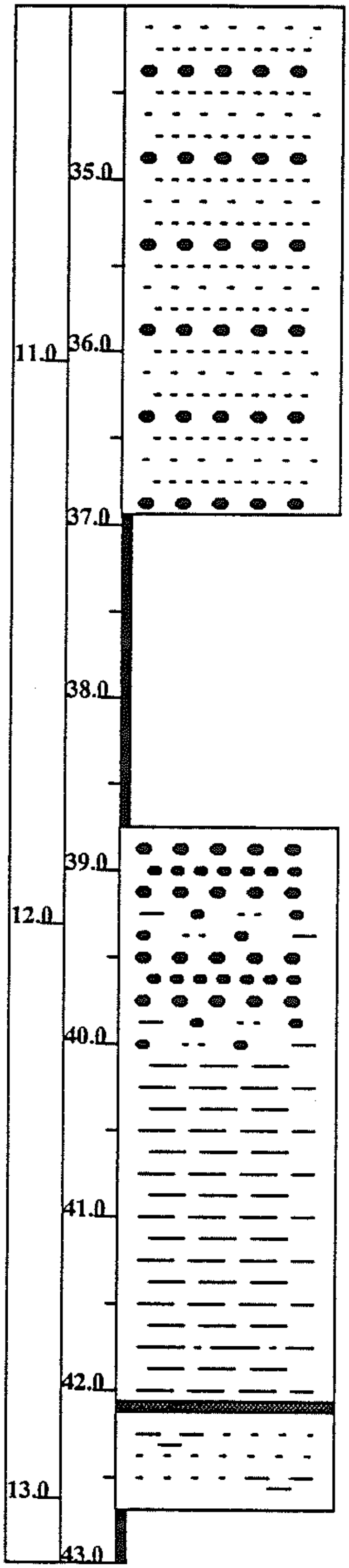
29.92–30.42 ft: Unsorted gravel and sand (granules to 2 cm diameter); noncalcareous.

30.42–32.33 ft: Gravel; heavily oxidized; dark reddish brown (10R3/4) pebbles coated with dark yellowish orange (10YR6/6) limonite/geothite.

32.33–32.5 ft: Clayey-gravelly sand; heavily oxidized.

32.75–33.50 ft: Gravel and sand, heavily oxidized, with white and reddish brown zonation as above; locally slightly calcareous. Clay content increases downward to clayey-gravelly sand at 33.5 ft.

33.75–34.0 ft: Fine sand with some coarse sand and lithic granules; strongly oxidized; moderate yellowish brown (10YR5/4) to moderate brown (5YR4/4).

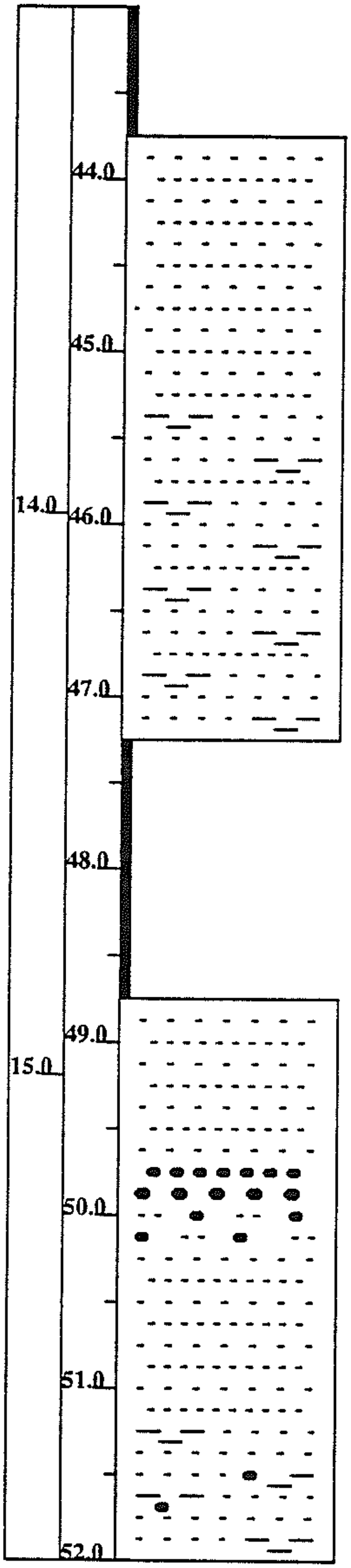


34.0–36.92 ft: Sand and gravel; oxidized; wet; moderate yellowish brown (10YR5/4) to moderate brown (5YR4/4).

38.75–40.0 ft: Gravel with minor sand and clay; large angular clasts to 4 cm length; oxidized.

40.0–42.08 ft: Clay; slightly calcareous in upper portion grading to non-calcareous at base; pale olive (10Y6/2) to light olive gray (5Y5/2) with zones of greenish black (5G2/1) and moderate yellowish brown (10YR5/4). 1 mm lamination of yellowish gray, calcareous silt at 41.75 ft.

42.08–42.67 ft: Abrupt contact at 42.08 ft formed by 1-mm lamination of black organic-rich material; remainder of section is very fine to fine clayey sand; noncalcareous; dark yellowish brown (10YR4/2).



43.75–45.33 ft: Poorly sorted, salt and pepper, medium sand with very coarse sand and occasional small lithic granules; noncalcareous; moderate yellowish brown (10YR5/4) to dark yellowish brown (10YR4/2).

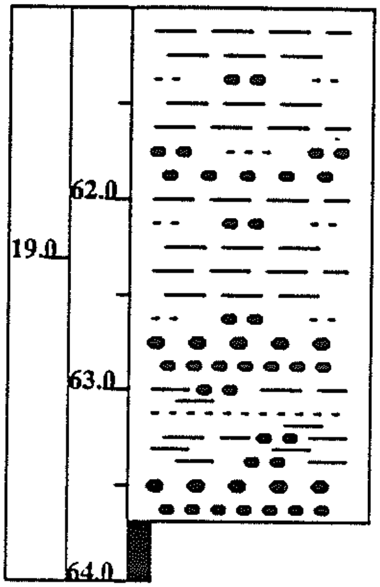
45.33–47.25 ft: (Graded contact, 45.25 to 45.42 ft.) Well sorted very fine clayey sand and sandy clay, with occasional lithic granules; noncalcareous in upper section, and slightly calcareous below 45.75 ft.

48.75–49.75 ft: Poorly sorted, coarse sand with abundant lithic granules, pebbles and some medium–fine sand; noncalcareous and oxidized; dark yellowish brown (10YR4/2) to moderate brown (5YR4/4) coarsens downward to dominantly gravel at 49.75 ft.

49.75–50.58 ft: Poorly sorted, coarse sand, same as above; fines downward to medium sand at 50.58 ft.

50.58–51.25 ft: Medium sand with abundant lithic granules and pebbles; slightly oxidized; pale yellowish brown (10YR6/2).

51.25–51.92 ft: Clay, sand and fine gravel (pebbles to 5mm diameter); oxidized pebbles common; dark yellowish brown (10YR4/2) to dusky yellowish brown (10YR2/2).

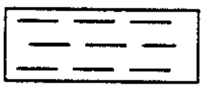


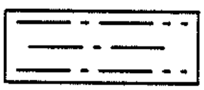
61.75 ft: Strongly oxidized gravel layer.

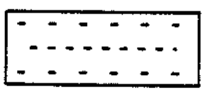
62.67–63.67 ft: Gravel (~50–60%) and clay (~40%) with minor medium to coarse sand; minor oxidation; high water content.

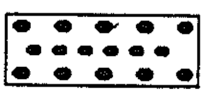
63.67 ft: Gravel (~70%) and clay (~25%).

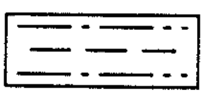
Explanation

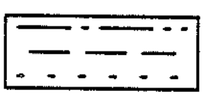
 Clay

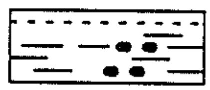
 Silt

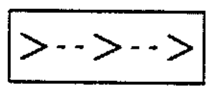
 Sand

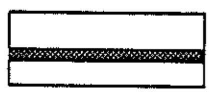
 Gravel

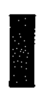
 Silty Clay

 Silty Sandy Clay

 Clay–Sand–Gravel

 Gypsum

 Organic–Rich Sediment

 Unrecovered core

(10YR5/2) U.S. National Bureau of Standards Color Designation

Appendix K

Relative Abundance of Ostracode Species with Depth in San Agustin Core SAC 3/4

Depth Interval	Limnocythere bradburyi	L.ceriotuberosa	L.platyforma	Candona patzcuaro
		Percent		
3.67 - 3.79	75	25	0	0
4.08 - 4.17	67	33	0	0
4.25 - 4.33	45	10	0	45
4.42 - 4.50	45	10	0	45
4.58 - 4.67	75	25	0	0
4.83 - 5.00	47.5	5	0	47.5
5.08 - 5.17	12.5	2.5	0	85
5.25 - 5.33	12.5	2.5	0	85
5.42 - 5.50	5	5	0	90
5.58 - 5.67	37.5	12.5	0	50
5.83 - 6.00	40	20	0	40
6.08 - 6.17	32	8	0	60
6.25 - 6.33	25	5	0	70
6.33 - 6.42	5	0	0	95
6.42 - 6.58	48.5	3	0	48.5
6.58 - 6.67	37.5	12.5	0	50
6.75 - 6.83	20	5	0	75
6.83 - 6.92	42.5	15	0	42.5
6.92 - 7.00	42.5	15	0	42.5
7.08 - 7.17	30	20	0	50
7.25 - 7.33	47.5	47.5	0	5
7.42 - 7.50	33	67	0	0
7.58 - 7.67	5	70	0	25
7.67 - 7.75	95	3	0	2
7.83 - 8.00	100	0	0	0
8.08 - 8.17	100	0	0	0
8.25 - 8.33	100	0	0	0
8.42 - 8.50	100	0	0	0
8.58 - 8.67	98.5	0	1.5	0
8.67 - 8.75	.90	8	2	0
8.83 - 9.00	90	5	5	0
9.00 - 9.17	62.5	37.5	0	0
9.25 - 9.33	50	50	0	0
9.42 - 9.50	96	2	2	0
9.58 - 9.67	100	0	0	0
9.83 - 9.92	60	40	0	0
9.92 - 10.0	75	25	0	0
10.08 - 10.17	100	0	0	0
10.25 - 10.33	96	3	1	0
10.42 - 10.50	98	2	0	0
10.58 - 10.67	50	37.5	12.5	0
10.92 - 11.00	100	0	0	0
11.08 - 11.17	46	46	8	0

11.25 - 11.33	15	77.5	7.5	0
11.42 - 11.50	2.5	97.5	0	0
11.58 - 11.67	15	85	0	0
11.67 - 11.75	15	85	0	0
11.83 - 11.92	20	80	0	0
11.92 - 12.00	5	90	5	0
12.08 - 12.17	55	45	0	0
12.25 - 12.33	33	67	0	0
12.42 - 12.50	2	96	2	0
12.67 - 12.83	20	80	0	0
12.83 - 13.00	15	85	0	0
13.08 - 13.17	10	90	0	0
13.25 - 13.33	85	15	0	0
13.42 - 13.50	98	2	0	0
13.58 - 13.67	80	20	0	0
14.00 - 14.08	70	15	15	0
14.08 - 14.17	65	25	10	0
14.17 - 14.25	85	15	0	0
14.25 - 14.33	65	35	0	0
14.33 - 14.42	65	20	15	0
14.42 - 14.50	25	25	50	0
14.50 - 14.58	45	20	35	0
14.58 - 14.67	60	25	15	0
14.67 - 14.75	65	25	10	0
15.00 - 15.08	70	20	10	0
15.08 - 15.17	80	15	5	0
15.17 - 15.25	80	15	5	0
15.25 - 15.33	95	5	0	0
15.42 - 15.50	85	15	0	0
15.58 - 15.67	100	0	0	0
15.67 - 15.75	95	5	0	0
15.83 - 15.96	95	5	0	0
16.00 - 16.08	100	0	0	0
16.08 - 16.17	87.5	6.25	6.25	0
16.25 - 16.33	90	10	0	0
16.58 - 16.67	80	5	15	0
16.75 - 16.83	96	2	2	0
17.00 - 17.08	95	5	0	0
17.08 - 17.17	90	8	2	0
17.25 - 17.33	90	2	8	0
17.42 - 17.50	75	25	0	0
17.58 - 17.67	65	35	0	0
17.67 - 17.75	50	50	0	0
17.83 - 17.92	60	40	0	0
18.00 - 18.08	50	0	50	0
18.08 - 18.17	60	5	35	0
18.25 - 18.33	60	5	35	0
18.42 - 18.50	70	12.5	17.5	0
18.50 - 18.58	75	15	10	0
18.58 - 18.67	90	5	5	0

18.67 - 18.75	90	6.5	3.5	0
18.83 - 18.92	50	25	25	0
18.92 - 19.00	85	5	10	0
19.08 - 19.17	92.5	2.5	5	0
19.25 - 19.33	85	12.5	2.5	0
19.42 - 19.50	96	2	2	0
19.50 - 19.58	95	1.5	3.5	0
19.58 - 19.67	95	1	4	0
19.67 - 19.75	95	1	4	0
19.83 - 19.92	50	15	35	0
19.92 - 20.00	55	10	35	0
20.08 - 20.17	60	10	30	0
20.25 - 20.33	60	10	30	0
20.42 - 20.50	60	15	25	0
20.50 - 20.58	55	15	30	0
20.58 - 20.67	50	15	35	0
20.75 - 20.83	25	50	25	0
20.83 - 20.92	50	50	0	0
20.92 - 21.00	55	45	0	0
21.13 - 21.25	95	2	3	0
21.25 - 21.33	30	65	5	0
21.42 - 21.50	55	45	0	0
21.58 - 21.67	65	35	0	0
21.67 - 21.75	90	10	0	0
21.75 - 21.83	90	10	0	0
21.83 - 21.92	95	5	0	0
21.92 - 22.00	97.5	2.5	0	0
22.08 - 22.25	80	10	10	0
22.25 - 22.33	90	9	1	0
22.42 - 22.50	95	5	0	0
22.58 - 22.67	95	5	0	0
22.67 - 22.75	90	0	10	0
22.83 - 22.92	75	12.5	12.5	0
22.92 - 23.00	55	45	0	0
23.08 - 23.17	80	12	8	0
23.25 - 23.33	15	85	0	0
23.42 - 23.50	80	20	0	0
23.58 - 23.67	85	15	0	0
23.67 - 23.75	100	0	0	0
23.83 - 23.92	85	15	0	0
24.00 - 24.17	2.5	95	2.5	0
24.25 - 24.33	0	87.5	12.5	0
24.42 - 24.50	2.5	95	2.5	0
24.58 - 24.67	9	100	0	0
24.75 - 24.83	0	100	0	0
24.83 - 24.92	2.5	97.5	0	0
24.92 - 25.00	2.5	97.5	0	0
25.08 - 25.17	0	100	0	0
25.25 - 25.33	2.5	97.5	0	0

25.42 - 25.63	0	100	0	0
26.00 - 26.17	2.5	97.5	0	0
26.17 - 26.33	0	100	0	0
26.33 - 26.50	0	100	0	0
27.00 - 27.21	0	100	0	0
27.21 - 27.42	0	100	0	0

Appendix L

San Agustin Isotopic Results

Sample No.	Depth (ft)	Age (ka)	Oxygen-18	Carbon-13	Ostracode Species/Assemblage
			CORE SA4		
LHMX3-1-2	3.13	14.54	31.761	-2.246	L.bradburyi/L.cerio-tuberosa/C.patzcuaro/H.spA
LB3-8-9.5	3.73	15.04	31.427	-3.091	L.bradburyi
LMX4-1-2	4.13	15.36	33.081	-1.551	L.bradburyi/L.ceriotuberosa
LHMX4-3-4	4.29	15.49	32.556	-1.481	L.bradburyi/L.cerio-tuberosa/C.patzcuaro/H.spA
LB4-7-8	4.63	15.77	31.832	-0.270	L.bradburyi
LB4-10-12	4.92	16.01	31.259	-1.221	L.bradburyi
HNS5-3-4	5.29	16.31	32.406	-2.239	C.patzcuaro/H.spA
HNS5-5-6	5.46	16.45	32.540	-1.412	C.patzcuaro/H.spA
LB5-7-8	5.63	16.59	31.071	-3.565	L.bradburyi
LB5-8-9	5.71	16.66	31.478	-1.964	L.bradburyi
LB6-1-2	6.13	17.00	30.632	-2.506	L.bradburyi
LB6-3-4	6.29	17.13	29.986	-4.325	L.bradburyi
LB6-4-5	6.38	17.20	30.555	-3.384	L.bradburyi
LB6-5-7	6.50	17.30	30.558	-2.439	L.bradburyi
LB6-7-8	6.63	17.41	30.612	-2.700	L.bradburyi
LB6-9-10	6.79	17.54	31.506	-1.335	L.bradburyi
LB6-10-11	6.88	17.62	31.647	-1.518	L.bradburyi
LB-6-11-12	6.96	17.68	32.667	-0.307	L.bradburyi
LB7-1-2	7.13	17.82	32.178	-0.234	L.bradburyi
LB7-3-4	7.29	17.95	32.417	-0.141	L.bradburyi
LB7-5-6	7.46	18.09	31.275	-0.853	L.bradburyi
LC-7-7-8	7.63	18.23	32.553	-1.808	L.ceriotuberosa
LB7-8-9	7.71	18.30	30.157	-2.370	L.bradburyi
LB7-10-12	7.92	18.43	31.700	-1.550	L.bradburyi
LB8-1-2	8.13	18.56	30.512	-1.910	L.bradburyi
LB8-3-4	8.29	18.67	32.269	-0.912	L.bradburyi
LB8-5-6	8.46	18.77	31.673	-1.627	L.bradburyi
LB8-7-8	8.63	18.88	32.228	-1.844	L.bradburyi
LB8-8-9	8.71	18.93	31.682	-1.265	L.bradburyi
LB-8-10-12	8.92	19.07	30.684	-2.040	L.bradburyi
LB9-0-2	9.08	19.17	31.066	-0.461	L.bradburyi
LB9-3-4	9.29	19.30	30.649	-2.064	L.bradburyi
LB9-5-6	9.46	19.41	28.764	-1.980	L.bradburyi
LB9-7-8	9.63	19.52	32.571	-2.196	L.bradburyi
LB-9-10-11	9.88	19.67	32.284	-1.037	L.bradburyi

LB-9-11-12	9.96	19.72	33.055	-0.277	L.bradburyi
LB10-1-2	10.13	19.83	31.415	-1.584	L.bradburyi
LB10-3-4	10.29	19.93	32.134	-2.050	L.bradburyi
LB10-5-6	10.46	20.04	31.766	-2.528	L.bradburyi
LMX10-7-8	10.63	20.15	30.970	-1.413	L.bradburyi/L.ceriotuberosa/L.platyforma
LB10-11-12	10.96	20.36	31.855	-3.150	L.bradburyi
LB11-1-2	11.13	20.46	30.228	-3.669	L.bradburyi
LB11-3-4	11.29	20.57	31.260	-2.723	L.bradburyi
LC11-5-6	11.46	20.67	32.247	-0.653	L.ceriotuberosa
LB11-7-8	11.63	20.78	31.332	-1.417	L.bradburyi
LC11-8-9	11.71	20.83	32.516	-0.132	L.ceriotuberosa
LB11-10-11	11.88	20.94	30.525	-1.923	L.bradburyi
LB11-11-12	11.96	20.99	31.189	-1.512	L.bradburyi
LB12-1-2	12.13	21.10	32.083	-1.649	L.bradburyi
LB12-3-4	12.29	21.20	31.811	-1.699	L.bradburyi
LC12-5-6	12.46	21.31	31.828	-0.416	L.ceriotuberosa
LC12-7-8	12.63	21.42	31.443	-2.253	L.ceriotuberosa
LB12-8-10	12.75	21.49	31.421	-1.347	L.bradburyi
LB12-10-12	12.92	21.60	31.001	-1.773	L.bradburyi
LMX13-1-2	13.13	21.73	31.668	-1.036	L.bradburyi/L.ceriotuberosa
LB13-3-4	13.29	21.83	31.903	-2.518	L.bradburyi
LB13-5-6	13.46	21.94	32.468	-2.729	L.bradburyi
LMX13-7-8	13.63	22.05	31.970	-2.568	L.bradburyi/L.ceriotuberosa
LMX14-0-1	14.04	22.31	31.181	-3.231	L.bradburyi/L.ceriotuberosa/L.platyforma
LMX14-1-2	14.13	22.36	30.876	-2.710	L.bradburyi/L.ceriotuberosa/L.platyforma
LB14-2-3	14.21	22.42	28.842	-4.272	L.bradburyi
LMX14-3-5	14.33	22.49	29.803	-3.615	L.bradburyi/L.ceriotuberosa/L.platyforma
LMX14-5-6	14.46	22.57	29.616	-4.021	L.bradburyi/L.ceriotuberosa/L.platyforma
LMX14-6-7	14.54	22.62	29.639	-3.558	L.bradburyi/L.ceriotuberosa/L.platyforma
LMX14-7-8	14.63	22.68	30.380	-2.778	L.bradburyi/L.ceriotuberosa/L.platyforma
LB14-8-9	14.71	22.73	30.913	-2.850	L.bradburyi
LB15-0-2	15.08	22.97	33.308	-2.456	L.bradburyi
LMX15-2-3	15.21	23.05	31.439	-2.255	L.bradburyi/L.ceriotuberosa/L.platyforma
LB15-3-4	15.29	23.10	31.611	-1.761	L.bradburyi
LB15-5-6	15.46	23.21	31.912	-1.579	L.bradburyi
LB15-7-8	15.63	23.32	29.721	-3.965	L.bradburyi
LB15-8-9	15.71	23.37	31.619	-2.943	L.bradburyi
LB15-10-11	15.90	23.49	32.305	-3.229	L.bradburyi
LB16-0-1	16.04	23.58	31.439	-2.321	L.bradburyi

LB16-1-2	16.13	23.63	30.623	-3.120	L.bradburyi
LB16-3-4	16.29	23.73	31.154	-1.997	L.bradburyi
LB16-5-6	16.46	23.84	32.777	-1.945	L.bradburyi
LMX16-7-8	16.63	23.95	30.980	-3.187	L.bradburyi/L.cerio-tuberosa/L.platyforma
LB16-9-10	16.79	24.05	31.036	-2.545	L.bradburyi
LB17-0-1	17.04	24.21	31.559	-2.154	L.bradburyi
LB17-1-2	17.13	24.26	31.715	-1.975	L.bradburyi
LB17-3-4	17.29	24.37	31.356	-2.364	L.bradburyi
LB17-5-6	17.46	24.47	31.145	-1.516	L.bradburyi
LB17-7-8	17.63	24.58	30.618	-2.135	L.bradburyi
LB17-8-9	17.71	24.63	31.180	-2.399	L.bradburyi
LMX17-10-11	17.88	24.74	29.825	-3.761	L.bradburyi/ L.ceriotuberosa
LB18-0-1	18.04	24.84	28.062	-5.297	L.bradburyi
LB18-1-2	18.13	24.90	28.217	-5.400	L.bradburyi
LB18-3-4	18.29	25.00	28.225	-5.405	L.bradburyi
LB18-5-6	18.46	25.11	29.746	-3.941	L.bradburyi
LB18-6-7	18.54	25.16	30.614	-3.025	L.bradburyi
LB18-7-8	18.63	25.22	30.859	-3.094	L.bradburyi
LB18-8-9	18.71	25.27	31.770	-2.475	L.bradburyi
LMX18-10-11	18.88	25.37	28.233	-4.561	L.bradburyi/L.cerio-tuberosa/L.platyforma
LB18-11-12	18.96	25.42	28.036	-4.730	L.bradburyi
LB19-1-2	19.13	25.53	27.583	-6.184	L.bradburyi
LB19-3-4	19.29	25.63	29.313	-4.882	L.bradburyi
LB19-5-6	19.46	25.74	30.313	-3.810	L.bradburyi
LB19-6-7	19.54	25.79	31.760	-3.357	L.bradburyi
LB19-7-8	19.63	25.85	30.891	-3.707	L.bradburyi
LB19-8-9	19.71	25.90	30.670	-3.261	L.bradburyi
LB19-10-11	19.88	26.01	28.701	-5.100	L.bradburyi
LB19-11-12	19.96	26.06	29.580	-5.266	L.bradburyi
LMX20-1-2	20.13	26.16	27.150	-5.819	L.bradburyi/ L.platyforma
LMX20-3-4	20.29	26.27	28.182	-4.880	L.bradburyi/L.cerio-tuberosa/L.platyforma
LCP20-5-6	20.46	26.37	28.845	-4.764	L.ceriotuberosa/ L.platyforma
LB20-6-7	20.54	26.42	28.882	-4.326	L.bradburyi
LB20-7-8	20.63	26.48	29.473	-3.929	L.bradburyi
LMX20-9-10	20.79	26.58	30.768	-2.969	L.bradburyi/L.cerio-tuberosa/L.platyforma
LB20-11-12	20.96	26.69	30.620	-2.689	L.bradburyi
LB21-1.5-3	21.19	26.84	31.742	-2.902	L.bradburyi
LMX21-3-4	21.29	26.90	31.272	-1.625	L.bradburyi/ L.ceriotuberosa
LMX21-5-6	21.46	27.01	31.535	-1.767	L.bradburyi/ L.ceriotuberosa

LMX21-7-8	21.63	27.12	32.419	-1.020	L.bradburyi/ L.ceriotuberosa
LB21-8-9	21.71	27.16	32.478	-1.628	L.bradburyi
LB21-9-10	21.79	27.22	32.114	-1.430	L.bradburyi
LB21-10-11	21.88	27.27	31.627	-2.041	L.bradburyi
LB21-11-12	21.96	27.32	31.107	-2.183	L.bradburyi
LB22-1-3	22.17	27.46	30.973	-2.388	L.bradburyi
LB22-3-4	22.29	27.53	31.248	-2.162	L.bradburyi
LB22-5-6	22.46	27.64	31.678	-1.783	L.bradburyi
LB22-7-8	22.63	27.75	30.749	-2.696	L.bradburyi
LB22-8-9	22.71	27.80	31.505	-3.733	L.bradburyi
LB22-10-11	22.88	27.91	30.586	-3.600	L.bradburyi
LB22-11-12	22.96	27.96	30.744	-2.772	L.bradburyi
LB23-1-2	23.13	28.06	30.532	-3.250	L.bradburyi
LC23-3-4	23.29	28.17	31.287	-1.514	L.ceriotuberosa
LB23-5-6	23.46	28.27	31.569	-1.504	L.bradburyi
LB23-7-8	23.63	28.38	31.136	-1.909	L.bradburyi
LB23-8-9	23.71	28.43	33.173	-1.579	L.bradburyi
LB23-10-11	23.88	28.54	31.450	-1.768	L.bradburyi
LMX24-0-2	24.08	28.67	30.588	-1.750	L.bradburyi/ L.ceriotuberosa
LC24-3-4	24.29	28.80	29.934	-1.912	L.ceriotuberosa
LC24-5-6	24.46	28.91	30.038	-1.259	L.ceriotuberosa
LC24-7-8	24.63	29.02	30.276	-1.244	L.ceriotuberosa
LC24-9-10	24.79	29.12	30.843	-0.732	L.ceriotuberosa
LC24-10-11	24.88	29.17	29.987	-1.130	L.ceriotuberosa
LC24-11-12	24.96	29.22	29.300	-1.452	L.ceriotuberosa
LC25-1-2	25.13	29.33	30.373	-1.047	L.ceriotuberosa
LC25-3-4	25.29	29.43	31.045	-1.136	L.ceriotuberosa
LC25-5-7.5	25.52	29.58	30.973	-1.405	L.ceriotuberosa
C-26-0-2	26.08		30.542	1.184	Unknown carbonate
C-26-2-4	26.25		31.309	1.629	Unknown carbonate
LC27-0-2.5	27.10	30.58	30.611	-1.150	L.ceriotuberosa
LC27-2.5-5	27.31	30.71	30.458	-1.135	L.ceriotuberosa

CORE SA3

LC26-0-4	26.17	29.78	30.009	-2.052	L.ceriotuberosa
LC26-4-6	26.42	29.94	29.709	-1.722	L.ceriotuberosa
C-26-6-8	26.58		30.746	1.084	Unknown carbonate
C-26-8-10	26.75		30.782	1.144	Unknown carbonate
C-26-10-12	26.92		29.860	1.304	Unknown carbonate
C-33-8-10	33.75		28.130	-1.297	Unknown carbonate
C-34-0-2	34.08		29.298	-0.933	Unknown carbonate
LC40-5-6	40.46	34.76	29.388	-2.783	L.ceriotuberosa
LC40-6-7	40.54	34.81	30.800	-2.243	L.ceriotuberosa
LC40-7-8	40.63	34.87	30.261	-2.659	L.ceriotuberosa
LC40-8-10	40.75	34.95	28.888	-2.198	L.ceriotuberosa
LC40-10-12	40.92	35.06	30.802	-2.576	L.ceriotuberosa
LC41-0-2	41.08	35.16	27.852	-3.453	L.ceriotuberosa
LC41-2-4	41.25	35.26	28.107	-3.432	L.ceriotuberosa
LC41-4-6	41.42	35.37	27.891	-4.927	L.ceriotuberosa

Appendix M

Transient Isotopic Model for Paleo-Owens River System

```
      PROGRAM           TRANSISO

*******************************************************************************
*******************************************************************************
* THIS PROGRAM CALCULATES CHANGES IN LAKE VOLUME AND ISOTOPIC COMPOSITION     *
* WITH RESPECT TO TIME.                                                       *
*******************************************************************************
*******************************************************************************

*******************************************************************************
*       A PROGRAM TO CALL THE RUNGE-KUTTA-FEHLBERG ORDER 4 ROUTINE TO SOLVE   *
*       A SYSTEM OF PARTIAL DIFFERENTIAL EQUATION OF THE FORM:  F(T,X)= X'    *
*******************************************************************************


      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION X(3),TOL(3),X_CS(4),TOL_CS(4)

      LOGICAL GUESS,GRAF,CPARAM,GUESS2,FOUND,SU

      PARAMETER(NUMA=25000,NUMB=2000)

      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S

      COMMON/BOTH/CL(NUMA),TSOD(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

      COMMON/BOTH2/CL_C,TSOD_C(10),TCL_C(10),TIME,TCO3_C(10),
     + CQO(10),GRAF,CONC_CL_C,DOLTEMP_C,DELDOL_C,
     + TCDELI,TCTEMP,CDEL_OUT,CCL_OUT,CL_S,NOPTS,
     + TSOD_S(10),TCL_S(10),TCO3_S(10),SQO(10),CONC_CL_S,
     + DOLTEMP_S,DELDOL_S,TSDELI,TSTEMP,SCL_DEP,TTLCL_IN,NPTS,
     + SUM_SCL_DEP,QI_C,QI_S,PQI

      COMMON/PREV/PCL_P,PCONC_CL_P,PSUM_PCL_DEP,PCL_C,PCONC_CL_C,
     + PCDEL_OUT,PCL_S,PCONC_CL_S,PSUM_SCL_DEP,PALLAREA,DATSAV,
     + SDC,SDP1,SDP2

      COMMON/HIST/QIHIST(1000),HUMTIME(500),NHPTS,SU,SUT(15),
     + SAVETIME,NUMST,QITIME(1000),NQPTS

      COMMON/SEARCH/IFIRST,ILAST,HIFIRST,HILAST

      COMMON/INFLOW/INCHOICE,A,B,C

      REAL START,FINISH,HALF

      INCHOICE=0
      A=0.D0
      B=0.D0
      C=0.D0
```

```
      OPEN(UNIT=98,FILE='OWENS.INP',STATUS='OLD')
      WRITE(6,*)
      WRITE(6,*)'READING OWENS LAKE STARTING PARAMETERS'
      READ(98,*)ITMAX,N,(X(I),I=1,N),DTMAX,DTMIN,(TOL(I),I=1,N),
     +     CONC_CL(1)

      OPEN(UNIT=89,FILE='C_S.INP',STATUS='OLD')
      WRITE(6,*)
      WRITE(6,*)'READING CHINA & SEARLES LAKE STARTING PARAMETERS'
      READ(89,*)ITMAX_CS,NCS,(X_CS(I),I=1,NCS),DTMAX_CS,DTMIN_CS,
     +     (TOL_CS(I),I=1,NCS),PCONC_CL_C,PCONC_CL_S

      OPEN(UNIT=69,FILE='P_D.INP',STATUS='OLD')
      WRITE(6,*)
      WRITE(6,*)'READING PANAMINT AND DEATH VALLEY STUFF'
      READ(69,*)AREA_P,PCONC_CL_P,PSUM_PCL_DEP,AREA_D

******************************************
** A CHANCE TO ADJUST INPUT PARAMETERS **
******************************************

      WRITE(6,*)
      WRITE(6,*)'THE CURRENT OWENS LAKE PARAMETERS ARE:'
      WRITE(6,*)'   1 : MAX ITERATIONS =',ITMAX
      WRITE(6,*)'   2 : LAKE VOL =',X(1)
      WRITE(6,*)'   3 : DEL O-18 DOLOMITE =',X(2)
      WRITE(6,*)'   4 : MAX TIME STEP =',DTMAX
      WRITE(6,*)'   5 : MIN TIME STEP =',DTMIN
      WRITE(6,*)'   6 : LAKE VOL TOLERANCE=',TOL(1)
      WRITE(6,*)'   7 : O-18 TOLERANCE=',TOL(2)
      WRITE(6,*)'   8 : CHLORIDE CONC=',CONC_CL(1)

      WRITE(6,*)
      WRITE(6,*)'CHANGE ANY OF THE STARTING PARAMETERS ?'
      WRITE(6,*)'1=YES    0=NO'
      READ(5,*)ISEE
      WRITE(6,*)

      IF(ISEE .EQ. 1)THEN

         WRITE(6,*)
         WRITE(6,*)'HOW MANY PARAMETERS WOULD YOU LIKE TO CHANGE ?'
         READ(5,*)K
         WRITE(6,*)

         DO 250 I = 1,K
            WRITE(6,*)
300         WRITE(6,*)'ENTER THE PARAMETER NUMBER'
            WRITE(6,*)
            WRITE(6,*)'   1 : MAX ITERATIONS ='
            WRITE(6,*)'   2 : LAKE VOL ='
            WRITE(6,*)'   3 : DEL O-18 DOLOMITE ='
            WRITE(6,*)'   4 : MAX TIME STEP ='
            WRITE(6,*)'   5 : MIN TIME STEP ='
            WRITE(6,*)'   6 : LAKE VOL TOLERANCE='
            WRITE(6,*)'   7 : O-18 TOLERANCE='
            WRITE(6,*)'   8 : CHLORIDE CONC='
            READ(5,*)NUMP
            IF(NUMP .EQ. 1)THEN
               WRITE(6,*)
               WRITE(6,*)'MAX ITERATIONS :',ITMAX
               WRITE(6,*)'ENTER NEW VALUE'
               READ(5,*)ITMAX
            ELSE IF(NUMP .EQ. 2)THEN
               WRITE(6,*)
               WRITE(6,'(A,D15.7)')' LAKE VOLUME :',X(1)
```

```
              WRITE(6,*)'NOTE: MAX LAKE VOL IS 30.02D9 (M^3)'
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)X(1)
           ELSE IF(NUMP .EQ. 3)THEN
              WRITE(6,*)
              WRITE(6,*)'DEL O-18 DOLOMITE :',X(2)
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)X(2)
           ELSE IF(NUMP .EQ. 4)THEN
              WRITE(6,*)
              WRITE(6,*)'MAX TIME STEP :',DTMAX
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)DTMAX
           ELSE IF(NUMP .EQ. 5)THEN
              WRITE(6,*)
              WRITE(6,*)'MIN TIME STEP :',DTMIN
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)DTMIN
           ELSE IF(NUMP .EQ. 6)THEN
              WRITE(6,*)
              WRITE(6,*)'LAKE VOL TOLERANCE :',TOL(1)
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)TOL(1)
           ELSE IF(NUMP .EQ. 7)THEN
              WRITE(6,*)
              WRITE(6,*)'DEL O-18 TOLERANCE :',TOL(2)
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)TOL(2)
           ELSE IF(NUMP .EQ. 8)THEN
              WRITE(6,*)
              WRITE(6,*)'CHLORIDE CONC :',CONC_CL(1)
              WRITE(6,*)'NOTE: 6.1 IS SATURATION'
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)CONC_CL(1)
           ELSE
              WRITE(6,*)
              WRITE(6,*)'YOU SUFFER FROM CALCULATOR DEPENDENCY'
              WRITE(6,*)'PICK A NUMBER BETWEEN 1 AND 8'
              WRITE(6,*)
              GO TO 300
           ENDIF
250     CONTINUE

************************************
** WRITE NEW VALUES TO INPUT FILE **
************************************

          REWIND 98
          WRITE(98,*)ITMAX,N,(X(I),I=1,N),DTMAX,DTMIN,(TOL(I),
     +         I=1,N),CONC_CL(1)
          CLOSE(UNIT=98)
       ENDIF

*****************************************************************
** A       VARIABLE USED TO WRITE STARTING VALUES TO FILES **
*****************************************************************

       GUESS=.TRUE.

****************************************************
** CALCULATE MOLES OF CHLORIDE FROM CONC AND VOL **
****************************************************

       IF(X(1) .LE. 0.D0)THEN
          CL(1)=0.D0
          CONC_CL(1)=0.D0
```

```
        ELSE
           CL(1)=CONC_CL(1)*(X(1)*1.D3)
        ENDIF


*************************************************************
** A CHANCE TO ADJUST INPUT PARAMETERS FOR CHINA LAKE **
*************************************************************

        WRITE(6,*)
        WRITE(6,*)'THE CURRENT CHINA LAKE PARAMETERS ARE:'
        WRITE(6,*)'   1 : MAX ITERATIONS =',ITMAX_CS
        WRITE(6,*)'   2 : LAKE VOL =',X_CS(1)
        WRITE(6,*)'   3 : DEL O-18 DOLOMITE =',X_CS(2)
        WRITE(6,*)'   4 : MAX TIME STEP =',DTMAX_CS
        WRITE(6,*)'   5 : MIN TIME STEP =',DTMIN_CS
        WRITE(6,*)'   6 : LAKE VOL TOLERANCE=',TOL_CS(1)
        WRITE(6,*)'   7 : O-18 TOLERANCE=',TOL_CS(2)
        WRITE(6,*)'   8 : CHLORIDE CONC=',PCONC_CL_C

        WRITE(6,*)
        WRITE(6,*)'CHANGE ANY OF THE STARTING PARAMETERS ?'
        WRITE(6,*)'1=YES    0=NO'
        READ(5,*)ISEE
        WRITE(6,*)

        IF(ISEE .EQ. 1)THEN

          WRITE(6,*)
          WRITE(6,*)'HOW MANY PARAMETERS WOULD YOU LIKE TO CHANGE ?'
          READ(5,*)K
          WRITE(6,*)

          DO 350 I = 1,K
             WRITE(6,*)
360          WRITE(6,*)'ENTER THE PARAMETER NUMBER'
             WRITE(6,*)
             WRITE(6,*)'   1 : MAX ITERATIONS ='
             WRITE(6,*)'   2 : LAKE VOL ='
             WRITE(6,*)'   3 : DEL O-18 DOLOMITE ='
             WRITE(6,*)'   4 : MAX TIME STEP ='
             WRITE(6,*)'   5 : MIN TIME STEP ='
             WRITE(6,*)'   6 : LAKE VOL TOLERANCE='
             WRITE(6,*)'   7 : O-18 TOLERANCE='
             WRITE(6,*)'   8 : CHLORIDE CONC='
             READ(5,*)NUMP
             IF(NUMP .EQ. 1)THEN
                WRITE(6,*)
                WRITE(6,*)'MAX ITERATIONS :',ITMAX_CS
                WRITE(6,*)'ENTER NEW VALUE'
                READ(5,*)ITMAX_CS
             ELSE IF(NUMP .EQ. 2)THEN
                WRITE(6,*)
                WRITE(6,'(A,D15.7)')' LAKE VOLUME :',X_CS(1)
                WRITE(6,*)'NOTE: MAX LAKE VOL IS 0.69609 (M^3)'
                WRITE(6,*)'ENTER NEW VALUE'
                READ(5,*)X_CS(1)
             ELSE IF(NUMP .EQ. 3)THEN
                WRITE(6,*)
                WRITE(6,*)'DEL O-18 DOLOMITE :',X_CS(2)
                WRITE(6,*)'ENTER NEW VALUE'
                READ(5,*)X_CS(2)
             ELSE IF(NUMP .EQ. 4)THEN
                WRITE(6,*)
                WRITE(6,*)'MAX TIME STEP :',DTMAX_CS
                WRITE(6,*)'ENTER NEW VALUE'
```

```
              READ(5,*)DTMAX_CS
           ELSE IF(NUMP .EQ. 5)THEN
              WRITE(6,*)
              WRITE(6,*)'MIN TIME STEP :',DTMIN_CS
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)DTMIN_CS
           ELSE IF(NUMP .EQ. 6)THEN
              WRITE(6,*)
              WRITE(6,*)'LAKE VOL TOLERANCE :',TOL_CS(1)
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)TOL_CS(1)
           ELSE IF(NUMP .EQ. 7)THEN
              WRITE(6,*)
              WRITE(6,*)'DEL O-18 TOLERANCE :',TOL_CS(2)
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)TOL_CS(2)
           ELSE IF(NUMP .EQ. 8)THEN
              WRITE(6,*)
              WRITE(6,*)'CHLORIDE CONC :',PCONC_CL_C
              WRITE(6,*)'NOTE: 6.1 IS SATURATION'
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)PCONC_CL_C
           ELSE
              WRITE(6,*)
              WRITE(6,*)'OBVIOUSLY MATH IS NOT YOUR FORTE'
              WRITE(6,*)'PICK A NUMBER BETWEEN 1 AND 8'
              WRITE(6,*)
              GO TO 360
           ENDIF
350       CONTINUE
        ENDIF

***************************************************
** CALCULATE MOLES OF CHLORIDE FROM CONC AND VOL **
***************************************************

        IF(X_CS(1) .LE. 0.D0)THEN
          PCL_C=0.D0
          PCONC_CL_C=0.D0
        ELSE
          PCL_C=PCONC_CL_C*(X_CS(1)*1.D3)
        ENDIF


**********************************************************
** A CHANCE TO ADJUST INPUT PARAMETERS FOR SEARLES LAKE **
**********************************************************

        WRITE(6,*)
        WRITE(6,*)'THE CURRENT SEARLES LAKE PARAMETERS ARE:'
        WRITE(6,*)'   1 : MAX ITERATIONS =',ITMAX_CS
        WRITE(6,*)'   2 : LAKE VOL =',X_CS(3)
        WRITE(6,*)'   3 : DEL O-18 DOLOMITE =',X_CS(4)
        WRITE(6,*)'   4 : MAX TIME STEP =',DTMAX_CS
        WRITE(6,*)'   5 : MIN TIME STEP =',DTMIN_CS
        WRITE(6,*)'   6 : LAKE VOL TOLERANCE=',TOL_CS(3)
        WRITE(6,*)'   7 : O-18 TOLERANCE=',TOL_CS(4)
        WRITE(6,*)'   8 : CHLORIDE CONC=',PCONC_CL_S

        WRITE(6,*)
        WRITE(6,*)'CHANGE ANY OF THE STARTING PARAMETERS ?'
        WRITE(6,*)'1=YES    0=NO'
        READ(5,*)ISEE
        WRITE(6,*)

        IF(ISEE .EQ. 1)THEN
```

```
      WRITE(6,*)
      WRITE(6,*)'HOW MANY PARAMETERS WOULD YOU LIKE TO CHANGE ?'
      READ(5,*)K
      WRITE(6,*)

      DO 370 I = 1,K
         WRITE(6,*)
380      WRITE(6,*)'ENTER THE PARAMETER NUMBER'
         WRITE(6,*)
         WRITE(6,*)'   1 : MAX ITERATIONS ='
         WRITE(6,*)'   2 : LAKE VOL ='
         WRITE(6,*)'   3 : DEL O-18 DOLOMITE='
         WRITE(6,*)'   4 : MAX TIME STEP ='
         WRITE(6,*)'   5 : MIN TIME STEP ='
         WRITE(6,*)'   6 : LAKE VOL TOLERANCE='
         WRITE(6,*)'   7 : O-18 TOLERANCE='
         WRITE(6,*)'   8 : CHLORIDE CONC='
         READ(5,*)NUMP
         IF(NUMP .EQ. 1)THEN
            WRITE(6,*)
            WRITE(6,*)'MAX ITERATIONS :',ITMAX_CS
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)ITMAX_CS
         ELSE IF(NUMP .EQ. 2)THEN
            WRITE(6,*)
            WRITE(6,'(A,D15.7)')' LAKE VOLUME :',X_CS(3)
            WRITE(6,*)'NOTE: MAX LAKE VOL IS 85.2809 (M^3)'
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)X_CS(3)
         ELSE IF(NUMP .EQ. 3)THEN
            WRITE(6,*)
            WRITE(6,*)'DEL O-18 DOLOMITE :',X_CS(4)
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)X_CS(4)
         ELSE IF(NUMP .EQ. 4)THEN
            WRITE(6,*)
            WRITE(6,*)'MAX TIME STEP :',DTMAX_CS
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)DTMAX_CS
         ELSE IF(NUMP .EQ. 5)THEN
            WRITE(6,*)
            WRITE(6,*)'MIN TIME STEP :',DTMIN_CS
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)DTMIN_CS
         ELSE IF(NUMP .EQ. 6)THEN
            WRITE(6,*)
            WRITE(6,*)'LAKE VOL TOLERANCE :',TOL_CS(3)
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)TOL_CS(3)
         ELSE IF(NUMP .EQ. 7)THEN
            WRITE(6,*)
            WRITE(6,*)'DEL O-18 TOLERANCE :',TOL_CS(4)
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)TOL_CS(4)
         ELSE IF(NUMP .EQ. 8)THEN
            WRITE(6,*)
            WRITE(6,*)'CHLORIDE CONC :',PCONC_CL_S
            WRITE(6,*)'NOTE: 6.1 IS SATURATION'
            WRITE(6,*)'ENTER NEW VALUE'
            READ(5,*)PCONC_CL_S
         ELSE
            WRITE(6,*)
            WRITE(6,*)'GET A CLUE KELP BREATH'
            WRITE(6,*)'PICK A NUMBER BETWEEN 1 AND 8'
            WRITE(6,*)
```

```
            GO TO 380
          ENDIF
370     CONTINUE
      ENDIF

************************************
** WRITE NEW VALUES TO INPUT FILE **
************************************

      REWIND 89
      WRITE(89,*)ITMAX_CS,NCS,(X_CS(I),I=1,NCS),DTMAX_CS,
     +     DTMIN_CS,(TOL_CS(I),I=1,NCS),PCONC_CL_C,
     +     PCONC_CL_S
      CLOSE(UNIT=89)

***************************************************
** CALCULATE MOLES OF CHLORIDE FROM CONC AND VOL **
***************************************************

      IF(X_CS(3) .LE. 0.D0)THEN
         PCL_S=0.D0
         PCONC_CL_S=0.D0
      ELSE
         PCL_S=PCONC_CL_S*(X_CS(3)*1.D3)
      ENDIF

*****************************************************************
** TOTAL MASS OF CHLORIDE DEPOSITED IN SEARLES AT START OF RUN **
*****************************************************************

      WRITE(6,*)
      WRITE(6,*)'ENTER THE MASS OF CL DEPOSITED IN SEARLES LAKE'
      WRITE(6,*)'AT THE BEGINNING OF THIS RUN (KG/M^2)'
      WRITE(6,*)
      READ(5,*)PSUM_SCL_DEP

***********************************************************************
** A CHANCE TO ADJUST INPUT PARAMETERS FOR PANAMINT AND DEATH VALLEY **
***********************************************************************

      WRITE(6,*)
      WRITE(6,*)'THE CURRENT PANAMINT LAKE PARAMETERS ARE:'
      WRITE(6,*)'   1 : SURFACE AREA =',AREA_P
      WRITE(6,*)'   2 : CHLORIDE CONC=',PCONC_CL_P
      WRITE(6,*)'   3 : CHLORIDE DEPOSITED (KG/M^2)',PSUM_PCL_DEP

      WRITE(6,*)
      WRITE(6,*)'CHANGE ANY OF THE STARTING PARAMETERS ?'
      WRITE(6,*)'1=YES    0=NO'
      READ(5,*)ISEE
      WRITE(6,*)

      IF(ISEE .EQ. 1)THEN

         WRITE(6,*)
         WRITE(6,*)'HOW MANY PARAMETERS WOULD YOU LIKE TO CHANGE ?'
         READ(5,*)K
         WRITE(6,*)

         DO 400 I = 1,K
            WRITE(6,*)
410         WRITE(6,*)'ENTER THE PARAMETER NUMBER'
            WRITE(6,*)
            WRITE(6,*)'   1 : SURFACE AREA'
            WRITE(6,*)'   2 : CHLORIDE CONC'
            WRITE(6,*)'   3 : CHLORIDE DEPOSITED (KG/M^2)'
```

```
           READ(5,*)NUMP
           IF(NUMP .EQ. 1)THEN
              WRITE(6,*)
              WRITE(6,'(A,D15.7)')' SURFACE AREA =',AREA_P
              WRITE(6,*)'NOTE: MAX SURFACE AREA IS 0.727D9'
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)AREA_P
           ELSE IF(NUMP .EQ. 2)THEN
              WRITE(6,*)
              WRITE(6,*)'CHLORIDE CONC :',PCONC_CL_P
              WRITE(6,*)'NOTE: 6.1 IS SATURATION'
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)PCONC_CL_P
           ELSE IF(NUMP .EQ. 3)THEN
              WRITE(6,*)
              WRITE(6,*)'SUM OF CHLORIDE DEPOSITED',PSUM_PCL_DEP
              WRITE(6,*)'ENTER NEW VALUE'
              READ(5,*)PSUM_PCL_DEP
           ELSE
              WRITE(6,*)
              WRITE(6,*)'EVERY DAY MUST BE MONDAY FOR SOMEONE LIKE
     +YOU'
              WRITE(6,*)'PICK A NUMBER BETWEEN 1 AND 8'
              WRITE(6,*)
              GO TO 410
           ENDIF
400     CONTINUE
        ENDIF

*****************************************************
** CALCULATE MOLES OF CHLORIDE FROM CONC AND VOL **
*****************************************************

        IF(PCONC_CL_P .LE. 0.D0)THEN
           PCL_P=0.D0
        ELSE
           VOL_P=PVOL(AREA_P)
           PCL_P=PCONC_CL_P/(VOL_P*1.D3)
        ENDIF

************************
** DEATH VALLEY STUFF **
************************

        WRITE(6,*)
        WRITE(6,*)'THE CURRENT DEATH VALLEY SURFACE AREA IS:'
        WRITE(6,*)'   1 : SURFACE AREA =',AREA_D

        WRITE(6,*)
        WRITE(6,*)'WOULD YOU LIKE TO CHANGE THIS ?'
        WRITE(6,*)'1=YES    0=NO'
        READ(5,*)ISEE
        WRITE(6,*)

        IF(ISEE .EQ. 1)THEN

           WRITE(6,*)
           WRITE(6,'(A,D15.7)')' SURFACE AREA =',AREA_D
           WRITE(6,*)'NOTE: MAX SURFACE AREA IS 0.583D9'
           WRITE(6,*)'ENTER NEW VALUE'
           READ(5,*)AREA_D

        ENDIF

************************************
** WRITE NEW VALUES TO INPUT FILE **
```

```
*************************************

       REWIND 69
       WRITE(69,*)AREA_P,PCONC_CL_P,PSUM_PCL_DEP,AREA_D
       CLOSE(UNIT=69)

************************************************************************************
*              LET'S CHOOSE AN INFLOW FUNCTION AND TIME INTERVAL                  *
************************************************************************************


       WRITE(6,*)
       WRITE(6,*)
115    WRITE(6,*)'WHICH INFLOW FUNCTION WOULD YOU LIKE FOR OWENS LAKE?'
       WRITE(6,*)'1=LINEAR'
       WRITE(6,*)'2=EXPONENTIAL'
       WRITE(6,*)'3=LOGARITHMIC'
       WRITE(6,*)'4=POWER'
       WRITE(6,*)'5=SINUSOIDAL'
       WRITE(6,*)'6=STEP'
       WRITE(6,*)'7=ZERO INFLOW'
       WRITE(6,*)'8=STEADY-STATE HISTORY, VARIABLE TEMP'
       WRITE(6,*)'9=STEADY-STATE HISTORY, CONSTANT HIGH TEMP'
       WRITE(6,*)'10=STEADY-STATE HISTORY, CONSTANT LOW TEMP'
       READ(5,*)INCHOICE

       WRITE(6,*)
       WRITE(6,*)
       IF(INCHOICE .EQ. 1)THEN
          WRITE(6,*)'YOU HAVE CHOSEN F(QI)= A*X+B'
          WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
       ELSE IF(INCHOICE .EQ. 2)THEN
          WRITE(6,*)'YOU HAVE CHOSEN F(QI)= B*EXP(A*X)'
          WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
       ELSE IF(INCHOICE .EQ. 3)THEN
          WRITE(6,*)'YOU HAVE CHOSEN F(QI)= B+A*LOG(X)'
          WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
       ELSE IF(INCHOICE .EQ. 4)THEN
          WRITE(6,*)'YOU HAVE CHOSEN F(QI)= B+A*(X**C)'
          WRITE(6,*)'ENTER VALUES FOR ''A'',''B'', AND ''C'''
       ELSE IF(INCHOICE .EQ. 5)THEN
          WRITE(6,*)'YOU HAVE CHOSEN F(QI)= B+A*SIN(C*X)'
          WRITE(6,*)'ENTER VALUES FOR ''A'',''B'', AND ''C'''
       ELSE IF(INCHOICE .EQ. 6)THEN
          WRITE(6,*)'YOU HAVE CHOSEN F(QI)= B+(A*B)'
          WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
       ELSE IF(INCHOICE .EQ. 7)THEN
          WRITE(6,*)'YOU HAVE CHOSEN ZERO INFLOW'
          WRITE(6,*)'GRAB YOUR CANTEEN AND HEAD FOR THE SHADE'
       ELSE IF(INCHOICE .EQ. 8)THEN
          WRITE(6,*)'YOU HAVE CHOSEN STEADY STATE "GUESS"'
          WRITE(6,*)'VARIABLE TEMPERATURE HISTORY'
       ELSE IF(INCHOICE .EQ. 9)THEN
          WRITE(6,*)'YOU HAVE CHOSEN STEADY STATE "GUESS"'
          WRITE(6,*)'CONSTANT HIGH TEMPERATURE HISTORY'
       ELSE IF(INCHOICE .EQ. 10)THEN
          WRITE(6,*)'YOU HAVE CHOSEN STEADY STATE "GUESS"'
          WRITE(6,*)'CONSTANT LOW TEMPERATURE HISTORY'
       ELSE
          WRITE(6,*)
          WRITE(6,*)
          WRITE(6,*)'NOT A VALID CHOICE MULLET-HEAD'
          WRITE(6,*)
          WRITE(6,*)
          GOTO 115
       ENDIF
```

```
      IF(INCHOICE .EQ. 8)THEN

         WRITE(6,*)'READING INFLOW HISTORY'
         OPEN(UNIT=29,FILE='CASEA1.DAT',STATUS='OLD')
         DO 450 I=1,1000
            READ(29,*,END=451)QITIME(I),F2,QIHIST(I)
            QITIME(I)=QITIME(I)*1.D6
450      CONTINUE
451      WRITE(6,*)'READ',I-1,' POINTS FROM INFLOW HISTORY'
         NQPTS=I-1

      ELSEIF(INCHOICE .EQ. 9)THEN

         WRITE(6,*)'READING INFLOW HISTORY'
         OPEN(UNIT=28,FILE='CASEB1.DAT',STATUS='OLD')
         DO 460 I=1,1000
            READ(28,*,END=461)QITIME(I),F2,QIHIST(I)
            QITIME(I)=QITIME(I)*1.D6
460      CONTINUE
461      WRITE(6,*)'READ',I-1,' POINTS FROM INFLOW HISTORY'
         NQPTS=I-1

      ELSEIF(INCHOICE .EQ. 10)THEN

         WRITE(6,*)'READING INFLOW HISTORY'
         OPEN(UNIT=27,FILE='CASEC1.DAT',STATUS='OLD')
         DO 470 I=1,1000
            READ(27,*,END=471)QITIME(I),F2,QIHIST(I)
            QITIME(I)=QITIME(I)*1.D6
470      CONTINUE
471      WRITE(6,*)'READ',I-1,' POINTS FROM INFLOW HISTORY'
         NQPTS=I-1

      ELSEIF(INCHOICE .EQ. 4 .OR. INCHOICE .EQ. 5)THEN
         READ(5,*)A,B,C
      ELSE IF(INCHOICE .NE. 7)THEN
         READ(5,*)A,B
      ENDIF

      WRITE(6,*)
      WRITE(6,*)'ENTER STARTING TIME AND ENDING TIME'
      WRITE(6,*)'0 CORRESPONDS TO PRESENT, 2.0 IS 2 MILLION YRS AGO'
      WRITE(6,*)
      WRITE(6,*)
      WRITE(6,*)'MANAGEMENT ACCEPTS NO RESPONSIBILITY FOR PEOPLE WHO'
      WRITE(6,*)'RUN THE MODEL BACKWARD IN TIME'
      WRITE(6,*)
      WRITE(6,*)
      READ(5,*)TBEG, TEND
      TBEG=TBEG*1.D6
      TEND=TEND*1.D6
      WRITE(6,*)

***********************************************************************
** TELL THE MODEL HOW MUCH TIME TO PUT BETWEEN SAVED DATA POINTS.    **
** THIS MAKES THE MODEL RUN MUCH FASTER AND REDUCES THE SIZE OF THE **
** DATA FILES.                                                       **
***********************************************************************

      WRITE(6,*)
      WRITE(6,*)'ENTER THE MINIMUM TIME BETWEEN SAVED DATA POINTS'
      WRITE(6,*)
      READ(5,*)DATSAV

************************************************************************
```

```
** SET "PEAK AND VALLEY" DETECTORS SO DATSAV SPACING DOESN'T MISS ANY **
** MAXIMA OR MINIMA                                                  **
***********************************************************************

      SDC=X_CS(4)
      SDP1=X_CS(4)
      SDP2=X_CS(4)

*********************************
** CONSTANT PARAMETER OPTION **
*********************************

      CPARAM=.FALSE.
      WRITE(6,*)
      WRITE(6,*)'DO YOU WANT TO RUN THE PROGRAM WITH CONSTANT PARAMETE
     +RS ?'
      WRITE(6,*)'1=YES   0=NO'
      WRITE(6,*)
      READ(5,*)INPARAM

      IF(INPARAM .EQ. 1)THEN
         CPARAM=.TRUE.

 490     WRITE(6,*)
         WRITE(6,*)'WOULD YOU LIKE UPPER LIMIT, LOWER LIMIT, OR CUSTOM
     + '
         WRITE(6,*)'1 = UPPER LIMIT'
         WRITE(6,*)'2 = LOWER LIMIT'
         WRITE(6,*)'3 = CUSTOM'
         WRITE(6,*)
         READ(5,*)LIMCHOICE

         IF(LIMCHOICE .EQ. 1)THEN
            OPEN(UNIT=68,FILE='UPPER.INP',STATUS='OLD')
            WRITE(6,*)'READING CONSTANT PARAMETERS'
            WRITE(6,*)
            READ(68,*)TEMPC,TEMPC_C,TEMPC_S,
     +           PRECIPC,PRECIPC_C,PRECIPC_S,EVAPC,EVAPC_C,EVAPC_S,
     +           EVAPC_P,EVAPC_D,DELAC,DELAC_C,DELAC_S,DELIC,DELPC,
     +           DELPC_C,DELPC_S
            CLOSE(UNIT=68)
         ELSE IF(LIMCHOICE .EQ. 2)THEN
            OPEN(UNIT=67,FILE='LOWER.INP',STATUS='OLD')
            WRITE(6,*)'READING CONSTANT PARAMETERS'
            WRITE(6,*)
            READ(67,*)TEMPC,TEMPC_C,TEMPC_S,
     +           PRECIPC,PRECIPC_C,PRECIPC_S,EVAPC,EVAPC_C,EVAPC_S,
     +           EVAPC_P,EVAPC_D,DELAC,DELAC_C,DELAC_S,DELIC,DELPC,
     +           DELPC_C,DELPC_S
            CLOSE(UNIT=67)
         ELSE IF(LIMCHOICE .EQ. 3)THEN

****************************
** PICK A TEMP, ANY TEMP **
****************************

            WRITE(6,*)
            WRITE(6,*)'ENTER THE TEMP FOR OWENS (DEGREES C)'
            WRITE(6,*)
            READ(5,*)TEMPC
            WRITE(6,*)

            WRITE(6,*)
            WRITE(6,*)'ENTER THE TEMP FOR CHINA (DEGREES C)'
            WRITE(6,*)
            READ(5,*)TEMPC_C
```

```
      WRITE(6,*)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE TEMP FOR SEARLES (DEGREES C)'
      WRITE(6,*)
      READ(5,*)TEMPC_S
      WRITE(6,*)

******************************
** CONSTANT PRECIPITATION **
******************************

      WRITE(6,*)
      WRITE(6,*)'ENTER THE PRECIPITATION FOR OWENS(METERS/YR)'
      WRITE(6,*)
      READ(5,*)PRECIPC
      WRITE(6,*)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE PRECIPITATION FOR CHINA(METERS/YR)'
      WRITE(6,*)
      READ(5,*)PRECIPC_C
      WRITE(6,*)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE PRECIPITATION FOR SEARLES(METERS/YR)'
      WRITE(6,*)
      READ(5,*)PRECIPC_S
      WRITE(6,*)

****************************
** CONSTANT EVAPORATION **
****************************

      WRITE(6,*)
      WRITE(6,*)'ENTER THE EVAPORATION FOR OWENS(METERS/YR)'
      WRITE(6,*)
      READ(5,*)EVAPC
      WRITE(6,*)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE EVAPORATION FOR CHINA(METERS/YR)'
      WRITE(6,*)
      READ(5,*)EVAPC_C
      WRITE(6,*)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE EVAPORATION FOR SEARLES(METERS/YR)'
      WRITE(6,*)
      READ(5,*)EVAPC_S
      WRITE(6,*)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE EVAPORATION FOR PANAMINT(METERS/YR)'
      WRITE(6,*)
      READ(5,*)EVAPC_P
      WRITE(6,*)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE EVAPORATION FOR DEATH VALLEY(METERS/Y
     +R)'
      WRITE(6,*)
      READ(5,*)EVAPC_D
      WRITE(6,*)

*******************************
```

```
** CONSTANT DEL ATMOSPHERE **
******************************

            WRITE(6,*)
            WRITE(6,*)'ENTER DEL ATMOSPHERE FOR OWENS (PER MIL)'
            WRITE(6,*)
            READ(5,*)DELAC
            WRITE(6,*)

            WRITE(6,*)
            WRITE(6,*)'ENTER DEL ATMOSPHERE FOR CHINA (PER MIL)'
            WRITE(6,*)
            READ(5,*)DELAC_C
            WRITE(6,*)

            WRITE(6,*)
            WRITE(6,*)'ENTER DEL ATMOSPHERE FOR SEARLES (PER MIL)'
            WRITE(6,*)
            READ(5,*)DELAC_S
            WRITE(6,*)

***************************
** CONSTANT DEL INFLOW  **
***************************

            WRITE(6,*)
            WRITE(6,*)'ENTER DEL INFLOW FOR OWENS (PER MIL)'
            WRITE(6,*)
            READ(5,*)DELIC
            WRITE(6,*)

***************************
** CONSTANT DEL PRECIP  **
***************************

            WRITE(6,*)
            WRITE(6,*)'ENTER DEL PRECIPITATION FOR OWENS (PER MIL)'
            WRITE(6,*)
            READ(5,*)DELPC
            WRITE(6,*)

            WRITE(6,*)
            WRITE(6,*)'ENTER DEL PRECIPITATION FOR CHINA (PER MIL)'
            WRITE(6,*)
            READ(5,*)DELPC_C
            WRITE(6,*)

            WRITE(6,*)
            WRITE(6,*)'ENTER DEL PRECIPITATION FOR SEARLES (PER MIL)'
            WRITE(6,*)
            READ(5,*)DELPC_S
            WRITE(6,*)

         ELSE
            WRITE(6,*)'NO,NO,NOOOOOOO...'
            WRITE(6,*)'PICK 1,2, OR 3 '
            WRITE(6,*)
            GO TO 490
         ENDIF


         OPEN(UNIT=80,FILE='HUMHIST.DAT',STATUS='OLD')
         DO 1750 I=1,400
            READ(80,*,END=1751)HUMTIME(I),SHUM(I)
            OHUM(I)=SHUM(I)
            CHUM(I)=SHUM(I)
```

```
1750      CONTINUE

1751      NHPTS=I-1
          WRITE(6,*)'READ',NHPTS,' FROM HUMHIST.DAT'
          CLOSE(80)

          DO 1760 I=1,NHPTS
             HUMTIME(I)=HUMTIME(I)*1.D6
1760      CONTINUE

       ENDIF

********************
** GRAPHICS STUFF **
********************

       GRAF=.FALSE.

****************************************************
** SAVE VALUES FOR MODEL STARTUP AT A GIVEN TIME **
****************************************************

       SU=.FALSE.

       WRITE(6,*)
       WRITE(6,*)'DO YOU WANT TO SAVE INFO TO RESTART THE MODEL'
       WRITE(6,*)'AT A SPECIFIC TIME'
       WRITE(6,*)' 1=YES    0=NO'
       WRITE(6,*)

       READ(5,*)ISU

       IF(ISU .EQ. 1)THEN
          SU=.TRUE.
          WRITE(6,*)
          WRITE(6,*)'HOW MANY STARTUP TIMES DO YOU WISH ?'
          WRITE(6,*)
          READ(5,*)NUMST

          DO 1900 I=1,NUMST
             WRITE(6,*)
             WRITE(6,*)'ENTER STARTUP TIME'
             READ(5,*)SUT(I)
             WRITE(6,*)
1900      CONTINUE

          SAVETIME=SUT(1)
          ISTCNT=1

       ENDIF



******************************************************************************
*    ALL WE'RE DOING HERE IS READING DATA FILES, THE GOOD STUFF IS LATER     *
******************************************************************************

       IF(.NOT. CPARAM)THEN

          WRITE(6,*)
          WRITE(6,*)
          WRITE(6,*)'READING DATA FILES ...'
          WRITE(6,*)
          WRITE(6,*)
```

```
      OPEN(UNIT=20,FILE='OWENS.UF',STATUS='OLD',
     +      FORM='UNFORMATTED')

      DO 500 I = 1, 2000
         READ (20,END=501) WTIME(I),OTEMP(I),OEVAP(I),GBG1,
     +         OPRECIP(I)
500   CONTINUE

501   NPTS=I-1

      DO 600 I = 1,NPTS
         WTIME(I)=WTIME(I)*1.0D6
600   CONTINUE

      DO 650 I=1,NPTS
         IF(WTIME(I).GT.TBEG)THEN
            ILAST=I
            GOTO 651
         ENDIF
650   CONTINUE

651   DO 660 I=1,NPTS
         IF(WTIME(I).GT.TEND)THEN
            IFIRST=I-1
            GOTO 661
         ENDIF
660   CONTINUE

661   CLOSE (UNIT=20)

      OPEN(UNIT=40,FILE='SEARLES.UF',STATUS='OLD',
     +      FORM='UNFORMATTED')

      DO 800 I = 1, NPTS
         READ (40) STEMP(I),SEVAP(I),GBG2,SPRECIP(I)
800   CONTINUE

      CLOSE (UNIT=40)

      DO 850 I=1,NPTS
         CTEMP(I)=STEMP(I)
         CEVAP(I)=SEVAP(I)
         CPRECIP(I)=SPRECIP(I)
850   CONTINUE

      OPEN(UNIT=21,FILE='ODELP.UF',STATUS='OLD',
     +      FORM='UNFORMATTED')
      DO 900 I = 1, NPTS
         READ(21)ODELP(I)
900   CONTINUE

      CLOSE(UNIT=21)

      OPEN(UNIT=41,FILE='SDELP.UF',STATUS='OLD',
     +      FORM='UNFORMATTED')
      DO 1100 I = 1, NPTS
         READ(41)SDELP(I)
1100  CONTINUE

      CLOSE(UNIT=41)

      DO 1150 I=1,NPTS
         CDELP(I)=SDELP(I)
1150  CONTINUE

      OPEN(UNIT=22,FILE='ODELA.UF',STATUS='OLD',
```

```
     +          FORM='UNFORMATTED')
          DO 1200 I = 1, NPTS
             READ(22) ODELA(I)
 1200     CONTINUE

          CLOSE(UNIT=22)

          OPEN(UNIT=32,FILE='CSDELA.UF',STATUS='OLD',
     +          FORM='UNFORMATTED')
          DO 1300 I = 1, NPTS
             READ(32) CSDELA(I)
 1300     CONTINUE

          CLOSE(UNIT=32)

          OPEN(UNIT=23,FILE='ODELI.UF',STATUS='OLD',
     +          FORM='UNFORMATTED')
          DO 1400 I = 1,NPTS
             READ(23)ODELI(I)
 1400     CONTINUE

          CLOSE(UNIT=23)

          OPEN(UNIT=85,FILE='PEVAP.UF',STATUS='OLD',
     +          FORM='UNFORMATTED')
          DO 1500 I = 1,NPTS
             READ(85)PEVAP(I)
 1500     CONTINUE

          CLOSE(UNIT=85)

          OPEN(UNIT=84,FILE='DEVAP.UF',STATUS='OLD',
     +          FORM='UNFORMATTED')
          DO 1600 I = 1,NPTS
             READ(84)DEVAP(I)
 1600     CONTINUE

          CLOSE(UNIT=84)

          OPEN(UNIT=80,FILE='HUMHIST.DAT',STATUS='OLD')
          DO 1700 I=1,400
             READ(80,*,END=1701)HUMTIME(I),SHUM(I)
             OHUM(I)=SHUM(I)
             CHUM(I)=SHUM(I)
 1700     CONTINUE

 1701     NHPTS=I-1

          CLOSE(UNIT=80)

          DO 1800 I=1,NHPTS
             HUMTIME(I)=HUMTIME(I)*1.D6
 1800     CONTINUE

       ENDIF

****************************************
** CONVERT DEL DOLOMITE TO DEL WATER **
****************************************

          IF(CPARAM)THEN
             X(2)= W_DEL(X(2),TEMPC)
             X_CS(2)=W_DEL(X_CS(2),TEMPC_C)
             X_CS(4)=W_DEL(X_CS(4),TEMPC_S)
          ELSE
```

```
          CALL FINDT(NPTS,TBEG,NDX,FOUND,WTIME)
          TOTEMP=(OTEMP(NDX+1)-OTEMP(NDX))/(WTIME(NDX+1)-
     +         WTIME(NDX))*(TBEG-WTIME(NDX))+OTEMP(NDX)
          TSTEMP=(STEMP(NDX+1)-STEMP(NDX))/(WTIME(NDX+1)-
     +         WTIME(NDX))*(TBEG-WTIME(NDX))+STEMP(NDX)
          TCTEMP=(CTEMP(NDX+1)-CTEMP(NDX))/(WTIME(NDX+1)-
     +         WTIME(NDX))*(TBEG-WTIME(NDX))+CTEMP(NDX)

          X(2)=W_DEL(X(2),TOTEMP)
          X_CS(2)=W_DEL(X_CS(2),TCTEMP)
          X_CS(4)=W_DEL(X_CS(4),TSTEMP)
        ENDIF

****************************
** START THE BALL ROLLING **
****************************



      CALL RKF(N,X,TBEG,TEND,TOL,DTMAX,DTMIN,ITMAX)


*********************************************
** CALL SOLVER FOR CHINA AND SEARLES LAKE **
*********************************************



      GUESS=.TRUE.
      GUESS2=.TRUE.

      CALL RKF_CS(NCS,X_CS,TBEG,TEND,TOL_CS,DTMAX_CS,
     +     DTMIN_CS,ITMAX_CS)



      END



      SUBROUTINE RKF(N,X,TBEG,TEND,TOL,DTMAX,DTMIN,ITMAX)
*******************************************************************************
*     SOLVE A SYSTEM OF PARTIAL DIFFERENTIAL EQUATION OF THE FORM:            *
*                       F(T,X)= X'                                            *
*      BETWEEN T1,T2, GIVEN THE INITIAL CONDITION XO(T1)                      *
*******************************************************************************
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      PARAMETER(NUMA=25000,NUMB=2000)
      PARAMETER (VOLMAX=30.02D9,QCL_IN=1.67D8)

      LOGICAL PASS,GRAF,ONLY1,ZEROVOL,ZEROCHK,SU

      SAVE

      COMMON/BOTH/CL(NUMA),TSOD(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

      COMMON/BOTH2/CL_C,TSOD_C(10),TCL_C(10),TIME,TCO3_C(10),
     + CQO(10),GRAF,CONC_CL_C,DOLTEMP_C,DELDOL_C,
     + TCDELI,TCTEMP,CDEL_OUT,CCL_OUT,CL_S,NOPTS,
     + TSOD_S(10),TCL_S(10),TCO3_S(10),SQO(10),CONC_CL_S,
     + DOLTEMP_S,DELDOL_S,TSDELI,TSTEMP,SCL_DEP,TTLCL_IN,NPTS,
```

```
     +  SUM_SCL_DEP,QI_C,QI_S,PQI

        COMMON/PREV/PCL_P,PCONC_CL_P,PSUM_PCL_DEP,PCL_C,PCONC_CL_C,
     +  PCDEL_OUT,PCL_S,PCONC_CL_S,PSUM_SCL_DEP,PALLAREA,DATSAV,
     +  SDC,SDP1,SDP2

        COMMON/HIST/QIHIST(1000),HUMTIME(500),NHPTS,SU,SUT(15),
     +  SAVETIME,NUMST,QITIME(1000),NQPTS

        COMMON/FINAL/FINDEL,FINVOL,FINAREA,FINCL

        COMMON/INFLOW/INCHOICE,A,B,C

        DIMENSION X(3),RK1(3),RK2(3),RK3(3),RK4(3),RK5(3),RK6(3),R(3)
        DIMENSION TERM(3),DEL(3),TOL(3)

**      OPEN(UNIT=95,FILE='QI.OUT',STATUS='UNKNOWN')
        OPEN(UNIT=70,FILE='START.OUT',STATUS='UNKNOWN')

        STEP=DTMAX
        KOUNT=1
        OTIME(KOUNT)=TBEG
        ODEL_OUT(KOUNT)=X(2)
        OCL_OUT(KOUNT)=0.D0
        ONLY1=.TRUE.

*********************************
** TO PROTECT YOU FROM DRYNESS **
*********************************

        ZEROVOL=.FALSE.
        ZEROCHK=.FALSE.

*************************************
** THE SOLVING ROUTINE BEGINS HERE **
*************************************

*******************************************
** INITIALIZE PARAMETERS TO RESTART MODEL **
*******************************************

        SAVETIME=SUT(1)
        ISTCNT=1
        ITER=0

        WRITE(6,*)
        WRITE(6,*)
        WRITE(6,*)'SOLVING DIFFERENTIAL EQUATIONS FOR OWENS'
        WRITE(6,*)
        WRITE(6,*)

        WHILE(KOUNT .LT. NUMA)DO
           ITER=ITER+1
           IF(OTIME(KOUNT).GT.TEND)THEN
              KNT=1
              T=OTIME(KOUNT)
              DO 200 I=1,N
                 RK1(I)=STEP*F(I,X,T,KNT,KOUNT,TBEG,TEND)
200           CONTINUE

***************************************************************************
** "TERM" IS AN ARRAY WHICH STORES APPROXIMATIONS OF VOL AND DEL O-18 **
** WHICH WILL BE USED IN FINAL CALCULATIONS IF ERRORS WITHIN THE STEP **
** ARE LESS THAN THE GIVEN TOLERANCES.                                **
***************************************************************************
```

```
          T=OTIME(KOUNT)-STEP/4.D0
          KNT=2
          DO 300 I=1,N
             TERM(I)=X(I)+ RK1(I)/4.D0
300       CONTINUE

*****************************************************************************
** WHEN BOTH "IF'S" ARE SATISFIED YOU ARE AS CLOSE TO ZERO VOLUME AS THE   **
** SOLVER IS GOING TO GET WITH THE GIVEN CONSTRAINTS, SO GO TO THE END OF **
** THE RKF AND MAKE VOLUME=0 AND DEL=DELINFLOW                            **
*****************************************************************************

          IF(TERM(1) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN).LT.1.D-6)THEN
                ZEROVOL=.TRUE.
                PASS=.TRUE.
                GOTO 1375
             ELSE
                ZEROCHK=.TRUE.
             ENDIF
          ENDIF
          DO 400 I=1,N
             RK2(I)=STEP*F(I,TERM,T,KNT,KOUNT,TBEG,TEND)
400       CONTINUE
          T=OTIME(KOUNT)-3.D0*STEP/8.D0
          KNT=3
          DO 500 I=1,N
             TERM(I)=X(I)+(3.D0*RK1(I)+9.D0*RK2(I))/32.D0
500       CONTINUE
          IF(TERM(1) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN).LT.1.D-6)THEN
                ZEROVOL=.TRUE.
                PASS=.TRUE.
                GOTO 1375
             ELSE
                ZEROCHK=.TRUE.
             ENDIF
          ENDIF
          DO 600 I=1,N
             RK3(I)=STEP*F(I,TERM,T,KNT,KOUNT,TBEG,TEND)
600       CONTINUE
          T=OTIME(KOUNT)-12.D0*STEP/13.D0
          KNT=4
          DO 700 I=1,N
             TERM(I)=X(I) + (1932.D0*RK1(I)-7200.D0*RK2(I)+
     +            7296.D0*RK3(I))/2197.D0
700       CONTINUE
          IF(TERM(1) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN).LT.1.D-6)THEN
                ZEROVOL=.TRUE.
                PASS=.TRUE.
                GOTO 1375
             ELSE
                ZEROCHK=.TRUE.
             ENDIF
          ENDIF
          DO 800 I=1,N
             RK4(I)=STEP*F(I,TERM,T,KNT,KOUNT,TBEG,TEND)
800       CONTINUE
          T=OTIME(KOUNT)-STEP
          KNT=5
          DO 900 I=1,N
             TERM(I)=X(I) +439.D0*RK1(I)/216.D0-
     +            8.D0*RK2(I)+3680.D0*RK3(I)/513.D0-
     +            845.D0*RK4(I)/4104.D0
900       CONTINUE
```

```
          IF(TERM(1) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN).LT.1.D-6)THEN
                ZEROVOL=.TRUE.
                PASS=.TRUE.
                GOTO 1375
             ELSE
                ZEROCHK=.TRUE.
             ENDIF
          ENDIF
          DO 1000 I=1,N
             RK5(I)=STEP*F(I,TERM,T,KNT,KOUNT,TBEG,TEND)
1000      CONTINUE
          T=OTIME(KOUNT)-STEP/2.D0
          KNT=6
          DO 1100 I=1,N
             TERM(I)=X(I)-8.D0*RK1(I)/27.D0+
     +            2.D0*RK2(I)-3544.D0*RK3(I)/2565.D0+
     +            1859.D0*RK4(I)/4104.D0-11.D0*RK5(I)/40.D0
1100      CONTINUE
          IF(TERM(1) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN).LT.1.D-6)THEN
                ZEROVOL=.TRUE.
                PASS=.TRUE.
                GOTO 1375
             ELSE
                ZEROCHK=.TRUE.
             ENDIF
          ENDIF
          DO 1200 I=1,N
             RK6(I)=STEP*F(I,TERM,T,KNT,KOUNT,TBEG,TEND)
1200      CONTINUE
          IF(TERM(1) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN).LT.1.D-6)THEN
                ZEROVOL=.TRUE.
                PASS=.TRUE.
                GOTO 1375
             ELSE
                ZEROCHK=.TRUE.
             ENDIF
          ENDIF
          PASS=.TRUE.

***********************************************
** CALCULATE ERRORS RESULTING FROM STEP SIZE **
***********************************************

          DO 1300 I=1,N
             R(I)=DABS(RK1(I)/360.D0 -128.D0*RK3(I)/4275.D0-
     +            2197.D0*RK4(I)/75240.D0+RK5(I)/50.D0+
     +            2.D0*RK6(I)/55.D0)/STEP
             IF(R(I).GT.TOL(I)) PASS=.FALSE.
1300      CONTINUE

*************************************************************************
** MAKE SURE THE SOLVER ISN'T "STUCK" BECAUSE OF THE ERROR TOLERANCES  **
*************************************************************************

          IF(DABS(R(1)-RIPREV).LT.1.D-6 .AND.
     +       DABS(STEP-DTMIN).LT.1.D-6)THEN
             IF(ZEROCHK)THEN
                PASS=.TRUE.
                ZEROVOL=.TRUE.
                GOTO 1375
             ELSE
                WRITE(6,*)
                WRITE(6,*)'THE CURRENT RUN IS "STUCK" BUT WE HAVE
```

```
     +FORCED IT TO MOVE ON'
                  WRITE(6,*)'DESPITE THE GIVEN TOLERANCES'
                  WRITE(6,*)
                  PASS=.TRUE.
                  GOTO 1375
               ENDIF
            ELSE
               RIPREV=R(1)
            ENDIF

            DO 1310 I=1,N
               IF(R(I) .LT. 1.D-15)R(I)=.1
1310        CONTINUE
            DELMIN=4.D0

*******************************************************
** 'DEL' IS A VARIABLE USED TO UPDATE THE STEP SIZE **
*******************************************************

            DO 1350 I = 1,N
               DEL(I)=0.84*(TOL(I)/R(I))**(1.D0/4.D0)
               DELMIN=DMIN1(DEL(I),DELMIN)
1350        CONTINUE

*********************************************************
** IF THE ERROR IS LESS THAN THE GIVEN TOLERANCES ... **
*********************************************************

1375        IF(PASS)THEN
               IF(OTIME(KOUNT)-STEP.GE.TEND)THEN
                  KOUNT=KOUNT+1
                  OTIME(KOUNT)=OTIME(KOUNT-1)-STEP
                  IF(ZEROVOL)THEN
                     X(1)=0.D0
                     X(2)=TODELI
                     ZEROVOL=.FALSE.
                     ZEROCHK=.FALSE.
                  ELSE

************************************
** CALCULATE VOLUME AND DEL O-18 **
************************************

                     DO 1400 I=1,N
                        X(I)=X(I)+25.D0*RK1(I)/216.D0+
     +                       1408.D0*RK3(I)/2565.D0+
     +                       2197.D0*RK4(I)/4104.D0- RK5(I)/5.D0
1400                 CONTINUE
                  ENDIF

*************************************************
** SALT BALANCE STUFF FOR THE ENTIRE TIME-STEP **
*************************************************

                  IF(X(1) .GT. VOLMAX)THEN
**************
** OVERFLOW **
**************
                     VOL_OUT=X(1)-VOLMAX
                     CL(KOUNT)=CL(KOUNT-1)+QCL_IN*STEP-
     +                    VOL_OUT*1.D3*CONC_CL(KOUNT-1)
                     IF(CL(KOUNT).LT.0.D0)CL(KOUNT)=3.95D-4*30.02D9*
     +                    1.D3
                     X(1)=VOLMAX
                     CONC_CL(KOUNT)=CL(KOUNT)/(X(1)*1.D3)
```

```
*******************************************************************************
** "OCL_OUT" RECORDS THE SUM MOLES OF CL THAT LEAVE DURING EACH TIME-STEP **
*******************************************************************************

                 OCL_OUT(KOUNT)= VOL_OUT*1.D3*CONC_CL(KOUNT-1)+
     +                OCL_OUT(KOUNT-1)

              ELSE IF(X(1) .LE. 10.D0)THEN
**************
** DRY LAKE **
**************

                 OCL_OUT(KOUNT)=OCL_OUT(KOUNT-1)
                 CL(KOUNT)=0.D0
                 X(1)=0.D0
                 CONC_CL(KOUNT)=0.D0
              ELSE

*****************************
** BETWEEN DRY AND OVERFLOW **
*****************************

                 OCL_OUT(KOUNT)=OCL_OUT(KOUNT-1)
                 CL(KOUNT)=CL(KOUNT-1)+QCL_IN*STEP
                 CONC_CL(KOUNT)=CL(KOUNT)/(X(1)*1.D3)

****************************
** CHECK FOR CL SATURATION **
****************************

                 IF(CONC_CL(KOUNT) .GT. 6.1D0)THEN
                    CONC_CL(KOUNT)=6.1D0
                    CL(KOUNT)=6.1D0*(X(1)*1.D3)
                 ENDIF
              ENDIF

***********************************************
** CALCULATE DEL DOLOMITE FROM DEL WATER, X(2)**
***********************************************

              ODEL_OUT(KOUNT)=X(2)
              DELDOL=FDDOL(DOLTEMP,X(2))

****************************************************
** STORE VALUES IN ARRAYS FOR THE PLOTTING ROUTINE **
****************************************************

              CALL FINDQT(NQPTS,OTIME(KOUNT),IQNDEX,QITIME)

              IF(INCHOICE .GE. 8)THEN
                 CALL QINTERP(OTIME(KOUNT),IQNDEX,QI,QIHIST,
     +                QITIME)
              ELSE
                 QI=FQI(TBEG-OTIME(KOUNT))
              ENDIF
              IF(QI .LE. 0.D0)QI=0.D0

***************************
** WRITE RESULTS TO FILE **
***************************

              AREA(KOUNT)=OAREA(X(1))
*              WRITE(96,*)OTIME(KOUNT),AREA(KOUNT)
*              WRITE(99,*)OTIME(KOUNT),DELDOL
**             WRITE(95,*)OTIME(KOUNT),QI

              IF(SU)THEN
```

```
                    IF(OTIME(KOUNT) .LT. SAVETIME .AND. ISTCNT .LE.
     +                   NUMST)THEN
                       WRITE(70,*)'OWENS LAKE'
                       WRITE(70,*)'STARTUP TIME #',ISTCNT
                       WRITE(70,*)OTIME(KOUNT),X(1),DELDOL,
     +                       CONC_CL(KOUNT)
                       WRITE(70,*)
                       ISTCNT=ISTCNT+1
                       SAVETIME=SUT(ISTCNT)
                    ENDIF
                 ENDIF

                 STEP=STEP+0.5D0*STEP
                 IF(STEP .GT. DTMAX)STEP=DTMAX
                 GOTO 100

***********************************
** MAKE SURE THE SOLVER DOESN'T **
** OVERSTEP DESIGNATED END TIME **
***********************************

*********************************************************************
** THIS ELSE IF CORRESPONDS TO "IF(OTIME(KOUNT)-STEP.GE.TEND)THEN **
*********************************************************************

               ELSEIF(OTIME(KOUNT)-STEP.LT.TEND)THEN
                  DTMAX=OTIME(KOUNT)-TEND
                  STEP=DTMAX
                  GOTO 100
               ENDIF

***************************************
** ADJUST THE SIZE OF THE TIME STEP **
***************************************

************************************************
** THIS ELSEIF CORRESPONDS TO "IF(PASS)THEN" **
************************************************

             ELSEIF(DELMIN .LE. 0.1)THEN
                STEP=STEP*1.0D-1
             ELSEIF(DELMIN .GE. 4.D0)THEN
                STEP=4.D0*STEP
             ELSE
                STEP=DELMIN*STEP
             ENDIF
             IF(STEP.GT.DTMAX)STEP=DTMAX
             IF(STEP.LT.DTMIN)STEP=DTMIN

**************************************************************
** THIS ELSE CORRESPONDS TO "IF(OTIME(KOUNT).GT.TEND)THEN" **
**************************************************************

          ELSE
             IF(.NOT. GRAF)THEN
                WRITE(6,*)
                WRITE(6,*)'THE RKF SOLVED',KOUNT,' POINTS IN',ITER,' IT
     +ERATIONS'
                WRITE(6,*)
                WRITE(6,*)'FINAL DEL FOR OWENS =',DELDOL
                WRITE(6,*)'FINAL AREA FOR OWENS  =',AREA(KOUNT)
                WRITE(6,*)'FINAL VOLUME FOR OWENS =',X(1)
                WRITE(6,*)'FINAL CL CONC, OWENS =',CONC_CL(KOUNT)
                FINDEL=DELDOL
                FINAREA=AREA(KOUNT)
                FINVOL=X(1)
```

```
                FINCL=CONC_CL(KOUNT)
             ENDIF

             NOPTS=KOUNT
             RETURN
          ENDIF
100    ENDWHILE
       NOPTS=KOUNT
**     IF(ITER .GT. ITMAX)WRITE(6,*)'MAX # OF ITERATIONS EXCEEDED'
       IF(.NOT. GRAF)THEN
          WRITE(6,*)
          WRITE(6,*)'THE RKF SOLVED',KOUNT,' POINTS IN',ITER,' IT
     +ERATIONS'
          WRITE(6,*)
          WRITE(6,*)'FINAL DEL FOR OWENS =',DELDOL
          WRITE(6,*)'FINAL AREA FOR OWENS  =',AREA(KOUNT)
          WRITE(6,*)'FINAL VOLUME FOR OWENS =',X(1)
          WRITE(6,*)'FINAL CL CONC, OWENS =',CONC_CL(KOUNT)
          FINDEL=DELDOL
          FINAREA=AREA(KOUNT)
          FINVOL=X(1)
          FINCL=CONC_CL(KOUNT)
          WRITE(6,*)
       ENDIF

       END


*******************************************************************************
*******************************************************************************
*    BELOW LIES A CHAOTIC CONVOLUTION OF ESOTERIC ENIGMAS THAT HOPEFULLY      *
*    ACCOMPLISH THE ISOTOPIC AND LAKE LEVEL VOODOO WE SET OUT TOODOO.         *
*    ACTUALLY THIS PART OF THE PROGRAM CALCULATES THE DERIVATIVES OF LAKE     *
*    VOLUME AND ISOTOPIC COMPOSITION WITH RESPECT TO TIME.                    *
*******************************************************************************
*******************************************************************************

       DOUBLE PRECISION FUNCTION F(I,TERM,T,KNT,KOUNT,TBEG,TEND)
       IMPLICIT DOUBLE PRECISION (A-H,O-Z)

       LOGICAL FOUND,GUESS,GRAF,CPARAM,GUESS2,SU

       PARAMETER(NUMA=25000,NUMB=2000)
       PARAMETER(QCL_IN=1.67D8,VOLMAX=30.02D9)

       SAVE

       COMMON/BOTH/CL(NUMA),TSOD(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

       COMMON/BOTH2/CL_C,TSOD_C(10),TCL_C(10),TIME,TCO3_C(10),
     + CQO(10),GRAF,CONC_CL_C,DOLTEMP_C,DELDOL_C,
     + TCDELI,TCTEMP,CDEL_OUT,CCL_OUT,CL_S,NOPTS,
     + TSOD_S(10),TCL_S(10),TCO3_S(10),SQO(10),CONC_CL_S,
     + DOLTEMP_S,DELDOL_S,TSDELI,TSTEMP,SCL_DEP,TTLCL_IN,NPTS,
     + SUM_SCL_DEP,QI_C,QI_S,PQI

       COMMON/PREV/PCL_P,PCONC_CL_P,PSUM_PCL_DEP,PCL_C,PCONC_CL_C,
     + PCDEL_OUT,PCL_S,PCONC_CL_S,PSUM_SCL_DEP,PALLAREA,DATSAV,
     + SDC,SDP1,SDP2
```

```
      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S

      COMMON/HIST/QIHIST(1000),HUMTIME(500),NHPTS,SU,SUT(15),
     + SAVETIME,NUMST,QITIME(1000),NQPTS

      COMMON/INFLOW/INCHOICE,A,B,C

      DIMENSION TERM(3),TCONC_CL(10),V_OUT(10)

      IF(I.EQ.1)THEN

*************************************
** CALCULATE DV/DT FOR OWENS LAKE **
*************************************

         VOL = TERM(1)
         IF(VOL .LE. 10.D0)VOL=0.D0


******************************
** CONSTANT PARAMETER OPTION **
******************************

         IF(CPARAM)THEN
            TODELI=DELIC
            TOTEMP=TEMPC
            TOEVAP=EVAPC
            TOPRECIP=PRECIPC
            TODELP=DELPC
            TODELA=DELAC
            CALL FINDHT(NHPTS,T,IHNDEX,HUMTIME)
            CALL FINDQT(NQPTS,T,IQNDEX,QITIME)
            CALL HUMTERP(T,IHNDEX,TOHUM,OHUM,HUMTIME)
         ELSE

**********************************************************************
** DETERMINE VALUES OF NECESSARY PARAMETERS BY ASSIGNING VALUES OR  **
** INTERPOLATING BETWEEN GIVEN VALUES                                **
**********************************************************************

            CALL FINDT(NPTS,T,INDEX,FOUND,WTIME)
            CALL FINDHT(NHPTS,T,IHNDEX,HUMTIME)
            CALL FINDQT(NQPTS,T,IQNDEX,QITIME)

            IF(FOUND)THEN
               TODELI = ODELI(INDEX)
               TOTEMP = OTEMP(INDEX)
               TOEVAP = OEVAP(INDEX)
               TOPRECIP = OPRECIP(INDEX)
               TODELP = ODELP(INDEX)
               TODELA = ODELA(INDEX)
            ELSE
               CALL OINTERP(T,INDEX,TODELI,TOTEMP,TOEVAP,TOPRECIP,
     +              TODELP,TODELA)
            ENDIF
            CALL HUMTERP(T,IHNDEX,TOHUM,OHUM,HUMTIME)
         ENDIF
```

```
*******************************************************************
** "REMEMBER" TEMP AT END OF TIMESTEP TO CALCULATE DEL DOLOMITE **
*******************************************************************

          IF(KNT .EQ. 5)DOLTEMP=TOTEMP

*************************************************
** TRACK TOTAL ELAPSED TIME TO CALCULATE INFLOW**
*************************************************

          ETIME = TBEG-T

*****************************************************
** ALSO NEED TO KNOW DEL TIME WITHIN THE TIME-STEP **
*****************************************************

          DTIME = OTIME(KOUNT)-T

****************************
** CALCULATE AREA OF LAKE **
****************************

          AREA(KOUNT) = OAREA(VOL)

******************************
** CALCULATE PRECIPITATION **
******************************

          QP = AREA(KOUNT)*TOPRECIP

**********************************
** THE INFAMOUS "SALT" BALANCE **
**********************************

          IF(DABS(T-OTIME(KOUNT)).LT.1.D-8)THEN
             TCL(KNT)=CL(KOUNT)
             TCONC_CL(KNT)=CONC_CL(KOUNT)
             TCO3(KNT) = TCL(KNT)*1.03D0
             TSOD(KNT)=TCL(KNT)+TCO3(KNT)
**************************************************
** CALCULATE TOTAL OUTFLOW VOLUME FROM TIME TO T **
** AND ADJUST IF VOL EXCEEDS VOLMAX             **
**************************************************

          ELSE
             IF(VOL .LE. VOLMAX)THEN
                V_OUT(KNT)=0.D0
             ELSE
                V_OUT(KNT) = VOL-VOLMAX
                VOL=VOLMAX
             ENDIF

********************************************************
**  CALCULATE CONCENTRATIONS FOR INTERMEDIATE TIME "T" **
********************************************************

             TCL(KNT) = (OTIME(KOUNT)-T)*QCL_IN+CL(KOUNT)-V_OUT(KNT)*
     +             CONC_CL(KOUNT)*1.D3
             IF(TCL(KNT).LT.0.D0)TCL(KNT)=3.95D-4*30.02D9*1.D3

*******************************************************************
** CONCENTRATION UNITS ARE MOLARITY SO VOL MUST BE MULT BY 1000 TO **
** CONVERT M^3 TO LITERS **
*******************************************************************
```

```
            IF(VOL .GT. 10.D0)THEN
               TCONC_CL(KNT)=TCL(KNT)/(VOL*1.0D3)
            ELSE
               TCONC_CL(KNT)=0.D0
               TCL(KNT)=0.D0
            ENDIF

************************************
** CHECK FOR CHLORIDE SATURATION **
************************************

            IF(TCONC_CL(KNT) .GT. 6.1D0)THEN
               TCONC_CL(KNT)=6.1D0
               TCL(KNT)=6.1D0*(VOL*1.D3)
            ENDIF

***************************************************
** TOTAL CO3 IS KEPT AT A CONSTANT RATIO WITH CL **
***************************************************

            TCO3(KNT) = TCL(KNT)*1.03D0

********************************************************************
** AMT OF SODIUM IS THE AMT NECESSARY TO ACHIEVE ELECTRONEUTALITY **
********************************************************************

            TSOD(KNT)=TCL(KNT)+TCO3(KNT)
         ENDIF

*********************************************
**  CALCULATE BACK-CONDENSATION FLUX (QC) **
*********************************************

         IF(VOL .GT. 10.D0 .AND. TCONC_CL(KNT) .GT. 0.D0)THEN
            PHI=FPHI(TCONC_CL(KNT),SUM)
         ELSE
            PHI=0.D0
         ENDIF

         AW=DEXP(-18.D0*PHI*SUM*0.5D0/1.D3)

********************************
** CALCULATE EVAPORATION (QE) **
********************************

         QE = AREA(KOUNT)*TOEVAP*AW

         IF(VOL .GT. 10.D0)THEN
            QC=(TOHUM*QE)/AW
         ELSE
            QC=0.D0
         ENDIF

*************************************
**  CALCULATE DV/DT IF VOL IS ZERO **
*************************************

         IF(TERM(1) .LE. 10.D0)THEN
            IF(INCHOICE .GE. 8)THEN
               CALL QINTERP(T,IQNDEX,QI,QIHIST,QITIME)
            ELSE
               QI=FQI(ETIME)
            ENDIF
            IF(QI .LT. 0.D0)QI=0.D0
            IF(GUESS)THEN
*              WRITE(96,*)TBEG,AREA(KOUNT)
```

```
**                WRITE(95,*)TBEG,QI
                  DELDOL=FDDOL(TOTEMP,TERM(2))
*                 WRITE(99,*)TBEG,DELDOL
                  GUESS=.FALSE.
               ENDIF

               QO(KNT)=0.D0

*****************************************
** ASSUME QO FOR OWENS = QI FOR CHINA **
*****************************************

               IF(KNT .EQ. 1)THEN
                  OQO(KOUNT)=QO(KNT)
*                 WRITE(97,*)T,OQO(KOUNT)
               ENDIF
               IF(DABS(T-TEND).LT.1.D-8)THEN
                  OQO(KOUNT+1)=QO(KNT)
*                 WRITE(97,*)T,OQO(KOUNT)
               ENDIF

               F=QI

               IF(F .LE. 0.D0)THEN
                  DV_DT=0.D0
               ELSE
                  DV_DT=F
               ENDIF

**************************************************
**  CALCULATE DV/DT IF VOL IS LESS THAN VOLMAX **
**************************************************

            ELSE IF(TERM(1)  .LT. VOLMAX)THEN

***************************
** CALCULATE INFLOW (QI) **
***************************

               IF(INCHOICE .GE. 8)THEN
                  CALL QINTERP(T,IQNDEX,QI,QIHIST,QITIME)
               ELSE
                  QI=FQI(ETIME)
               ENDIF
               IF(QI .LT. 0.D0)QI=0.D0
               IF(GUESS)THEN
*                 WRITE(96,*)TBEG,AREA(KOUNT)
**                WRITE(95,*)TBEG,QI
                  DELDOL=FDDOL(TOTEMP,TERM(2))
*                 WRITE(99,*)TBEG,DELDOL
                  GUESS=.FALSE.
               ENDIF

**********************
**  CALCULATE DV/DT **
**********************

               F=QI+QC-QE+QP
               DV_DT=F
               QO(KNT)=0.D0
               IF(KNT .EQ. 1)THEN
                  OQO(KOUNT)=QO(KNT)
*                 WRITE(97,*)T,OQO(KOUNT)
               ENDIF
               IF(DABS(T-TEND).LT.1.D-8)THEN
                  OQO(KOUNT+1)=QO(KNT)
```

```
*                WRITE(97,*)T,OQO(KOUNT)
              ENDIF

*********************************************
**  CALCULATE DV/DT IF OWENS LAKE IS FULL **
*********************************************

           ELSE

******************
** CALCULATE QI **
******************

              IF(INCHOICE .GE. 8)THEN
                 CALL QINTERP(T,IQNDEX,QI,QIHIST,QITIME)
              ELSE
                 QI=FQI(ETIME)
              ENDIF

              IF(GUESS)THEN
*                 WRITE(96,*)TBEG,AREA(KOUNT)
**               WRITE(95,*)TBEG,QI
                 DELDOL=FDDOL(TOTEMP,TERM(2))
*                WRITE(99,*)TBEG,DELDOL
                 GUESS=.FALSE.
              ENDIF

******************
** CALCULATE QO **
******************

              QO(KNT)=QI+QC-QE+QP

*********************
** CALCULATE DV/DT **
*********************

              IF(QO(KNT) .LT. 0.D0)THEN
                 QO(KNT)=0.D0
                 IF(KNT .EQ. 1)THEN
                    OQO(KOUNT)=QO(KNT)
*                   WRITE(97,*)T,OQO(KOUNT)
                 ENDIF
                 IF(DABS(T-TEND).LT.1.D-8)THEN
                    OQO(KOUNT+1)=QO(KNT)
*                   WRITE(97,*)T,OQO(KOUNT)
                 ENDIF
                 F=QI+QC-QE+QP
                 DV_DT=F
              ELSE
                 F=QO(KNT)
                 IF(KNT .EQ. 1)THEN
                    OQO(KOUNT)=QO(KNT)
*                   WRITE(97,*)T,OQO(KOUNT)
                 ENDIF
                 IF(DABS(T-TEND).LT.1.D-8)THEN
                    OQO(KOUNT+1)=QO(KNT)
*                   WRITE(97,*)T,OQO(KOUNT)
                 ENDIF
                 DV_DT=0.D0
              ENDIF
           ENDIF

******************************************************************************
*   A FUNCTION SUBROUTINE TO CALCULATE THE ISOTOPIC HISTORY OF OWENS LAKE    *
******************************************************************************
```

```
      ELSE IF(I.EQ.2)THEN

         DELL = TERM(2)

************************
** CALCULATE DDEL/DT  **
************************

         IF(VOL .LE. 10.D0)THEN
            F=0.D0

         ELSE

******************************************
** CALCULATE ISOTOPIC ENRICHMENT FACTOR **
******************************************

            EPS = FEPS(TOTEMP)

********************************************
** CALCULATE DEL OF THE BACK-CONDENSATION **
********************************************

            ODELC =EPS*(1.D0+(TODELA/1.D3))+TODELA

**************************************
** CALCULATE DEL OF THE EVAPORATION **
**************************************

            ODELE = DELE(DELL,EPS,TOHUM)

*****************************************************
** SET DEL OF THE OUTFLOW EQUAL TO DEL OF THE LAKE **
*****************************************************

            ODELO = DELL

            F=(QI*TODELI+QC*ODELC+QP*TODELP-QO(KNT)*ODELO-QE*
     +           ODELE-DELL*DV_DT)/VOL

         ENDIF

      END IF
      RETURN
      END



      SUBROUTINE RKF_CS(NCS,X_CS,TBEG,TEND,TOL_CS,DTMAX_CS,DTMIN_CS,
     +     ITMAX_CS)
*******************************************************************************
*     SOLVE A SYSTEM OF PARTIAL DIFFERENTIAL EQUATION OF THE FORM:            *
*                       F(T,X)= X'                                            *
*      BETWEEN T1,T2, GIVEN THE INITIAL CONDITION XO(T1)                      *
*******************************************************************************
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      PARAMETER(NUMA=25000,NUMB=2000)
      PARAMETER (VOLMAX_C=0.696D9,VOLMAX_S=85.28D9,AREAMAX_P=.727D9,
     +     AREAMAX_D=0.583D9,SLTCONST=0.0127)

      LOGICAL PASS,GRAF,ONLY1,ZEROVOL_C,ZEROCHK_C,ZEROVOL_S,
     + ZEROCHK_S,COAL,DECOUP,COUP,FOUND,CPARAM,GUESS,SU

      SAVE
```

```
      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/BOTH/CL(NUMA),TSOO(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

      COMMON/BOTH2/CL_C,TSOO_C(10),TCL_C(10),TIME,TCO3_C(10),
     + CQO(10),GRAF,CONC_CL_C,DOLTEMP_C,DELDOL_C,
     + TCDELI,TCTEMP,CDEL_OUT,CCL_OUT,CL_S,NOPTS,
     + TSOO_S(10),TCL_S(10),TCO3_S(10),SQO(10),CONC_CL_S,
     + DOLTEMP_S,DELDOL_S,TSDELI,TSTEMP,SCL_DEP,TTLCL_IN,NPTS,
     + SUM_SCL_DEP,QI_C,QI_S,PQI

      COMMON/PREV/PCL_P,PCONC_CL_P,PSUM_PCL_DEP,PCL_C,PCONC_CL_C,
     + PCDEL_OUT,PCL_S,PCONC_CL_S,PSUM_SCL_DEP,PALLAREA,DATSAV,
     + SDC,SDP1,SDP2

      COMMON/NEW/OPREV,CPREV,SPREV,PPREV,DPREV

      COMMON/HIST/QIHIST(1000),HUMTIME(500),NHPTS,SU,SUT(15),
     + SAVETIME,NUMST,QITIME(1000),NQPTS

      COMMON/FINAL/FINDEL,FINVOL,FINAREA,FINCL

      DIMENSION X_CS(4),RK1(4),RK2(4),RK3(4),RK4(4),RK5(4),RK6(4),R(4)
      DIMENSION TERM(4),DEL(4),TOL_CS(4)

      OPEN(UNIT=87,FILE='CONC_S.OUT',STATUS='UNKNOWN')
      OPEN(UNIT=81,FILE='ALLAREA.OUT',STATUS='UNKNOWN')
      OPEN(UNIT=78,FILE='DEL_S.OUT',STATUS='UNKNOWN')
      OPEN(UNIT=77,FILE='AREA_S.OUT',STATUS='UNKNOWN')
      OPEN(UNIT=76,FILE='QI_S.OUT',STATUS='UNKNOWN')
      OPEN(UNIT=74,FILE='SUMCL.OUT',STATUS='UNKNOWN')

      OPEN(UNIT=91,FILE='QO_S.OUT',STATUS='UNKNOWN')
**    OPEN(UNIT=83,FILE='PCL_DEP.OUT',STATUS='UNKNOWN')
      OPEN(UNIT=55,FILE='RES.OUT',STATUS='UNKNOWN')
      OPEN(UNIT=56,FILE='UPDATE.OUT',STATUS='UNKNOWN')


      XMIN=TBEG
      XMAX=TEND

      TIME=TBEG
      WRTM=TBEG
      STEP=DTMAX_CS
      KOUNT=1

      PCDEL_OUT=X_CS(2)
      SCL_DEP=0.D0
      CCL_OUT=0.D0

***********************************
** WRITE INITIAL VALUES TO FILE **
***********************************

      WRITE(74,*)TBEG/1.D6,PSUM_SCL_DEP
**    WRITE(83,*)TBEG/1.D6,PSUM_PCL_DEP
      WRITE(87,*)TBEG/1.D6,PCONC_CL_S
```

```
        ONLY1=.TRUE.
        COAL=.FALSE.
        DECOUP=.FALSE.
        COUP=.FALSE.
        ZEROVOL_C=.FALSE.
        ZEROCHK_C=.FALSE.
        ZEROVOL_S=.FALSE.
        ZEROCHK_S=.FALSE.

        IF(X_CS(3) .GT. 65.87D9)COAL=.TRUE.

*********************************************
** INITIALIZE PARAMETERS TO RESTART MODEL **
*********************************************

        SAVETIME=SUT(1)
        ISTCNT=1
        ITER=0

**************************************
** THE SOLVING ROUTINE BEGINS HERE **
**************************************

        WRITE(6,*)
        WRITE(6,*)
        WRITE(6,*)'SOLVING DIFFERENTIAL EQUATIONS'
        WRITE(6,*)'FOR CHINA AND SEARLES LAKE'
        WRITE(6,*)
        WRITE(6,*)

        WHILE(ITER .LT. 50000)DO
           ITER=ITER+1
           IF(TIME.GT.TEND)THEN

              IF(COAL)THEN
                 NST=3
              ELSE
                 NST=1
              ENDIF

              KNT=1
              T=TIME

              ZEROVOL_C=.FALSE.
              ZEROCHK_C=.FALSE.
              ZEROVOL_S=.FALSE.
              ZEROCHK_S=.FALSE.

              DO 200 I=NST,NCS
                 RK1(I)=STEP*F_CS(I,X_CS,T,KNT,KOUNT,TBEG,COAL)
200           CONTINUE

***************************************************************************
** "TERM" IS AN ARRAY WHICH STORES APPROXIMATIONS OF VOL AND DEL O-18 **
** WHICH WILL BE USED IN FINAL CALCULATIONS IF ERRORS WITHIN THE STEP **
** ARE LESS THAN THE GIVEN TOLERANCES.                                  **
***************************************************************************

              T=TIME-STEP/4.D0
              KNT=2
              DO 300 I=NST,NCS
                 TERM(I)=X_CS(I)+ RK1(I)/4.D0
300           CONTINUE
              IF(TERM(1) .LE. 10.D0 .AND. .NOT. COAL)THEN
                 IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
```

```
            ZEROVOL_C=.TRUE.
            PASS=.TRUE.
         ELSE
            ZEROCHK_C=.TRUE.
         ENDIF
      ENDIF
      IF(TERM(3) .LE. 10.D0)THEN
         IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
            ZEROVOL_S=.TRUE.
            PASS=.TRUE.
         ELSE
            ZEROCHK_S=.TRUE.
         ENDIF
      ENDIF
      DO 400 I=NST,NCS
         RK2(I)=STEP*F_CS(I,TERM,T,KNT,KOUNT,TBEG,COAL)
400   CONTINUE
      T=TIME-3.D0*STEP/8.D0
      KNT=3
      DO 500 I=NST,NCS
         TERM(I)=X_CS(I)+(3.D0*RK1(I)+9.D0*RK2(I))/32.D0
500   CONTINUE

      IF(TERM(1) .LE. 10.D0 .AND. .NOT. COAL)THEN
         IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
            ZEROVOL_C=.TRUE.
            PASS=.TRUE.
         ELSE
            ZEROCHK_C=.TRUE.
         ENDIF
      ENDIF
      IF(TERM(3) .LE. 10.D0)THEN
         IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
            ZEROVOL_S=.TRUE.
            PASS=.TRUE.
         ELSE
            ZEROCHK_S=.TRUE.
         ENDIF
      ENDIF

      DO 600 I=NST,NCS
         RK3(I)=STEP*F_CS(I,TERM,T,KNT,KOUNT,TBEG,COAL)
600   CONTINUE
      T=TIME-12.D0*STEP/13.D0
      KNT=4
      DO 700 I=NST,NCS
         TERM(I)=X_CS(I)+(1932.D0*RK1(I)-7200.D0*RK2(I)+
     +                7296.D0*RK3(I))/2197.D0
700   CONTINUE

      IF(TERM(1) .LE. 10.D0 .AND. .NOT. COAL)THEN
         IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
            ZEROVOL_C=.TRUE.
            PASS=.TRUE.
         ELSE
            ZEROCHK_C=.TRUE.
         ENDIF
      ENDIF
      IF(TERM(3) .LE. 10.D0)THEN
         IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
            ZEROVOL_S=.TRUE.
            PASS=.TRUE.
         ELSE
            ZEROCHK_S=.TRUE.
         ENDIF
      ENDIF
```

```
          DO 800 I=NST,NCS
             RK4(I)=STEP*F_CS(I,TERM,T,KNT,KOUNT,TBEG,COAL)
 800      CONTINUE
          T=TIME-STEP
          KNT=5
          DO 900 I=NST,NCS
             TERM(I)=X_CS(I)+439.D0*RK1(I)/216.D0-
     +             8.D0*RK2(I)+3680.D0*RK3(I)/513.D0-
     +             845.D0*RK4(I)/4104.D0
 900      CONTINUE

          IF(TERM(1) .LE. 10.D0 .AND. .NOT. COAL)THEN
             IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
                ZEROVOL_C=.TRUE.
                PASS=.TRUE.
             ELSE
                ZEROCHK_C=.TRUE.
             ENDIF
          ENDIF
          IF(TERM(3) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
                ZEROVOL_S=.TRUE.
                PASS=.TRUE.
             ELSE
                ZEROCHK_S=.TRUE.
             ENDIF
          ENDIF

          DO 1000 I=NST,NCS
             RK5(I)=STEP*F_CS(I,TERM,T,KNT,KOUNT,TBEG,COAL)
 1000     CONTINUE
          T=TIME-STEP/2.D0
          KNT=6
          DO 1100 I=NST,NCS
             TERM(I)=X_CS(I)-8.D0*RK1(I)/27.D0+
     +             2.D0*RK2(I)-3544.D0*RK3(I)/2565.D0+
     +             1859.D0*RK4(I)/4104.D0-11.D0*RK5(I)/40.D0
 1100     CONTINUE

          IF(TERM(1) .LE. 10.D0 .AND. .NOT. COAL)THEN
             IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
                ZEROVOL_C=.TRUE.
                PASS=.TRUE.
             ELSE
                ZEROCHK_C=.TRUE.
             ENDIF
          ENDIF
          IF(TERM(3) .LE. 10.D0)THEN
             IF(DABS(STEP-DTMIN_CS).LT.1.D-8)THEN
                ZEROVOL_S=.TRUE.
                PASS=.TRUE.
             ELSE
                ZEROCHK_S=.TRUE.
             ENDIF
          ENDIF

          DO 1200 I=NST,NCS
             RK6(I)=STEP*F_CS(I,TERM,T,KNT,KOUNT,TBEG,COAL)
 1200     CONTINUE

          IF(ZEROVOL_C .OR. ZEROVOL_S)GOTO 1375

          PASS=.TRUE.

**************************************************
```

```
** CALCULATE ERRORS RESULTING FROM STEP SIZE **
************************************************

          DO 1300 I=NST,NCS
             R(I)=DABS(RK1(I)/360.D0 -128.D0*RK3(I)/4275.D0-
     +             2197.D0*RK4(I)/75240.D0+RK5(I)/50.D0+
     +             2.D0*RK6(I)/55.D0)/STEP

             IF(R(I).GT.TOL_CS(I))PASS=.FALSE.

             IF(R(I).GT.TOL_CS(I).AND. DABS(STEP-DTMIN_CS)
     +             .LT.1.D-8)THEN
                WRITE(6,*)
                WRITE(6,*)'STUCK ...',I,TOL_CS(I),R(I)
                WRITE(6,*)
             endif


1300      CONTINUE

*****************************************************************************
** MAKE SURE THE SOLVER ISN'T "STUCK" BECAUSE OF THE ERROR TOLERANCES **
*****************************************************************************

          IF(DABS(R(1)-RIPREV).LT.1.D-6 .AND.
     +       DABS(STEP-DTMIN_CS).LT.1.D-6)THEN
             IF(ZEROCHK_C)ZEROVOL_C=.TRUE.
             IF(ZEROCHK_S)ZEROVOL_S=.TRUE.
             IF(ZEROVOL_C .OR. ZEROVOL_S .AND. .NOT. COAL)THEN
                PASS=.TRUE.
                GOTO 1375
             ELSE
                PASS=.TRUE.
                GOTO 1375
             ENDIF
          ELSE
             RIPREV=R(1)
          ENDIF

          DO 1310 I=NST,NCS
             IF(R(I) .LT. 1.0D-3)R(I)=.1
1310      CONTINUE
          DELMIN=4.D0

**********************************************************
** 'DEL' IS A VARIABLE USED TO UPDATE THE STEP SIZE **
**********************************************************

          DO 1350 I = NST,NCS
             DEL(I)=0.84*(TOL_CS(I)/R(I))**(1.D0/4.D0)
             DELMIN=DMIN1(DEL(I),DELMIN)
1350      CONTINUE

************************************************************
** IF THE ERROR IS LESS THAN THE GIVEN TOLERANCES ... **
************************************************************

1375      IF(PASS)THEN

             IF(TIME-STEP.GE.TEND)THEN
                KOUNT=KOUNT+1
                SLTCORR=SLTCONST*STEP
                TIME=TIME-STEP

                IF(COAL)THEN
```

```
      ***************************************************************
      ** CHECK TO SEE IF CHINA AND SEARLES ARE STILL COALESCED **
      ***************************************************************

                       DO 1700 I=3,4
                          X_CS(I)=X_CS(I)+25.D0*RK1(I)/216.D0+
     +                    1408.D0*RK3(I)/2565.D0+
     +                    2197.D0*RK4(I)/4104.D0- RK5(I)/5.D0
 1700                  CONTINUE

                       IF(X_CS(3) .LE. 65.87D9)THEN
                          COAL=.FALSE.
                          X_CS(1)=0.696D9
                          X_CS(2)=X_CS(4)
                          X_CS(3)=X_CS(3)-X_CS(1)
                          WRITE(6,*)
                          WRITE(6,*)'CHINA AND SEARLES HAVE DECOUPLED'
                          WRITE(6,*)'AT',TIME
                          WRITE(6,*)
                          DECOUP=.TRUE.
                       ELSE
                          X_CS(1)=0.D0
                       ENDIF

                    ELSE IF(ZEROVOL_C .AND. ZEROVOL_S)THEN

      ***********************************************************
      ** CALCULATE VOLUME AND DEL O-18 IF BOTH LAKES ARE  DRY**
      ***********************************************************

                       X_CS(1)=0.D0
                       X_CS(2)=TCDELI
                       ZEROVOL_C=.FALSE.
                       ZEROCHK_C=.FALSE.
                       X_CS(3)=0.D0
                       X_CS(4)=PCDEL_OUT
                       ZEROVOL_S=.FALSE.
                       ZEROCHK_S=.FALSE.

                    ELSE IF(ZEROVOL_C .AND. .NOT. ZEROVOL_S)THEN

      *********************************************************
      ** CALCULATE VOLUME AND DEL O-18 IF CHINA LAKE IS DRY**
      *********************************************************

                       X_CS(1)=0.D0
                       X_CS(2)=TCDELI
                       ZEROVOL_C=.FALSE.
                       ZEROCHK_C=.FALSE.
                       DO 1400 I=3,4
                          X_CS(I)=X_CS(I)+25.D0*RK1(I)/216.D0+
     +                       1408.D0*RK3(I)/2565.D0+
     +                       2197.D0*RK4(I)/4104.D0- RK5(I)/5.D0
 1400                  CONTINUE

                    ELSE IF(ZEROVOL_S .AND. .NOT. ZEROVOL_C)THEN

      ***********************************************************
      ** CALCULATE VOLUME AND DEL O-18 IF SEARLES LAKE IS DRY**
      ***********************************************************

                       X_CS(3)=0.D0
                       X_CS(4)=PCDEL_OUT
                       ZEROVOL_S=.FALSE.
                       ZEROCHK_S=.FALSE.
                       DO 1500 I=1,2
```

```
                  X_CS(I)=X_CS(I)+25.D0*RK1(I)/216.D0+
     +                  1408.D0*RK3(I)/2565.D0+
     +                  2197.D0*RK4(I)/4104.D0- RK5(I)/5.D0
 1500             CONTINUE

               ELSE

************************************************************
** CALCULATE VOLUME AND DEL O-18 IF NEITHER LAKE IS DRY **
** AND CHECK TO SEE IF CHINA AND SEARLES COALESCE       **
************************************************************

                  DO 1600 I=1,NCS
                     X_CS(I)=X_CS(I)+25.D0*RK1(I)/216.D0+
     +                  1408.D0*RK3(I)/2565.D0+
     +                  2197.D0*RK4(I)/4104.D0- RK5(I)/5.D0
 1600             CONTINUE

                  IF(X_CS(3) .GT. 65.87D9)THEN
                     COAL=.TRUE.
                     COUP=.TRUE.
                     X_CS(3)=X_CS(3)+0.696D9
                     CPCNT=0.696D9/X_CS(3)
                     SPCNT=1.D0-CPCNT
                     X_CS(4)=CPCNT*X_CS(2)+SPCNT*X_CS(4)
                     X_CS(1)=0.D0
                     X_CS(2)=X_CS(4)
                     WRITE(6,*)
                     WRITE(6,*)'CHINA AND SEARLES HAVE COALESCED'
                     WRITE(6,*)'AT',TIME
                     WRITE(6,*)
                  ENDIF

               ENDIF

               IF(X_CS(1) .LT. 0.D0)THEN
                  X_CS(1)=0.D0
                  X_CS(2)=TCDELI
               ENDIF

               IF(X_CS(3) .LT. 0.D0)THEN
                  X_CS(3)=0.D0
                  X_CS(4)=PCDEL_OUT
               ENDIF

*****************************************************
** SALT BALANCE STUFF FOR THE ENTIRE TIME-STEP **
*****************************************************

*********************************************
** IF THE LAKES HAVE JUST BEEN DECOUPLED **
*********************************************
               IF(DECOUP)THEN
                  ALL_CL=PCL_S +TTLCL_IN
                  ALL_VOL=X_CS(1)+X_CS(3)
                  ALL_CONC=ALL_CL/(ALL_VOL*1.D3)
** CHINA **
                  CONC_CL_C =ALL_CONC
                  IF(CONC_CL_C  .GT. 6.1D0)THEN
                     CONC_CL_C =6.1D0
                  ENDIF
                  CL_C =CONC_CL_C *X_CS(1)*1.D3
                  CCL_OUT=0.D0
** SEARLES **
                  CONC_CL_S =ALL_CONC
                  IF(CONC_CL_S  .GT. 6.1D0)THEN
```

```
                 CLDEP=CONC_CL_S -6.1D0
                 CONC_CL_S =6.1D0
                 SCL_DEP=CLDEP*X_CS(3)*35.453D0*
     +                3.22D-8
              ELSE
                 SCL_DEP=0.D0
              ENDIF

              SUM_SCL_DEP=PSUM_SCL_DEP +
     +             SCL_DEP+SLTCORR
              CL_S =X_CS(3)*1.D3*CONC_CL_S
              DECOUP=.FALSE.

***********************************************
** IF CHINA AND SEARLES HAVE JUST COALESCED **
***********************************************

           ELSE IF(COUP)THEN
              CL_S =PCL_C +PCL_S +TTLCL_IN
              CONC_CL_S =CL_S /(X_CS(3)*1.D3)
              IF(CONC_CL_S  .GT. 6.1D0)THEN
                 CLDEP=CONC_CL_S -6.1D0
                 CONC_CL_S =6.1D0
                 SCL_DEP=CLDEP*X_CS(3)*35.453D0*
     +                3.22D-8
              ELSE
                 SCL_DEP=0.D0
              ENDIF

              SUM_SCL_DEP=PSUM_SCL_DEP +
     +             SCL_DEP+SLTCORR

              COUP=.FALSE.

************************************************************************
** IF CHINA AND SEARLES ARE STILL COALESCED FROM THE LAST TIME STEP **
************************************************************************

           ELSE IF(COAL)THEN
              IF(X_CS(3) .GT. VOLMAX_S)THEN
                 VOL_OUT_S=X_CS(3)-VOLMAX_S
                 CL_S =PCL_S +TTLCL_IN-
     +                VOL_OUT_S*1.D3*PCONC_CL_S
                 IF(CL_S .LT.0.D0)CL_S =3.95D-4*
     +                85.2809*1.0D3
                 X_CS(3)=VOLMAX_S
                 CONC_CL_S =CL_S /(X_CS(3)*1.D3)
                 SCL_DEP=0.D0
                 SUM_SCL_DEP=PSUM_SCL_DEP +
     +                SCL_DEP+SLTCORR
              ELSE
                 CL_S =PCL_S +TTLCL_IN
                 CONC_CL_S =CL_S /(X_CS(3)*1.D3)
** SATURATION CHECK **
                 IF(CONC_CL_S  .GT. 6.1D0)THEN
                    CLDEP=CONC_CL_S -6.1D0
                    CONC_CL_S =6.1D0
                    CL_S =X_CS(3)*1.D3*CONC_CL_S
                    SCL_DEP=CLDEP*X_CS(3)*35.453D0*
     +                   3.22D-8
                 ELSE
                    SCL_DEP=0.D0
                 ENDIF

                 SUM_SCL_DEP=PSUM_SCL_DEP +
     +                SCL_DEP+SLTCORR
```

```
                    ENDIF

*************************************************************************
** IF CHINA AND SEARLES AREN'T DOING ANY OF THE COUP/DECOUP STUFF **
*************************************************************************

                  ELSE IF(X_CS(1).GT.VOLMAX_C.AND.X_CS(3).GT.VOLMAX_S)
     +                   THEN
***************
** OVERFLOW **  ** CHINA **
***************


                    VOL_OUT_C=X_CS(1)-VOLMAX_C
                    CL_C =PCL_C +TTLCL_IN-
     +                   VOL_OUT_C*1.D3*PCONC_CL_C
                    IF(CL_C .LT.0.D0)CL_C =3.95D-4*
     +                   0.696D9*1.0D3
                    X_CS(1)=VOLMAX_C
                    CONC_CL_C =CL_C /(X_CS(1)*1.D3)
                    CCL_OUT= VOL_OUT_C*1.D3*PCONC_CL_C
***************
** OVERFLOW **  ** SEARLES **
***************

                    VOL_OUT_S=X_CS(3)-VOLMAX_S
                    CL_S =PCL_S +CCL_OUT-
     +                   VOL_OUT_S*1.D3*PCONC_CL_S
                    IF(CL_S .LT.0.D0)CL_S =3.95D-4*
     +                   85.280D9*1.0D3
                    X_CS(3)=VOLMAX_S
                    CONC_CL_S =CL_S /(X_CS(3)*1.D3)
                    SCL_DEP=0.D0
                    SUM_SCL_DEP=PSUM_SCL_DEP +
     +                   SCL_DEP+SLTCORR

*************************************************************
** CHINA LAKE IS OVERFLOWING, SEARLES IS STILL FILLING **
*************************************************************

                  ELSE IF(X_CS(1).GT.VOLMAX_C.AND.X_CS(3).GT.10.D0)
     +                   THEN
***************
** OVERFLOW **  ** CHINA **
***************
                    VOL_OUT_C=X_CS(1)-VOLMAX_C
                    CL_C =PCL_C +TTLCL_IN-
     +                   VOL_OUT_C*1.D3*PCONC_CL_C
                    IF(CL_C .LT.0.D0)CL_C =3.95D-4*
     +                   0.696D9*1.0D3
                    X_CS(1)=VOLMAX_C
                    CONC_CL_C =CL_C /(X_CS(1)*1.D3)
                    CCL_OUT= VOL_OUT_C*1.D3*PCONC_CL_C
**************
** FILLING ** ** SEARLES **
**************
                    CL_S =PCL_S +CCL_OUT
                    CONC_CL_S =CL_S /(X_CS(3)*1.D3)
** SATURATION CHECK **
                    IF(CONC_CL_S  .GT. 6.1D0)THEN
                       CLDEP=CONC_CL_S -6.1D0
                       CONC_CL_S =6.1D0
                       CL_S =X_CS(3)*1.D3*CONC_CL_S
                       SCL_DEP=CLDEP*X_CS(3)*35.453D0*
     +                      3.22D-8
```

```
                    ELSE
                       SCL_DEP=0.D0
                    ENDIF

                    SUM_SCL_DEP=PSUM_SCL_DEP +
     +                   SCL_DEP+SLTCORR

*****************************************************************
** CHINA LAKE IS BETWEEN OVERFLOW & DRY, SEARLES LAKE IS DRY **
*****************************************************************

                 ELSE IF(X_CS(1) .GT. 10.D0 .AND. X_CS(3) .LT. 10.D0)
     +                   THEN

** CHINA **
                    CL_C =PCL_C +TTLCL_IN
                    CONC_CL_C =CL_C /(X_CS(1)*1.D3)
                    IF(CONC_CL_C .GT.6.1D0)THEN
                       CONC_CL_C =6.1D0
                       CL_C =6.1D0*X_CS(1)*1.D3
                    ENDIF
                    CCL_OUT=0.D0

** SEARLES **

                    CL_S =0.D0
                    X_CS(3)=0.D0
                    CONC_CL_S =0.D0
                    SCL_DEP=PCL_S *35.453D0/1.0D3*
     +                   3.22D-8
                    SUM_SCL_DEP=PSUM_SCL_DEP +
     +                   SCL_DEP+SLTCORR

********************************************************************************
** LET'S ASSUME THAT IF CHINA LAKE IS DRY, SEARLES LAKE WON'T BE OVERFLOWING **
********************************************************************************

*****************************************************
** CHINA LAKE IS DRY, SEARLES LAKE IS NOT YET DRY **
*****************************************************

                 ELSE IF(X_CS(1) .LE. 10.D0 .AND. X_CS(3) .GT.. 10.D0)
     +                   THEN

**************
** DRY LAKE ** ** CHINA **
**************

                    CL_C =0.D0
                    X_CS(1)=0.D0
                    CONC_CL_C =0.D0
                    CCL_OUT=0.D0

*****************
** NOT YET DRY ** SEARLES **
*****************

                    CL_S =PCL_S
                    CONC_CL_S =CL_S /(X_CS(3)*1.D3)
** SATURATION CHECK **
                    IF(CONC_CL_S  .GT. 6.1D0)THEN
                       CLDEP=CONC_CL_S -6.1D0
                       CONC_CL_S =6.1D0
                       CL_S =X_CS(3)*1.D3*CONC_CL_S
                       SCL_DEP=CLDEP*X_CS(3)*35.453D0*
     +                      3.22D-8
```

```
                     ELSE
                        SCL_DEP=0.D0
                     ENDIF

                     SUM_SCL_DEP=PSUM_SCL_DEP +
     +                    SCL_DEP+SLTCORR

*********************************************
** CHINA LAKE IS DRY, SEARLES LAKE IS DRY **
*********************************************

                  ELSE IF(X_CS(1) .LE. 10.D0 .AND. X_CS(3) .LE. 10.D0)
     +                    THEN

**************
** DRY LAKE ** ** CHINA **
**************

                     CL_C =0.D0
                     X_CS(1)=0.D0
                     CONC_CL_C =0.D0
                     CCL_OUT=0.D0
**************
** DRY LAKE ** ** SEARLES **
**************
                     CL_S =0.D0
                     X_CS(1)=0.D0
                     CONC_CL_S =0.D0
                     SCL_DEP=PCL_S *35.453D0/1.0D3*
     +                    3.22D-8
                     SUM_SCL_DEP=PSUM_SCL_DEP +
     +                    SCL_DEP+SLTCORR

                  ELSE
******************************************
** BOTH LAKES BETWEEN DRY AND OVERFLOW **
******************************************

** CHINA **
                     CL_C =PCL_C +TTLCL_IN
                     CONC_CL_C =CL_C /(X_CS(1)*1.D3)
                     IF(CONC_CL_C .GT.6.1D0)THEN
                        CONC_CL_C =6.1D0
                        CL_C =6.1D0*X_CS(1)*1.D3
                     ENDIF
                     CCL_OUT=0.D0

** SEARLES **
                     CL_S =PCL_S
                     CONC_CL_S =CL_S /(X_CS(3)*1.D3)
**SATURATION CHECK**
                     IF(CONC_CL_S  .GT. 6.1D0)THEN
                        CLDEP=CONC_CL_S -6.1D0
                        CONC_CL_S =6.1D0
                        CL_S =X_CS(3)*1.D3*CONC_CL_S
                        SCL_DEP=CLDEP*X_CS(3)*35.453D0*
     +                       3.22D-8
                     ELSE
                        SCL_DEP=0.D0
                     ENDIF
                     SUM_SCL_DEP=PSUM_SCL_DEP +
     +                    SCL_DEP+SLTCORR
                  ENDIF

*********************************************
** CALCULATE DEL DOLOMITE FROM DEL WATER **
```

```
************************************************

                    CDEL_OUT =X_CS(2)
                    DELDOL_C=FDDOL(DOLTEMP_C,X_CS(2))
                    DELDOL_S=FDDOL(DOLTEMP_S,X_CS(4))

******************************************************************************
** CALCULATE SURFACE AREAS AND SALT BALANCE FOR PANAMINT AND DEATH VALLEY **
******************************************************************************

                    IF(CPARAM)THEN
                       TPEVAP=EVAPC_P
                       TDEVAP=EVAPC_D
                    ELSE
                       CALL FINDT(NPTS,TIME,INDEX,FOUND,WTIME)

                       IF(FOUND)THEN
                          TPEVAP = PEVAP(INDEX)
                          TDEVAP = DEVAP(INDEX)
                       ELSE
                          CALL PDINTERP(TIME,INDEX,TPEVAP,TDEVAP)
                       ENDIF
                    ENDIF

                    AREA_P=PQI/TPEVAP

                    PQO=0.D0

                    IF(AREA_P .GT. AREAMAX_P)THEN
                       PQO=PQI-TPEVAP*AREAMAX_P
                       VOL_OUT_P=PQO*STEP
                       AREA_P=AREAMAX_P
                    ELSE IF(AREA_P .GE. 0.D0)THEN
                       PQO=0.D0
                    ENDIF

                    VOL_P=PVOL(AREA_P)

                    IF(VOL_P .GT. 0.D0)THEN

                       CL_P =PCL_P +VOL_OUT_S*1.D3*
     +                      PCONC_CL_S -VOL_OUT_P*1.D3*
     +                      PCONC_CL_P
                       CONC_CL_P =CL_P /(1.D3*VOL_P)
**********************
** SATURATION CHECK **
**********************
                       IF(CONC_CL_P  .GT. 6.1D0)THEN
                          CLDEP=CONC_CL_P -6.1D0
                          CONC_CL_P =6.1D0
                          PCL_DEP=2.D0*CLDEP*1.D3*VOL_P*35.453D0/1.D3*
     +                         3.22D-8
                          SUM_PCL_DEP =PSUM_PCL_DEP +
     +                         PCL_DEP
                       ELSE
                          SUM_PCL_DEP =PSUM_PCL_DEP
                       ENDIF
                    ELSE
                       CL_P =0.D0
                       CONC_CL_P=0.D0
                       PCL_DEP=PCL_P *2.D0*35.453/1.0D3*3.22D-8
                       SUM_PCL_DEP =PSUM_PCL_DEP +PCL_DEP
                    ENDIF
******************
** DEATH VALLEY **
******************
```

```
                  DQI=PQO
                  AREA_D=DQI/TDEVAP

                  IF(AREA_D .GT. AREAMAX_D)THEN
                     DQO=DQI-TDEVAP*AREAMAX_D
                     VOL_OUT_D=DQO*STEP
                     AREA_D=AREAMAX_D
                  ELSE IF(AREA_D .GT. 0.D0)THEN
                     DQO=0.D0
                  ENDIF

                  VOL_D=DVOL(AREA_D)

                  IF(VOL_D.GT. 0.D0)THEN

                  ENDIF

***************************************************
** WRITE PANAMINT/DEATH VALLEY STUFF TO FILE **
***************************************************

**                WRITE(83,*)TIME/1.D6,SUM_PCL_DEP

**********************************************
** CALCULATE SUMMATION OF ALL LAKE AREAS **
**********************************************

                  AREA_C=CAREA(X_CS(1))
                  AREA_S=SAREA(X_CS(3))

                  DELTA_O=SOAREA-OPREV
                  DELTA_C=AREA_C-CPREV
                  DELTA_S=AREA_S-SPREV
                  DELTA_P=AREA_P-PPREV
                  DELTA_D=AREA_D-DPREV


                  ALLAREA =DELTA_O+DELTA_C+DELTA_S+DELTA_P
     +                 +DELTA_D+PALLAREA

                  IF(ALLAREA  .LT. 0.D0)ALLAREA =0.D0

                  OPREV=SOAREA
                  CPREV=AREA_C
                  SPREV=AREA_S
                  PPREV=AREA_P
                  DPREV=AREA_D

******************************************
** UPDATE "PEAK AND VALLEY INDICATORS **
******************************************

                  SDP2=SDP1
                  SDP1=SDC
                  SDC=DELDOL_S
                  DIF1=DABS(SDC-SDP1)
                  DIF2=DABS(SDP1-SDP2)
                  DIFMAX=DMAX1(DIF1,DIF2)

************************************
** CHLORIDE SATURATION "WARNING" **
************************************

                  IF(DABS(CONC_CL_S -6.1D0).LT.1.0D-5)THEN
                     RTIME=TIME/1.D6
                     WRITE(6,*)
```

```
                  WRITE(6,'(A,2X,F8.5)')'SEARLES Cl SAT. AT'
     +                 ,RTIME
                  WRITE(6,*)
               ENDIF
*****************************
** WRITE RESULTS TO FILE **
*****************************

               IF((WRTM-TIME).GT. DATSAV)THEN
                  WRTM=TIME
                  WRITE(87,*)TIME/1.D6,CONC_CL_S
** SQO = PQI **
                  WRITE(91,*)TIME/1.D6,PQI
                  WRITE(77,*)TIME/1.D6,AREA_S/1.D9
                  WRITE(78,*)TIME/1.D6,DELDOL_S
                  WRITE(76,*)TIME/1.D6,QI_S
                  WRITE(74,*)TIME/1.D6,SUM_SCL_DEP
                  WRITE(81,*)TIME/1.D6,ALLAREA/1.D9
               ELSE IF(SDC .GT. SDP1 .AND. SDP2 .GT. SDP1)THEN
                  WRTM=TIME
                  WRITE(87,*)TIME/1.D6,CONC_CL_S
** SQO = PQI **
                  WRITE(91,*)TIME/1.D6,PQI
                  WRITE(77,*)TIME/1.D6,AREA_S/1.D9
                  WRITE(78,*)TIME/1.D6,DELDOL_S
                  WRITE(76,*)TIME/1.D6,QI_S
                  WRITE(74,*)TIME/1.D6,SUM_SCL_DEP
                  WRITE(81,*)TIME/1.D6,ALLAREA/1.D9
               ELSE IF(SDC .LT. SDP1 .AND. SDP2 .LT. SDP1)THEN
                  WRTM=TIME
                  WRITE(87,*)TIME/1.D6,CONC_CL_S
** SQO = PQI **
                  WRITE(91,*)TIME/1.D6,PQI
                  WRITE(77,*)TIME/1.D6,AREA_S/1.D9
                  WRITE(78,*)TIME/1.D6,DELDOL_S
                  WRITE(76,*)TIME/1.D6,QI_S
                  WRITE(74,*)TIME/1.D6,SUM_SCL_DEP
                  WRITE(81,*)TIME/1.D6,ALLAREA/1.D9
               ELSE IF(DABS(TIME-TEND).LT.1.D-5)THEN
                  WRITE(87,*)TIME/1.D6,CONC_CL_S
** SQO = PQI **
                  WRITE(91,*)TIME/1.D6,PQI
                  WRITE(77,*)TIME/1.D6,AREA_S/1.D9
                  WRITE(78,*)TIME/1.D6,DELDOL_S
                  WRITE(76,*)TIME/1.D6,QI_S
                  WRITE(74,*)TIME/1.D6,SUM_SCL_DEP
                  WRITE(81,*)TIME/1.D6,ALLAREA /1.D9
               ENDIF
               IF(SU)THEN
                  IF(TIME .LT. SAVETIME .AND. ISTCNT
     +                .LE. NUMST)THEN
                     WRITE(70,*)'CHINA LAKE'
                     WRITE(70,*)'STARTUP TIME #',ISTCNT
                     WRITE(70,*)TIME,X_CS(1),DELDOL_C,
     +                    CONC_CL_C
                     WRITE(70,*)'SEARLES LAKE'
                     WRITE(70,*)TIME,X_CS(3),AREA_S,DELDOL_S,
     +                    CONC_CL_S ,SUM_SCL_DEP
                     WRITE(70,*)'PANAMINT LAKE'
                     WRITE(70,*)TIME,AREA_P,
     +                    CONC_CL_P ,SUM_PCL_DEP
                     WRITE(70,*)'DEATH VALLEY'
                     WRITE(70,*)TIME,AREA_D
                     WRITE(70,*)
                     ISTCNT=ISTCNT+1
                     SAVETIME=SUT(ISTCNT)
```

```
                     ENDIF
                  ENDIF

                  PALLAREA=ALLAREA
                  PCL_P=CL_P
                  PCONC_CL_P=CONC_CL_P
                  PSUM_PCL_DEP=SUM_PCL_DEP
                  PCL_C=CL_C
                  PCONC_CL_C=CONC_CL_C
                  PCDEL_OUT=CDEL_OUT
                  PCL_S=CL_S
                  PCONC_CL_S=CONC_CL_S
                  PSUM_SCL_DEP=SUM_SCL_DEP

                  STEP=STEP+STEP*0.5D0
                  IF(STEP .GT. DTMAX_CS)STEP=DTMAX_CS
                  IF(TIME-STEP.LT.TEND)THEN
                     DTMAX_CS=TIME-TEND
                     STEP=DTMAX_CS
                  ENDIF

                  GOTO 100

**********************************
** MAKE SURE THE SOLVER DOESN'T **
** OVERSTEP DESIGNATED END TIME **
**********************************

               ELSEIF(TIME-STEP.LT.TEND)THEN
                  DTMAX_CS=TIME-TEND
                  STEP=DTMAX_CS
                  GOTO 100
               ENDIF

**************************************
** ADJUST THE SIZE OF THE TIME STEP **
**************************************

            ELSEIF(DELMIN .LE. 0.1)THEN
               STEP=STEP*1.0D-1
            ELSEIF(DELMIN .GE. 4.D0)THEN
               STEP=4.D0*STEP
            ELSE
               STEP=DELMIN*STEP
            ENDIF
            IF(STEP.GT.DTMAX_CS)STEP=DTMAX_CS
            IF(STEP.LT.DTMIN_CS)STEP=DTMIN_CS
         ELSE

            CLOSE(UNIT=92)
            CLOSE(UNIT=91)
            CLOSE(UNIT=87)
            CLOSE(UNIT=85)
            CLOSE(UNIT=84)
            CLOSE(UNIT=83)
            CLOSE(UNIT=81)
            CLOSE(UNIT=78)
            CLOSE(UNIT=77)
            CLOSE(UNIT=76)
            CLOSE(UNIT=75)
            CLOSE(UNIT=74)

            IF( .NOT. GRAF)THEN
               WRITE(6,*)
               WRITE(6,*)'THE RKF SOLVED',KOUNT,' POINTS IN',ITER,' IT
     +ERATIONS'
```

```
          WRITE(6,*)'FOR CHINA AND SEARLES'
          WRITE(6,*)
          WRITE(6,*)'FINAL DEL FOR OWENS =',FINDEL
          WRITE(6,*)'FINAL VOLUME FOR OWENS =',FINVOL/1.D9
          WRITE(6,*)'FINAL CL CONC, OWENS =',FINCL
          WRITE(6,*)
          WRITE(6,*)'FINAL DEL FOR CHINA =',DELDOL_C
          WRITE(6,*)'FINAL VOLUME FOR CHINA =',X_CS(1)/1.D9
          WRITE(6,*)'FINAL CL CONC, CHINA =',CONC_CL_C
          WRITE(6,*)
          WRITE(6,*)'FINAL DEL FOR SEARLES =',DELDOL_S
          WRITE(6,*)'FINAL VOLUME FOR SEARLES =',X_CS(3)/1.D9
          WRITE(6,*)'FINAL AREA FOR SEARLES =',AREA_S/1.D9
          WRITE(6,*)'FINAL CL CONC, SEARLES =',CONC_CL_S
          WRITE(6,*)'TOTAL CL DEPOSITED IN SEARLES',
     +         SUM_SCL_DEP
          WRITE(6,*)
          WRITE(6,*)'FINAL AREA OF PANAMINT',AREA_P/1.D9
          WRITE(6,*)'FINAL CL CONC, PANAMINT',CONC_CL_P
          WRITE(6,*)'TOTAL CL DEPOSITED IN PANAMINT',
     +         SUM_PCL_DEP
          WRITE(6,*)'FINAL AREA OF LAKE MANLY',AREA_D/1.D9
          WRITE(6,*)'FINAL TIME IS:',TIME

          WRITE(55,*)
          WRITE(55,*)'THE RKF SOLVED',KOUNT,' POINTS IN',ITER,' I
     +TERATIONS'
          WRITE(55,*)'FOR CHINA AND SEARLES'
          WRITE(55,*)
          WRITE(55,*)'FINAL DEL FOR OWENS =',FINDEL
          WRITE(55,*)'FINAL VOLUME FOR OWENS =',FINVOL/1.D9
          WRITE(55,*)'FINAL CL CONC, OWENS =',FINCL
          WRITE(55,*)
          WRITE(55,*)'FINAL DEL FOR CHINA =',DELDOL_C
          WRITE(55,*)'FINAL VOLUME FOR CHINA =',X_CS(1)/1.D9
          WRITE(55,*)'FINAL CL CONC, CHINA =',CONC_CL_C
          WRITE(55,*)
          WRITE(55,*)'FINAL DEL FOR SEARLES =',DELDOL_S
          WRITE(55,*)'FINAL VOLUME FOR SEARLES =',X_CS(3)/1.D9
          WRITE(55,*)'FINAL AREA FOR SEARLES =',AREA_S/1.D9
          WRITE(55,*)'FINAL CL CONC, SEARLES =',CONC_CL_S
          WRITE(55,*)'TOTAL CL DEPOSITED IN SEARLES',
     +         SUM_SCL_DEP
          WRITE(55,*)
          WRITE(55,*)'FINAL AREA OF PANAMINT',AREA_P/1.D9
          WRITE(55,*)'FINAL CL CONC, PANAMINT',CONC_CL_P
          WRITE(55,*)'TOTAL CL DEPOSITED IN PANAMINT',
     +         SUM_PCL_DEP
          WRITE(55,*)'FINAL AREA OF LAKE MANLY',AREA_D/1.D9
          WRITE(55,*)'FINAL TIME IS:',TIME

*********************************************************
** WRITE NUMBERS TO FILE TO UPDATE STARTING PARAMETERS **
*********************************************************

          WRITE(56,*)FINDEL,FINVOL,FINCL,DELDOL_C,X_CS(1),
     +    CONC_CL_C,DELDOL_S,X_CS(3),CONC_CL_S,AREA_P,
     +    CONC_CL_P,SUM_PCL_DEP,AREA_D
          WRITE(56,*)
          WRITE(55,*)'TIME IS:',TIME
        ENDIF

        RETURN
      ENDIF

100   ENDWHILE
```

```
**     IF(ITER .GT. ITMAX_CS)WRITE(6,*)'MAX # OF ITERATIONS EXCEEDED'
       IF( .NOT. GRAF)THEN
          WRITE(6,*)
          WRITE(6,*)'THE RKF SOLVED',KOUNT,' POINTS IN',ITER,' IT
     +ERATIONS'
          WRITE(6,*)'FOR CHINA AND SEARLES'
          WRITE(6,*)
          WRITE(6,*)'FINAL DEL FOR OWENS =',FINDEL
          WRITE(6,*)'FINAL VOLUME FOR OWENS =',FINVOL/1.D9
          WRITE(6,*)'FINAL CL CONC, OWENS =',FINCL
          WRITE(6,*)
          WRITE(6,*)'FINAL DEL FOR CHINA =',DELDOL_C
          WRITE(6,*)'FINAL VOLUME FOR CHINA =',X_CS(1)/1.D9
          WRITE(6,*)'FINAL CL CONC, CHINA =',CONC_CL_C
          WRITE(6,*)
          WRITE(6,*)'FINAL DEL FOR SEARLES =',DELDOL_S
          WRITE(6,*)'FINAL VOLUME FOR SEARLES =',X_CS(3)/1.D9
          WRITE(6,*)'FINAL AREA FOR SEARLES =',AREA_S/1.D9
          WRITE(6,*)'FINAL CL CONC, SEARLES =',CONC_CL_S
          WRITE(6,*)'TOTAL CL DEPOSITED IN SEARLES',
     +          SUM_SCL_DEP
          WRITE(6,*)
          WRITE(6,*)'FINAL AREA OF PANAMINT',AREA_P/1.D9
          WRITE(6,*)'FINAL CL CONC, PANAMINT',CONC_CL_P
          WRITE(6,*)'TOTAL CL DEPOSITED IN PANAMINT',
     +          SUM_PCL_DEP
          WRITE(6,*)'FINAL AREA OF LAKE MANLY',AREA_D/1.D9
          WRITE(6,*)'FINAL TIME IS:',TIME

          WRITE(55,*)
          WRITE(55,*)'THE RKF SOLVED',KOUNT,' POINTS IN',ITER,' ITERATI
     +ONS'
          WRITE(55,*)'FOR CHINA AND SEARLES'
          WRITE(55,*)
          WRITE(55,*)'FINAL DEL FOR OWENS =',FINDEL
          WRITE(55,*)'FINAL VOLUME FOR OWENS =',FINVOL/1.D9
          WRITE(55,*)'FINAL CL CONC, OWENS =',FINCL
          WRITE(55,*)
          WRITE(55,*)'FINAL DEL FOR CHINA =',DELDOL_C
          WRITE(55,*)'FINAL VOLUME FOR CHINA =',X_CS(1)/1.D9
          WRITE(55,*)'FINAL CL CONC, CHINA =',CONC_CL_C
          WRITE(55,*)
          WRITE(55,*)'FINAL DEL FOR SEARLES =',DELDOL_S
          WRITE(55,*)'FINAL VOLUME FOR SEARLES =',X_CS(3)/1.D9
          WRITE(55,*)'FINAL AREA FOR SEARLES =',AREA_S/1.D9
          WRITE(55,*)'FINAL CL CONC, SEARLES =',CONC_CL_S
          WRITE(55,*)'TOTAL CL DEPOSITED IN SEARLES',
     +                SUM_SCL_DEP
          WRITE(55,*)
          WRITE(55,*)'FINAL AREA OF PANAMINT',AREA_P/1.D9
          WRITE(55,*)'FINAL CL CONC, PANAMINT',CONC_CL_P
          WRITE(55,*)'TOTAL CL DEPOSITED IN PANAMINT',
     +                SUM_PCL_DEP
          WRITE(55,*)'FINAL AREA OF LAKE MANLY',AREA_D/1.D9
          WRITE(55,*)'FINAL TIME IS:',TIME

*************************************************************
** WRITE NUMBERS TO FILE TO UPDATE STARTING PARAMETERS **
*************************************************************

          WRITE(56,*)FINDEL,FINVOL,FINCL,DELDOL_C,X_CS(1),
     +          CONC_CL_C,DELDOL_S,X_CS(3),CONC_CL_S,AREA_P,
     +          CONC_CL_P,SUM_PCL_DEP,AREA_D
          WRITE(56,*)
          WRITE(55,*)'TIME IS:',TIME
       ENDIF
```

```
      END


*****************************************************************************
*****************************************************************************
** A FUNCTION TO CALCULATE DV/DT AND DDEL/DT FOR CHINA AND SEARLES LAKE **
*****************************************************************************
*****************************************************************************

      DOUBLE PRECISION FUNCTION F_CS(I,TERM,T,KNT,KOUNT,TBEG,COAL)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      LOGICAL FOUND,GUESS,GRAF,CPARAM,GUESS2,FOUND2,COAL,SU

      PARAMETER(NUMA=25000,NUMB=2000)
      PARAMETER(VOLMAX_C=0.69609,VOLMAX_S=85.2809)

      SAVE

      COMMON/BOTH/CL(NUMA),TSOD(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

      COMMON/BOTH2/CL_C,TSOD_C(10),TCL_C(10),TIME,TCO3_C(10),
     + CQO(10),GRAF,CONC_CL_C,DOLTEMP_C,DELDOL_C,
     + TCDELI,TCTEMP,CDEL_OUT,CCL_OUT,CL_S,NOPTS,
     + TSOD_S(10),TCL_S(10),TCO3_S(10),SQO(10),CONC_CL_S,
     + DOLTEMP_S,DELDOL_S,TSDELI,TSTEMP,SCL_DEP,TTLCL_IN,NPTS,
     + SUM_SCL_DEP,QI_C,QI_S,PQI

      COMMON/PREV/PCL_P,PCONC_CL_P,PSUM_PCL_DEP,PCL_C,PCONC_CL_C,
     + PCDEL_OUT,PCL_S,PCONC_CL_S,PSUM_SCL_DEP,PALLAREA,DATSAV,
     + SDC,SDP1,SDP2

      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S

     - COMMON/NEW/OPREV,CPREV,SPREV,PPREV,DPREV

      COMMON/HIST/QIHIST(1000),HUMTIME(500),NHPTS,SU,SUT(15),
     + SAVETIME,NUMST,QITIME(1000),NQPTS

      DIMENSION TERM(4),TCONC_CL_C(10),V_OUT_C(10),
     + TCONC_CL_S(10),V_OUT_S(10)

      IF(KOUNT .EQ. 1)THEN
         ALLAREA=PALLAREA
         CL_C=PCL_C
         CONC_CL_C=PCONC_CL_C
         CDEL_OUT=PCDEL_OUT
         CL_S=PCL_S
         CONC_CL_S=PCONC_CL_S
         SUM_SCL_DEP=PSUM_SCL_DEP
```

```
        ENDIF

        IF(I.EQ.1)THEN

 *************************************
 ** CALCULATE DV/DT FOR CHINA LAKE **
 *************************************

           VOL_C = TERM(1)
           IF(VOL_C .LE. 10.D0)VOL_C=0.D0


 ******************************
 ** CONSTANT PARAMETER OPTION **
 ******************************

           IF(CPARAM)THEN
              TCTEMP=TEMPC_C
              TCEVAP=EVAPC_C
              TCPRECIP=PRECIPC_C
              TCDELP=DELPC_C
              TCDELA=DELAC_C
              CALL FINDHT(NHPTS,T,IHNDEX,HUMTIME)
              CALL HUMTERP(T,IHNDEX,TCHUM,CHUM,HUMTIME)
           ELSE

 **********************************************************************
 ** DETERMINE VALUES OF NECESSARY PARAMETERS BY ASSIGNING VALUES OR  **
 ** INTERPOLATING BETWEEN GIVEN VALUES                               **
 **********************************************************************

              CALL FINDT(NPTS,T,INDEX,FOUND,WTIME)
              CALL FINDHT(NHPTS,T,IHNDEX,HUMTIME)


              IF(FOUND)THEN
                 TCTEMP = CTEMP(INDEX)
                 TCEVAP = CEVAP(INDEX)
                 TCPRECIP = CPRECIP(INDEX)
                 TCDELP = CDELP(INDEX)
                 TCDELA = CSDELA(INDEX)
              ELSE
                 CALL CINTERP(T,INDEX,TCTEMP,TCEVAP,TCPRECIP,
     +                 TCDELP,TCDELA)
              ENDIF
              CALL HUMTERP(T,IHNDEX,TCHUM,CHUM,HUMTIME)
           ENDIF

 *****************************************************************************
 ** INTERPOLATE TO FIND VALUES FOR PARAMETERS CALCULATED DURING OWENS ROUTINE **
 *****************************************************************************

           CALL FINDT2(NOPTS,T,INDEX2,FOUND2,OTIME)
           IF(FOUND2)THEN
              QI=OQO(INDEX2)
              TCDELI=ODEL_OUT(INDEX2)
              TAREA=AREA(INDEX2)
           ELSE
              CALL C2INTERP(T,INDEX2,TCDELI,QI,TAREA)
           ENDIF

           IF(QI .LT. 0.D0)QI=0.D0

 ******************************************************************
 ** "REMEMBER" TEMP AT END OF TIMESTEP TO CALCULATE DEL DOLOMITE **
 ******************************************************************
```

```
          IF(KNT .EQ. 5)DOLTEMP_C=TCTEMP

          IF(KNT .EQ. 5)SOAREA=TAREA

************************************************************
** "REMEMBER" QI AT END OF TIMESTEP FOR GRAPHICS ARRAY **
************************************************************

          IF(KNT .EQ. 5)QI_C=QI

          ETIME = TBEG-T

*****************************
** CALCULATE AREA OF LAKE **
*****************************

          AREA_C = CAREA(VOL_C)

******************************
** CALCULATE PRECIPITATION **
******************************

          QP = AREA_C*TCPRECIP

**********************************
** THE INFAMOUS "SALT" BALANCE **
**********************************

          IF(DABS(T-TIME).LT.1.D-8)THEN
             CALL C3INTERP(T,INDEX2,TIMECL_IN)
             TCL_C(KNT)=CL_C
             TCONC_CL_C(KNT)=CONC_CL_C

****************************************************
** CALCULATE TOTAL OUTFLOW VOLUME FROM TIME TO T **
** AND ADJUST IF VOL EXCEEDS VOLMAX             **
****************************************************

          ELSE
             IF(VOL_C .LE. VOLMAX_C)THEN
                V_OUT_C(KNT)=0.D0
             ELSE
                V_OUT_C(KNT) = VOL_C-VOLMAX_C
                VOL_C=VOLMAX_C
             ENDIF

**********************************************************
**  CALCULATE CONCENTRATIONS FOR INTERMEDIATE TIME "T" **
**********************************************************

*******************************************************
** CALCULATE HOW MUCH 'SALT' CAME IN FROM OWENS LAKE **
*******************************************************

             CALL C3INTERP(T,INDEX2,CL_IN)
             TCL_IN=CL_IN-TIMECL_IN
             IF(KNT .EQ. 5)TTLCL_IN=TCL_IN
             TCL_C(KNT) = TCL_IN+CL_C -V_OUT_C(KNT)*
     +            CONC_CL_C *1.D3
             IF(TCL_C(KNT).LT.0.D0)TCL_C(KNT)=3.95D-4*0.696D9*1.D3

*********************************************************************
** CONCENTRATION UNITS ARE MOLARITY SO VOL MUST BE MULT BY 1000 TO **
** CONVERT M^3 TO LITERS **
*********************************************************************
```

```
            IF(VOL_C .GT. 10.D0)THEN
               TCONC_CL_C(KNT)=TCL_C(KNT)/(VOL_C*1.0D3)
            ELSE
               TCONC_CL_C(KNT)=0.D0
               TCL_C(KNT)=0.D0
            ENDIF

************************************
** CHECK FOR CHLORIDE SATURATION **
************************************

            IF(TCONC_CL_C(KNT) .GT. 6.1D0)THEN
               TCONC_CL_C(KNT)=6.1D0
               TCL_C(KNT)=6.1D0*(VOL_C*1.D3)
            ENDIF

***************************************************************
** CALCULATE HOW MUCH SALT GOES TO SEARLES FROM TIME TO T **
***************************************************************


            TCCL_OUT=V_OUT_C(KNT)*1.D3*CONC_CL_C

         ENDIF

*********************************************
**  CALCULATE BACK-CONDENSATION FLUX (QC) **
*********************************************

         IF(VOL_C .GT. 10.D0 .AND. TCONC_CL_C(KNT) .GT. 0.D0)THEN
            PHI=FPHI(TCONC_CL_C(KNT),SUM)
         ELSE
            PHI=0.D0
         ENDIF
         AW=DEXP(-18.D0*PHI*SUM*0.5D0/1.D3)

********************************
** CALCULATE EVAPORATION (QE) **
********************************

         QE = AREA_C*TCEVAP*AW

         IF(VOL_C .GT. 10.D0)THEN
            QC=(TCHUM*QE)/AW
         ELSE
            QC=0.D0
         ENDIF

*************************************
**  CALCULATE DV/DT IF VOL IS ZERO **
*************************************

         IF(TERM(1) .LE. 10.D0)THEN
            IF(GUESS)THEN
               DELDOL_C=FDDOL(TCTEMP,TERM(2))
               GUESS=.FALSE.
            ENDIF

            CQO(KNT)=0.D0

            F_CS=QI

            IF(F_CS .LE. 0.D0)THEN
               DV_DT=0.D0
            ELSE
```

```
                DV_DT=F_CS
             ENDIF

*****************************************************
**  CALCULATE DV/DT IF VOL IS < VOLMAX BUT > 0 **
*****************************************************

          ELSE IF(TERM(1)  .LT. VOLMAX_C)THEN

***********************
**  CALCULATE DV/DT **
***********************

             F_CS=QI+QC-QE+QP
             DV_DT=F_CS
             CQO(KNT)=0.D0

*******************************************************
**  CALCULATE DV/DT IF VOL IS GREATER THAN VOLMAX **
*******************************************************

          ELSE

*******************
** CALCULATE QO **
*******************

            CQO(KNT)=QI+QC-QE+QP

*********************
** CALCULATE DV/DT **
*********************

             IF(CQO(KNT) .LT. 0.D0)THEN
                CQO(KNT)=0.D0
                F_CS=QI+QC-QE+QP
                DV_DT=F_CS
             ELSE
                F_CS=CQO(KNT)
                DV_DT=0.D0
             ENDIF
          ENDIF

***************************************
** CALCULATE DDEL/DT FOR CHINA LAKE **
***************************************

        ELSE IF(I.EQ.2)THEN

          DELL_C = TERM(2)

          IF(VOL_C .LE. 10.D0)THEN
             F_CS=0.D0

          ELSE

*******************************************
** CALCULATE ISOTOPIC ENRICHMENT FACTOR **
*******************************************

            EPS = FEPS(TCTEMP)

**********************************************
** CALCULATE DEL OF THE BACK-CONDENSATION **
**********************************************
```

```
            CDELC =EPS*(1.D0+(TCDELA/1.D3))+TCDELA

*****************************************
** CALCULATE DEL OF THE EVAPORATION **
*****************************************

            CDELE = DELE(DELL_C,EPS,TCHUM)

*********************************************************
** SET DEL OF THE OUTFLOW EQUAL TO DEL OF THE LAKE **
*********************************************************

            CDELO = DELL_C

            F_CS=(QI*TCDELI+QC*CDELC+QP*TCDELP-CQO(KNT)*CDELO-QE*
     +              CDELE-DELL_C*DV_DT)/VOL_C

          ENDIF

        ELSE IF(I.EQ.3)THEN

*****************************************
** CALCULATE DV/DT FOR SEARLES LAKE **
*****************************************

          VOL_S = TERM(3)
          IF(VOL_S .LE. 10.D0)VOL_S=0.D0

*********************************
** CONSTANT PARAMETER OPTION **
*********************************

          IF(CPARAM)THEN
             TSTEMP=TEMPC_S
             TSEVAP=EVAPC_S
             TSPRECIP=PRECIPC_S
             TSDELP=DELPC_S
             TSDELA=DELAC_S
             IF(COAL)THEN
                CALL FINDT(NPTS,T,INDEX,FOUND,WTIME)
                CALL FINDHT(NHPTS,T,IHNDEX,HUMTIME)
             ENDIF
             CALL HUMTERP(T,IHNDEX,TSHUM,SHUM,HUMTIME)
          ELSE

*************************************************************************
** DETERMINE VALUES OF NECESSARY PARAMETERS BY ASSIGNING VALUES OR   **
** INTERPOLATING BETWEEN GIVEN VALUES                                 **
*************************************************************************

             IF(COAL)THEN
                CALL FINDT(NPTS,T,INDEX,FOUND,WTIME)
                CALL FINDHT(NHPTS,T,IHNDEX,HUMTIME)
             ENDIF

             IF(FOUND)THEN
                TSTEMP = STEMP(INDEX)
                TSEVAP = SEVAP(INDEX)
                TSPRECIP = SPRECIP(INDEX)
                TSDELP = SDELP(INDEX)
             ELSE

**********************************************************************************
** INTERPOLATE TO FIND VALUES FOR PARAMETERS CALCULATED DURING OWENS ROUTINE **
**********************************************************************************
```

```
            CALL SINTERP(T,INDEX,TSTEMP,TSEVAP,TSPRECIP,
     +                TSDELP,TSDELA)

         ENDIF
         CALL HUMTERP(T,IHNDEX,TSHUM,SHUM,HUMTIME)
      ENDIF

*************************************************************************
** THESE SEARLES PARAMETERS ARE EQUAL TO THEIR CHINA COUNTERPARTS **
*************************************************************************

      IF(COAL)THEN
         CALL FINDT2(NOPTS,T,INDEX2,FOUND2,OTIME)
         IF(FOUND2)THEN
            QI=OQO(INDEX2)
            TSDELI=ODEL_OUT(INDEX2)
            TAREA=AREA(INDEX2)
         ELSE
            CALL C2INTERP(T,INDEX2,TSDELI,QI,TAREA)
         ENDIF
         IF(QI .LT. 0.D0)QI=0.D0
      ELSE
         TSDELI = CDELO
      ENDIF

**********************************************************************
** "REMEMBER" TEMP AT END OF TIMESTEP TO CALCULATE DEL DOLOMITE **
**********************************************************************

      IF(KNT .EQ. 5)DOLTEMP_S=TSTEMP

      IF(KNT .EQ. 5)SOAREA=TAREA

      ETIME = TBEG-T

*****************************
** CALCULATE AREA OF LAKE **
*****************************

      AREA_S = SAREA(VOL_S)

******************************
** CALCULATE PRECIPITATION **
******************************

      QP = AREA_S*TSPRECIP

************************
** QI = QO FROM CHINA **
************************

      IF(.NOT. COAL)THEN
         QI =CQO(KNT)
      ENDIF

      IF(KNT .EQ. 5)QI_S=QI

**********************************
** THE INFAMOUS "SALT" BALANCE **
**********************************

      IF(DABS(T-TIME).LT.1.D-8)THEN

*****************************************************************
** IF CHINA AND SEARLES COALESCE, THE INITIAL SALT BALANCE **
** IS CALCULATED IN THE RKF_CS SOLVER                       **
```

```
**************************************************************

              TCL_S(KNT)=CL_S
              TCONC_CL_S(KNT)=CONC_CL_S

****************************************************
** CALCULATE TOTAL OUTFLOW VOLUME FROM TIME TO T **
** AND ADJUST IF VOL EXCEEDS VOLMAX              **
****************************************************

           ELSE
              IF(VOL_S .LE. VOLMAX_S)THEN
                 V_OUT_S(KNT)=0.D0
              ELSE
                 V_OUT_S(KNT) = VOL_S-VOLMAX_S
                 VOL_S=VOLMAX_S
              ENDIF

**********************************************************
**  CALCULATE CONCENTRATIONS FOR INTERMEDIATE TIME "T" **
**********************************************************

********************************************************
** CALCULATE HOW MUCH 'SALT' CAME IN FROM CHINA LAKE **
********************************************************

              IF(COAL)THEN
                 CALL C3INTERP(TIME,INDEX2,TIMECL_IN)
                 CALL C3INTERP(T,INDEX2,CL_IN)
                 TCL_IN=CL_IN-TIMECL_IN
                 IF(KNT .EQ. 5)TTLCL_IN=TCL_IN
              ELSE
                 TCL_IN=TCCL_OUT
              ENDIF

              TCL_S(KNT) = TCL_IN+CL_S -V_OUT_S(KNT)*
     +             CONC_CL_S *1.D3

              IF(TCL_S(KNT).LT.0.D0)TCL_S(KNT)=3.95D-4*85.2809*1.D3

**********************************************************************
** CONCENTRATION UNITS ARE MOLARITY SO VOL MUST BE MULT BY 1000 TO **
** CONVERT M^3 TO LITERS **
**********************************************************************

              IF(VOL_S .GT. 10.D0)THEN
                 TCONC_CL_S(KNT)=TCL_S(KNT)/(VOL_S*1.0D3)
              ELSE
                 TCONC_CL_S(KNT)=0.D0
              ENDIF

************************************
** CHECK FOR CHLORIDE SATURATION **
************************************

              IF(TCONC_CL_S(KNT) .GT. 6.1D0)THEN
                 TCONC_CL_S(KNT)=6.1D0
                 TCL_S(KNT)=6.1D0*(VOL_S*1.D3)
              ENDIF

           ENDIF

*********************************************
**  CALCULATE BACK-CONDENSATION FLUX (QC) **
*********************************************
```

```
          IF(VOL_S .GT. 10.D0 .AND. TCONC_CL_S(KNT) .GT. 0.D0)THEN
             PHI=FPHI(TCONC_CL_S(KNT),SUM)
          ELSE
             PHI=0.D0
          ENDIF
          AW=DEXP(-18.D0*PHI*SUM*0.5D0/1.D3)

*********************************
** CALCULATE EVAPORATION (QE) **
*********************************

          QE = AREA_S*TSEVAP*AW

          IF(VOL_S .GT. 10.D0)THEN
             QC=(TSHUM*QE)/AW
          ELSE
             QC=0.D0
          ENDIF

**************************************
**  CALCULATE DV/DT IF VOL IS ZERO **
**************************************

          IF(TERM(3) .LE. 10.D0)THEN
             IF(GUESS2)THEN
                WRITE(77,*)TBEG/1.D6,AREA_S/1.D9
                WRITE(76,*)TBEG/1.D6,QI
                WRITE(91,*)TBEG/1.D6,0.0
                DELDOL_S=FDDOL(TSTEMP,TERM(4))
                WRITE(78,*)TBEG/1.D6,DELDOL_S
                GUESS2=.FALSE.
                PALLAREA=AREA(1)+AREA_S+AREA_C+AREA_P+AREA_D
                OPREV=AREA(1)
                CPREV=AREA_C
                SPREV=AREA_S
                PPREV=AREA_P
                DPREV=AREA_P
                WRITE(81,*)TBEG/1.D6,PALLAREA/1.D9
             ENDIF

             SQO(KNT)=0.D0

****************************************************
** SET SEARLES OUTFLUX EQUAL TO PANAMINT INFLUX **
****************************************************

             IF(KNT .EQ. 5)THEN
                PQI=SQO(KNT)
             ENDIF

             IF(QI .LT. 0.D0)QI=0.D0

             F_CS=QI

             IF(F_CS .LE. 0.D0)THEN
                DV_DT=0.D0
             ELSE
                DV_DT=F_CS
             ENDIF

****************************************************
**  CALCULATE DV/DT IF VOL IS < VOLMAX BUT > 0 **
****************************************************

         ELSE IF(TERM(3)  .LT. VOLMAX_S)THEN
```

```
**************************************
** WRITE INITIAL VALUES TO FILE **
**************************************

               IF(GUESS2)THEN
                  WRITE(77,*)TBEG/1.D6,AREA_S/1.D9
                  WRITE(76,*)TBEG/1.D6,QI
                  WRITE(91,*)TBEG/1.D6,0.0
                  DELDOL_S=FDDOL(TSTEMP,TERM(4))
                  WRITE(78,*)TBEG/1.D6,DELDOL_S
                  PALLAREA=AREA(1)+AREA_S+AREA_C+AREA_P+AREA_D
                  OPREV=AREA(1)
                  CPREV=AREA_C
                  SPREV=AREA_S
                  PPREV=AREA_P
                  DPREV=AREA_D
                  GUESS2=.FALSE.
                  WRITE(81,*)TBEG/1.D6,PALLAREA/1.D9
               ENDIF

************************
**  CALCULATE DV/DT **
************************

               F_CS=QI+QC-QE+QP
               DV_DT=F_CS
               SQO(KNT)=0.D0
               IF(KNT .EQ. 5)THEN
                  PQI=SQO(KNT)
               ENDIF

*******************************************************
**  CALCULATE DV/DT IF VOL IS GREATER THAN VOLMAX **
*******************************************************

            ELSE

               IF(GUESS2)THEN
                  WRITE(77,*)TBEG/1.D6,AREA_S/1.D9
                  WRITE(76,*)TBEG/1.D6,QI
                  WRITE(91,*)TBEG/1.D6,QI+QC-QE+QP
                  DELDOL_S=FDDOL(TSTEMP,TERM(4))
                  WRITE(78,*)TBEG/1.D6,DELDOL_S
                  PALLAREA=AREA(1)+AREA_S+AREA_C+AREA_P+AREA_D
                  GUESS2=.FALSE.
                  OPREV=AREA(1)
                  CPREV=AREA_C
                  SPREV=AREA_S
                  PPREV=AREA_P
                  DPREV=AREA_D
                  WRITE(81,*)TBEG/1.D6,PALLAREA/1.D9
               ENDIF

*******************
** CALCULATE QO **
*******************

               SQO(KNT)=QI+QC-QE+QP

**********************
** CALCULATE DV/DT **
**********************
```

```
            IF(SQO(KNT) .LT. 0.DO)THEN
               SQO(KNT)=0.DO
               IF(KNT .EQ. 5)PQI=SQO(KNT)
               F_CS=QI+QC-QE+QP
               DV_DT=F_CS
            ELSE
               F_CS=SQO(KNT)
               IF(KNT .EQ. 5)THEN
                  PQI=SQO(KNT)
               ENDIF
               DV_DT=0.DO
            ENDIF
         ENDIF

****************************************
** CALCULATE DDEL/DT FOR SEARLES LAKE **
****************************************

        ELSE IF(I.EQ.4)THEN

           DELL_S = TERM(4)

           IF(VOL_S .LE. 10.DO)THEN
             F_CS=0.DO

           ELSE

********************************************
** CALCULATE ISOTOPIC ENRICHMENT FACTOR **
********************************************

              EPS = FEPS(TSTEMP)

**********************************************
** CALCULATE DEL OF THE BACK-CONDENSATION **
**********************************************

              SDELC =EPS*(1.DO+(TSDELA/1.D3))+TSDELA

****************************************
** CALCULATE DEL OF THE EVAPORATION **
****************************************

              SDELE = DELE(DELL_S,EPS,TSHUM)

*******************************************************
** SET DEL OF THE OUTFLOW EQUAL TO DEL OF THE LAKE **
*******************************************************

              SDELO = DELL_S

              F_CS=(QI*TSDELI+QC*SDELC+QP*TSDELP-SQO(KNT)*SDELO-QE*
     +              SDELE-DELL_S*DV_DT)/VOL_S

           ENDIF

        END IF
        RETURN
        END


*******************************************************************************
*******************************************************************************
**          A FUNCTION SUBPROGRAM TO CALCULATE THE AREA OF OWENS LAKE        **
*******************************************************************************
```

```
******************************************************************************

      DOUBLE PRECISION FUNCTION OAREA (VOL)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      TVOL=VOL/1.0D9

      IF(TVOL .GE. 0.0D0 .AND. TVOL .LT. 0.16D0) THEN
         OAREA = 1.25D0*TVOL
      ELSE IF (TVOL .GE. 0.16D0 .AND. TVOL .LT. 3.15D0) THEN
         OAREA = 3.01D-2*(TVOL-0.16D0)+0.2D0
      ELSE IF(TVOL .GE. 3.15D0 .AND. TVOL .LT. 30.02D0) THEN
         OAREA = 1.5035D-2*(TVOL-3.15D0)+0.29D0
      ELSE
         OAREA = 0.694D0
      END IF
      OAREA=OAREA*1.0D9
      RETURN
      END


******************************************************************************
******************************************************************************
**        A FUNCTION SUBPROGRAM TO CALCULATE THE AREA OF CHINA LAKE        **
******************************************************************************
******************************************************************************

      DOUBLE PRECISION FUNCTION CAREA (VOL)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      TVOL=VOL/1.0D9
      IF(TVOL .GE. 0.0D0 .AND. TVOL .LT. 0.036D0) THEN
         CAREA = 0.75D0*TVOL
      ELSE IF(TVOL .GE. 0.036D0 .AND. TVOL .LT. 0.696D0) THEN
         CAREA =(.128D0/.66D0)*(TVOL-0.036D0)+0.027D0
      ELSE
         CAREA = 0.155D0
      END IF
      CAREA=CAREA*1.0D9
      RETURN
      END


******************************************************************************
******************************************************************************
**       A FUNCTION SUBPROGRAM TO CALCULATE THE AREA OF SEARLES LAKE       **
******************************************************************************
******************************************************************************

      DOUBLE PRECISION FUNCTION SAREA (VOL)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      TVOL=VOL/1.0D9

      IF(TVOL .GE. 0.0D0 .AND. TVOL .LT. 2.04D0) THEN
         SAREA =(0.245D0/2.04D0)*TVOL
      ELSE IF(TVOL .GE. 2.04D0 .AND. TVOL .LT. 10.75D0) THEN
         SAREA=(0.055D0/8.71D0)*(TVOL-2.04D0)+0.245D0
      ELSE IF(TVOL .GE. 10.75D0 .AND. TVOL .LT. 20.82D0) THEN
         SAREA=(0.05D0/10.07D0)*(TVOL-10.75D0)+0.3D0
      ELSE IF(TVOL .GE. 20.82D0 .AND. TVOL .LT. 33.46D0) THEN
         SAREA=(0.092D0/12.64D0)*(TVOL-20.82D0)+0.35D0
      ELSE IF(TVOL .GE. 33.46D0 .AND. TVOL .LT. 46.6D0) THEN
         SAREA=(0.091D0/13.14D0)*(TVOL-33.46D0)+0.442D0
```

```
      ELSE IF(TVOL .GE. 46.6D0 .AND. TVOL .LT. 65.87D0) THEN
         SAREA=(0.182D0/19.27D0)*(TVOL-46.6D0)+0.533D0
      ELSE IF(TVOL .GE. 65.87D0 .AND. TVOL .LT. 85.28D0) THEN
         SAREA=(0.124D0/19.41D0)*(TVOL-65.87D0)+0.87D0
      ELSE
         SAREA=0.994D0
      ENDIF
      SAREA=SAREA*1.0D9
      RETURN
      END


*********************************************************************************
*********************************************************************************
**    A FUNCTION SUBPROGRAM TO CALCULATE THE VOLUME OF PANAMINT LAKE          **
*********************************************************************************
*********************************************************************************

      DOUBLE PRECISION FUNCTION PVOL(AREA)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      TAREA=AREA/1.0D9

      IF(TAREA .GE. 0.0D0 .AND. TAREA .LT. 0.118D0) THEN
         PVOL =(0.71D0/.118D0)*TAREA
      ELSE IF(TAREA .GE. 0.118D0 .AND. TAREA .LT. 0.175D0) THEN
         PVOL=(2.91D0/0.057D0)*(TAREA-0.118D0)+0.71D0
      ELSE IF(TAREA .GE. 0.175D0 .AND. TAREA .LT. 0.189D0) THEN
         PVOL=(2.D0/0.014D0)*(TAREA-0.175D0)+3.62D0
      ELSE IF(TAREA .GE. 0.189D0 .AND. TAREA .LT. 0.242D0) THEN
         PVOL=(6.45D0/0.53D0)*(TAREA-0.189D0)+5.62D0
      ELSE IF(TAREA .GE. 0.242D0 .AND. TAREA .LT. 0.289D0) THEN
         PVOL=(8.22D0/0.047D0)*(TAREA-0.242D0)+12.07D0
      ELSE IF(TAREA .GE. 0.289D0 .AND. TAREA .LT. 0.329D0) THEN
         PVOL=(9.26D0/0.04D0)*(TAREA-0.289D0)+20.29D0
      ELSE IF(TAREA .GE. 0.329D0 .AND. TAREA .LT. 0.369D0) THEN
         PVOL=(10.81D0/0.04D0)*(TAREA-0.329D0)+29.55D0
      ELSE IF(TAREA .GE. 0.369D0 .AND. TAREA .LT. 0.428D0) THEN
         PVOL=(13.93D0/0.059D0)*(TAREA-0.369D0)+40.36D0
      ELSE IF(TAREA .GE. 0.428D0 .AND. TAREA .LT. 0.488D0) THEN
         PVOL=(9.15D0/0.06D0)*(TAREA-0.428D0)+54.29D0
      ELSE IF(TAREA .GE. 0.488D0 .AND. TAREA .LT. 0.524D0) THEN
         PVOL=(7.59D0/0.036D0)*(TAREA-0.488D0)+63.44D0
      ELSE IF(TAREA .GE. 0.524D0 .AND. TAREA .LT. 0.568D0) THEN
         PVOL=(10.92D0/0.044D0)*(TAREA-0.524D0)+71.03D0
      ELSE IF(TAREA .GE. 0.568D0 .AND. TAREA .LT. 0.638D0) THEN
         PVOL=(15.67D0/0.07D0)*(TAREA-0.568D0)+81.95D0
      ELSE IF(TAREA .GE. 0.638D0 .AND. TAREA .LT. 0.727D0) THEN
         PVOL=(10.23D0/0.089D0)*(TAREA-0.638D0)+97.62D0
      ELSE
         PVOL=107.85D0
      ENDIF
      PVOL=PVOL*1.0D9
      RETURN
      END


*********************************************************************************
*********************************************************************************
**       A FUNCTION SUBPROGRAM TO CALCULATE THE VOLUME OF MANLY LAKE          **
*********************************************************************************
*********************************************************************************

      DOUBLE PRECISION FUNCTION DVOL(AREA)
```

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      TAREA=AREA/1.009

      IF(TAREA .GE. 0.0D0 .AND. TAREA .LT. 0.05D0) THEN
         DVOL =(0.65D0/.05D0)*TAREA
      ELSE IF(TAREA .GE. 0.05D0 .AND. TAREA .LT. 47.0D0) THEN
         DVOL=(46.35D0/0.533D0)*(TAREA-0.05D0)+0.65D0
      ELSE
         DVOL=47.D0
      ENDIF

      DVOL=DVOL*1.009

      RETURN
      END


******************************************************************************
******************************************************************************
*  THIS IS A FUNCTION SUBPROGRAM TO CALCULATE THE ISOTOPIC ENRICHMENT FACTOR  *
******************************************************************************
******************************************************************************

      DOUBLE PRECISION FUNCTION FEPS(TEMP)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)

**************************
**CONVERT TEMP TO KELVIN**
**************************

      TEMPK=TEMP+273.15D0
      FEPS =(DEXP((1.534D0*(1.0D6/(TEMPK)**2)-3.206D0*(1.0D3/TEMPK)+
     +      2.644D0)/1.D3)-1.D0)*1.D3
      RETURN
      END


******************************************************************************
******************************************************************************
*      THIS IS A FUNCTION SUBPROGRAM TO CALCULATE THE OSMOTIC COEFFICIENT     *
******************************************************************************
******************************************************************************

      DOUBLE PRECISION FUNCTION FPHI(CL_M,SUM)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

*********************************
**CALCULATE MOLARITIES FOR IONS**
*********************************

      SOD_M=CL_M
      SUM=CL_M+SOD_M

**************************************
**CALCULATE OSMOTIC COEFFICIENT, PHI**
**************************************

      XI = (SOD_M+CL_M+4.D0)/2.D0
      XF = 0.392D0*(DSQRT(XI)/(1.D0+1.2D0*DSQRT(XI)))
      BCL = 0.0765D0 + 0.2664D0 * DEXP(-2.D0*DSQRT(XI))
      BCO3 = 0.18975D0+0.846D0*DEXP(-2.D0*DSQRT(XI))
      DSUM1 = 2.D0*SOD_M*CL_M*(BCL+SOD_M*0.00127D0)
```

```
      DSUM2 = 2.D0*SOD_M*(BCO3-0.048032D0*SOD_M/DSQRT(2.D0))
      FPHI = 1.41210D0*(1.D0+(1.D0/SUM*(2.D0*XI*XF+DSUM1+DSUM2)))
      RETURN
      END


***************************************************************************
***************************************************************************
*     A SUBROUTINE TO LINEARLY INTERPOLATE BETWEEN POINTS IN DATA SETS    *
*                   CONTAINING OWENS LAKE PARAMETERS                      *
***************************************************************************
***************************************************************************

      SUBROUTINE OINTERP(TIME,NDX,TODELI,TOTEMP,TOEVAP,TOPRECIP,
     +     TODELP,TODELA)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      LOGICAL GUESS,CPARAM,GUESS2

      PARAMETER(NUMB=2000)
      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S

      TODELI=(ODELI(NDX+1)-ODELI(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +     (TIME-WTIME(NDX))+ODELI(NDX)

      TOTEMP=(OTEMP(NDX+1)-OTEMP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +     (TIME-WTIME(NDX))+OTEMP(NDX)

      TOEVAP=(OEVAP(NDX+1)-OEVAP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +     (TIME-WTIME(NDX))+OEVAP(NDX)

      TOPRECIP=(OPRECIP(NDX+1)-OPRECIP(NDX))/(WTIME(NDX+1)-
     +     WTIME(NDX))*(TIME-WTIME(NDX))+OPRECIP(NDX)

      TODELP = (ODELP(NDX+1)-ODELP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +     (TIME-WTIME(NDX))+ODELP(NDX)

      TODELA = (ODELA(NDX+1)-ODELA(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +     (TIME-WTIME(NDX))+ODELA(NDX)

      RETURN
      END


***************************************************************************
***************************************************************************
*         A SUBROUTINE TO CALCULATE HUMIDITY FOR ALL OF THE LAKES        **
***************************************************************************
***************************************************************************


      SUBROUTINE HUMTERP(TIME,NDX,T_HUM,HUM,HUMTIME)

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
```

```
      PARAMETER(NUM=500)
      DOUBLE PRECISION HUM(NUM),HUMTIME(NUM)

      T_HUM=(HUM(NDX+1)-HUM(NDX))/(HUMTIME(NDX+1)-HUMTIME(NDX))*
     +      (TIME-HUMTIME(NDX))+HUM(NDX)


      RETURN
      END



******************************************************************************
******************************************************************************
*          A SUBROUTINE TO CALCULATE INFLOW FROM HISTORY                    **
******************************************************************************
******************************************************************************


      SUBROUTINE QINTERP(TIME,NDX,QI,QHIST,QITIME)

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)

      DOUBLE PRECISION QHIST(1000),QITIME(1000)

      QI=(QHIST(NDX+1)-QHIST(NDX))/(QITIME(NDX+1)-QITIME(NDX))*
     +      (TIME-QITIME(NDX))+QHIST(NDX)


      RETURN
      END



******************************************************************************
******************************************************************************
*      A SUBROUTINE TO LINEARLY INTERPOLATE BETWEEN POINTS IN DATA SETS      *
*                     CONTAINING CHINA LAKE PARAMETERS                       *
******************************************************************************
******************************************************************************

      SUBROUTINE CINTERP(TIME,NDX,TCTEMP,TCEVAP,TCPRECIP,
     +      TCDELP,TCDELA)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      LOGICAL GUESS,CPARAM,GUESS2

      PARAMETER(NUMB=2000)
      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S


      TCTEMP=(CTEMP(NDX+1)-CTEMP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +      (TIME-WTIME(NDX))+CTEMP(NDX)

      TCEVAP=(CEVAP(NDX+1)-CEVAP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +      (TIME-WTIME(NDX))+CEVAP(NDX)
```

```
      TCPRECIP=(CPRECIP(NDX+1)-CPRECIP(NDX))/(WTIME(NDX+1)-
     +     WTIME(NDX))*(TIME-WTIME(NDX))+CPRECIP(NDX)

      TCDELP=(CDELP(NDX+1)-CDELP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +     (TIME-WTIME(NDX))+CDELP(NDX)

      TCDELA=(CSDELA(NDX+1)-CSDELA(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +     (TIME-WTIME(NDX))+CSDELA(NDX)

      RETURN
      END


*******************************************************************************
*******************************************************************************
*     A SUBROUTINE TO LINEARLY INTERPOLATE BETWEEN POINTS IN DATA SETS        *
*               FROM THE OWENS LAKE CALCULATIONS                              *
*******************************************************************************
*******************************************************************************

      SUBROUTINE C2INTERP(TIME,NDX,TCDELI,CQI,TAREA)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      LOGICAL GUESS,CPARAM,GUESS2

      PARAMETER(NUMA=25000,NUMB=2000)

      COMMON/BOTH/CL(NUMA),TSOO(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S


      TCDELI = (ODEL_OUT(NDX+1)-ODEL_OUT(NDX))/(OTIME(NDX+1)-
     +     OTIME(NDX))*(TIME-OTIME(NDX))+ODEL_OUT(NDX)

      CQI = (OQO(NDX+1)-OQO(NDX))/(OTIME(NDX+1)-OTIME(NDX))*
     +     (TIME-OTIME(NDX))+OQO(NDX)

      TAREA = (AREA(NDX+1)-AREA(NDX))/(OTIME(NDX+1)-OTIME(NDX))*
     +     (TIME-OTIME(NDX))+AREA(NDX)

      RETURN
      END


*******************************************************************************
*******************************************************************************
*     A SUBROUTINE TO LINEARLY INTERPOLATE BETWEEN POINTS IN AN ARRAY         *
```

```
*                FROM THE OWENS LAKE SALT OUTFLOW HISTORY                    *
******************************************************************************
******************************************************************************

      SUBROUTINE C3INTERP(TIME,NDX,SLTNUM)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      LOGICAL GUESS,CPARAM,GUESS2

      PARAMETER(NUMA=25000,NUMB=2000)

      COMMON/BOTH/CL(NUMA),TSOD(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S


      SLTNUM = (OCL_OUT(NDX+1)-OCL_OUT(NDX))/(OTIME(NDX+1)-
     +      OTIME(NDX))*(TIME-OTIME(NDX))+OCL_OUT(NDX)

      RETURN
      END


******************************************************************************
******************************************************************************
*     A SUBROUTINE TO LINEARLY INTERPOLATE BETWEEN POINTS IN DATA SETS       *
*                  CONTAINING SEARLES LAKE PARAMETERS                        *
******************************************************************************
******************************************************************************

      SUBROUTINE SINTERP(TIME,NDX,TSTEMP,TSEVAP,TSPRECIP,
     +      TSDELP,TSDELA)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      LOGICAL GUESS,CPARAM,GUESS2

      PARAMETER(NUMB=2000)
      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S
```

```
      TSTEMP=(STEMP(NDX+1)-STEMP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +      (TIME-WTIME(NDX))+STEMP(NDX)

      TSEVAP=(SEVAP(NDX+1)-SEVAP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +      (TIME-WTIME(NDX))+SEVAP(NDX)

      TSPRECIP=(SPRECIP(NDX+1)-SPRECIP(NDX))/(WTIME(NDX+1)-
     +      WTIME(NDX))*(TIME-WTIME(NDX))+SPRECIP(NDX)

      TSDELP=(SDELP(NDX+1)-SDELP(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +      (TIME-WTIME(NDX))+SDELP(NDX)

      TSDELA=(CSDELA(NDX+1)-CSDELA(NDX))/(WTIME(NDX+1)-WTIME(NDX))*
     +      (TIME-WTIME(NDX))+CSDELA(NDX)

      RETURN
      END


*******************************************************************************
*******************************************************************************
*     A SUBROUTINE TO LINEARLY INTERPOLATE BETWEEN POINTS IN AN ARRAY        *
*                FROM THE OWENS LAKE SALT OUTFLOW HISTORY                     *
*******************************************************************************
*******************************************************************************

      SUBROUTINE PDINTERP(TIME,NDX,TPEVAP,TDEVAP)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      LOGICAL GUESS,CPARAM,GUESS2

      PARAMETER(NUMA=25000,NUMB=2000)

      COMMON/BOTH/CL(NUMA),TSOD(10),TCL(10),OTIME(NUMA),TCO3(10),
     + OQO(NUMA),QO(10),CONC_CL(NUMA),DOLTEMP,DELDOL,TODELI,TOTEMP,
     + ODEL_OUT(NUMA),OCL_OUT(NUMA),PEVAP(NUMB),DEVAP(NUMB),
     + SUM_PCL_DEP,AREA_P,AREA_D,
     + AREA(NUMA),SOAREA,ALLAREA,CONC_CL_P,
     + CL_P

      COMMON/FIRST/WTIME(NUMB),OTEMP(NUMB),OEVAP(NUMB),
     + OHUM(500),OPRECIP(NUMB),ODELP(NUMB),ODELA(NUMB),ODELI(NUMB),
     + GUESS,TEMPC,EVAPC,PRECIPC,DELAC,DELIC,CPARAM,DELPC,
     + EVAPC_P,EVAPC_D

      COMMON/SECOND/CTEMP(NUMB),CEVAP(NUMB),CHUM(500),CPRECIP(NUMB),
     + CDELP(NUMB),CSDELA(NUMB),GUESS2,TEMPC_C,EVAPC_C,
     + PRECIPC_C,DELAC_C,DELPC_C,STEMP(NUMB),SEVAP(NUMB),SHUM(500),
     + SPRECIP(NUMB),SDELP(NUMB),TEMPC_S,EVAPC_S,
     + PRECIPC_S,DELAC_S,DELPC_S


      TPEVAP = (PEVAP(NDX+1)-PEVAP(NDX))/(WTIME(NDX+1)-
     +      WTIME(NDX))*(TIME-WTIME(NDX))+PEVAP(NDX)

      TDEVAP = (DEVAP(NDX+1)-DEVAP(NDX))/(WTIME(NDX+1)-
     +      WTIME(NDX))*(TIME-WTIME(NDX))+DEVAP(NDX)

      RETURN
      END


*******************************************************************************
```

```
******************************************************************************
**  A FUNCTION SUBPROGRAM TO CALCULATE THE RELATIVE ISOTOPIC ENRICHMENT OF   **
**  THE EVAPORATING WATER.                                                   **
******************************************************************************
******************************************************************************

      DOUBLE PRECISION FUNCTION DELE(DELL,EPS,HUM)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      PARAMETER(C=6.8D-3)

      DELE=(((1.D0+1.D-3*DELL)*(1.D0-C))/((1.D0+1.D-3*EPS)*(1.D0-C*
     +     HUM))-1.D0)*1.0D3

      RETURN
      END



******************************************************************************
******************************************************************************
**          A FUNCTION SUBPROGRAM TO CALCULATE THE INFLOW, QI(T)            **
******************************************************************************
******************************************************************************

      DOUBLE PRECISION FUNCTION FQI(X)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)

      COMMON/INFLOW/INCHOICE,A,B,C

      IF(INCHOICE .EQ. 1)THEN
         FQI=A*X+B
      ELSE IF(INCHOICE .EQ. 2)THEN
         FQI=B*DEXP(A*X)
      ELSE IF(INCHOICE .EQ. 3)THEN
         IF(X .LT. 1.D0)X=1.D0
         FQI=B+A*DLOG10(X)
      ELSE IF(INCHOICE .EQ. 4)THEN
         FQI=B+A*X**C
      ELSE IF(INCHOICE .EQ. 5)THEN
         FQI=B+A*DSIN(C*X)
      ELSE IF(INCHOICE .EQ. 6)THEN
         IF(X .LT. 1.D0)THEN
            FQI=B
         ELSE
            FQI=B+A*B
         ENDIF
      ELSE IF(INCHOICE .EQ. 7)THEN
         FQI=0.D0
      ENDIF
      IF(FQI .LT. 0.D0)FQI=0.D0
      RETURN
      END



******************************************************************************
******************************************************************************
*          A SUBROUTINE TO FIND THE TIME INDEX WITH A BINARY SEARCH          *
******************************************************************************
******************************************************************************

      SUBROUTINE FINDT(NPTS,TM,INDEX,FOUND,TIME)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND
      PARAMETER(NUMB=2000)
      DIMENSION TIME(NUMB)
```

```
      COMMON/SEARCH/IFIRST,ILAST,HIFIRST,HILAST

      TFST=IFIRST
      TLST=ILAST
      FOUND = .FALSE.
      DO 200 I = 1,100
         IF(TLST .LT. TFST) THEN
            INDEX=TLST
            GOTO 210
         ENDIF
         IF( TFST .LE. TLST .AND. .NOT. FOUND)THEN
            MIDDLE = (TFST+TLST)/2
            TEST = ABS(TM-TIME(MIDDLE))
            IF( TEST .LT. 1.0D-1) THEN
               FOUND = .TRUE.
               INDEX=MIDDLE
               GOTO 210
            ELSE IF(TM .LT. TIME(MIDDLE))THEN
               TLST = MIDDLE-1
            ELSE
               TFST = MIDDLE+1
            END IF
         END IF
200   CONTINUE

210   RETURN
      END



*********************************************************************************
*********************************************************************************
*     A SUBROUTINE TO FIND THE HUMIDITY TIME INDEX WITH A BINARY SEARCH        *
*********************************************************************************
*********************************************************************************

      SUBROUTINE FINDHT(NPTS,TM,INDEX,TIME)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND
      DIMENSION TIME(500)
      IFIRST=1
      ILAST=NPTS
      FOUND = .FALSE.
      DO 200 I = 1,100
         IF(ILAST .LT. IFIRST) THEN
            INDEX=ILAST
            GOTO 210
         ENDIF
         IF( IFIRST .LE. ILAST .AND. .NOT. FOUND)THEN
            MIDDLE = (IFIRST+ILAST)/2
            TEST = ABS(TM-TIME(MIDDLE))
            IF( TEST .LT. 1.0D-1) THEN
               FOUND = .TRUE.
               INDEX=MIDDLE
               GOTO 210
            ELSE IF(TM .LT. TIME(MIDDLE))THEN
               ILAST = MIDDLE-1
            ELSE
               IFIRST = MIDDLE+1
            END IF
         END IF
200   CONTINUE

210   RETURN
      END
```

```
*******************************************************************************
*******************************************************************************
*     A SUBROUTINE TO FIND THE INFLOW TIME INDEX WITH A BINARY SEARCH         *
*******************************************************************************
*******************************************************************************

      SUBROUTINE FINDQT(NPTS,TM,INDEX,TIME)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND
      DIMENSION TIME(1000)
      IFIRST=1
      ILAST=NPTS
      FOUND = .FALSE.
      DO 200 I = 1,100
         IF(ILAST .LT. IFIRST) THEN
            INDEX=ILAST
            GOTO 210
         ENDIF
         IF( IFIRST .LE. ILAST .AND. .NOT. FOUND)THEN
            MIDDLE = (IFIRST+ILAST)/2
            TEST = ABS(TM-TIME(MIDDLE))
            IF( TEST .LT. 1.0D-1) THEN
               FOUND = .TRUE.
               INDEX=MIDDLE
               GOTO 210
            ELSE IF(TM .LT. TIME(MIDDLE))THEN
               ILAST = MIDDLE-1
            ELSE
               IFIRST = MIDDLE+1
            END IF
         END IF
200   CONTINUE

210   RETURN
      END


*******************************************************************************
*******************************************************************************
*          A SUBROUTINE TO FIND THE TIME INDEX WITH A BINARY SEARCH           *
*******************************************************************************
*******************************************************************************

      SUBROUTINE FINDT2(NPTS,TM,INDEX2,FOUND2,TIME)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND2
      PARAMETER(NUMA=25000)
      DIMENSION TIME(NUMA)
      IFIRST=1
      ILAST=NPTS
      FOUND2 = .FALSE.
      DO 200 I = 1,100
         IF(ILAST .LT. IFIRST) THEN
            INDEX2=ILAST
            GOTO 210
         ENDIF
         IF( IFIRST .LE. ILAST .AND. .NOT. FOUND2) THEN
            MIDDLE = (IFIRST+ILAST)/2
            TEST = ABS(TM-TIME(MIDDLE))
            IF( TEST .LT. 1.0D-1) THEN
               FOUND2 = .TRUE.
               INDEX2=MIDDLE
               GOTO 210
```

```
            ELSE IF(TM .GT. TIME(MIDDLE))THEN
               ILAST = MIDDLE-1
            ELSE
               IFIRST = MIDDLE+1
            END IF
         END IF
200   CONTINUE

210   RETURN
      END


************************************************************
************************************************************
**    A FUNCTION SUBPROGRAM TO  CALCULATE DEL DOLOMITE **
************************************************************
************************************************************

      DOUBLE PRECISION FUNCTION FDDOL(TEMP,DELH2O)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

*************************************
** CONVERT TEMP TO DEGREES KELVIN **
*************************************

      TEMPK = TEMP+273.15D0

**********************************************
** CALCULATE EPSILON FOR DOLOMITE AND WATER **
**********************************************

      EPSDOL=(DEXP((3.2D0*(1.D6/TEMPK**2)-4.3D0)/1.D3)-1.D0)*1.D3

      FDDOL=EPSDOL+DELH2O*(EPSDOL/1.0D3+1.D0)

      RETURN
      END


************************************************************
************************************************************
**    A FUNCTION SUBPROGRAM TO  CALCULATE DEL WATER    **
************************************************************
************************************************************

      DOUBLE PRECISION FUNCTION W_DEL(DELDOL,TEMP)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

*************************************
** CONVERT TEMP TO DEGREES KELVIN **
*************************************

      TEMPK = TEMP+273.15D0

**********************************************
** CALCULATE EPSILON FOR DOLOMITE AND WATER **
**********************************************

      EPSDOL=(DEXP((3.2D0*(1.D6/TEMPK**2)-4.3D0)/1.D3)-1.D0)*1.D3

      W_DEL=(DELDOL-EPSDOL)/((EPSDOL/1.D3)+1.D0)

      RETURN
      END
```

Appendix N

Transient Isotopic Model for Lake San Agustin

```
*****************************************************************************
*****************************************************************************
* THIS PROGRAM CALCULATES CHANGES IN LAKE VOLUME AND ISOTOPIC COMPOSITION   *
* WITH RESPECT TO TIME.                                                     *
*****************************************************************************
*****************************************************************************

*****************************************************************************
*       A PROGRAM TO CALL THE RUNGE-KUTTA-FEHLBERG ORDER 4 ROUTINE TO SOLVE *
*       A SYSTEM OF PARTIAL DIFFERENTIAL EQUATION OF THE FORM:  F(T,X)= X'  *
*****************************************************************************
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DIMENSION X(3),TOL(3)
        LOGICAL FOUND,GUESS,PASS,GRAF,CPARAM,ONLY1,ZEROVOL,ZEROCHK

        COMMON/FIRST/INCHOICE,A,B,STIME(300),TEMP(300),EVAP(300),
     +  HUM(300),DELA(300),DELI(300),GUESS,C,TEMPC,HUMC,EVAPC,DELAC,
     +  DELIC,CPARAM,TEMPCURV,NPTS,HTIME(300),NHPTS

        COMMON/TRADE1/GRAF,CALTEMP,DELCAL,TDELI,TTEMP
        OPEN(UNIT=98,FILE='SANAG.INP',STATUS='OLD')
        WRITE(6,*)
        WRITE(6,*)'READING ITMAX,N,X(I),DTMAX,DTMIN,TOL(I)'
        READ(98,*)ITMAX,N,(X(I),I=1,N),DTMAX,DTMIN,(TOL(I),I=1,N)

*****************************************************************
** THIS VARIABLE IS USED TO "GUESS" AT AN INITIAL INFLOW QI **
*****************************************************************
        GUESS=.TRUE.

*******************************************
** A CHANCE TO ADJUST INPUT PARAMETERS **
*******************************************

        WRITE(6,*)
        WRITE(6,*)'THE CURRENT PARAMETERS ARE:'
        WRITE(6,*)'   1 : MAX ITERATIONS =',ITMAX
        WRITE(6,*)'   2 : LAKE VOL =',X(1)
        WRITE(6,*)'   3 : DEL O-18 =',X(2)
        WRITE(6,*)'   4 : MAX TIME STEP =',DTMAX
        WRITE(6,*)'   5 : MIN TIME STEP =',DTMIN
        WRITE(6,*)'   6 : LAKE VOL TOLERANCE=',TOL(1)
        WRITE(6,*)'   7 : O-18 TOLERANCE=',TOL(2)

        WRITE(6,*)
        WRITE(6,*)
        WRITE(6,*)'DO YOU WANT TO CHANGE ANY OF THE STARTING PARAMETERS ?'
        WRITE(6,*)'1=YES    0=NO'
        READ(5,*)ISEE
        WRITE(6,*)
        WRITE(6,*)

        IF(ISEE .EQ. 1)THEN

          WRITE(6,*)
          WRITE(6,*)'HOW MANY PARAMETERS WOULD YOU LIKE TO CHANGE ?'
          READ(5,*)K
          WRITE(6,*)

          DO 250 I = 1,K
             WRITE(6,*)
300          WRITE(6,*)'ENTER THE PARAMETER NUMBER'
             WRITE(6,*)
             WRITE(6,*)'   1 : MAX ITERATIONS ='
             WRITE(6,*)'   2 : LAKE VOL ='
             WRITE(6,*)'   3 : DEL O-18 ='
             WRITE(6,*)'   4 : MAX TIME STEP ='
             WRITE(6,*)'   5 : MIN TIME STEP ='
             WRITE(6,*)'   6 : LAKE VOL TOLERANCE='
             WRITE(6,*)'   7 : O-18 TOLERANCE='
             READ(5,*)NUMP
             IF(NUMP .EQ. 1)THEN
                WRITE(6,*)
                WRITE(6,*)'MAX ITERATIONS :',ITMAX
                WRITE(6,*)'ENTER NEW VALUE'
```

```
                 READ(5,*)ITMAX
              ELSE IF(NUMP .EQ. 2)THEN
                 WRITE(6,*)
                 WRITE(6,'(A,D15.7)')' LAKE VOLUME :',X(1)
                 WRITE(6,*)'NOTE: MAX LAKE VOL IS 39.89809 (M^3)'
                 WRITE(6,*)'ENTER NEW VALUE'
                 READ(5,*)X(1)
              ELSE IF(NUMP .EQ. 3)THEN
                 WRITE(6,*)
                 WRITE(6,*)'DEL O-18 :',X(2)
                 WRITE(6,*)'ENTER NEW VALUE'
                 READ(5,*)X(2)
              ELSE IF(NUMP .EQ. 4)THEN
                 WRITE(6,*)
                 WRITE(6,*)'MAX TIME STEP :',DTMAX
                 WRITE(6,*)'ENTER NEW VALUE'
                 READ(5,*)DTMAX
              ELSE IF(NUMP .EQ. 5)THEN
                 WRITE(6,*)
                 WRITE(6,*)'MIN TIME STEP :',DTMIN
                 WRITE(6,*)'ENTER NEW VALUE'
                 READ(5,*)DTMIN
              ELSE IF(NUMP .EQ. 6)THEN
                 WRITE(6,*)
                 WRITE(6,*)'LAKE VOL TOLERANCE :',TOL(1)
                 WRITE(6,*)'ENTER NEW VALUE'
                 READ(5,*)TOL(1)
              ELSE IF(NUMP .EQ. 7)THEN
                 WRITE(6,*)
                 WRITE(6,*)'DEL O-18 TOLERANCE :',TOL(2)
                 WRITE(6,*)'ENTER NEW VALUE'
                 READ(5,*)TOL(2)
              ELSE
                 WRITE(6,*)
                 WRITE(6,*)'GET A CLUE BLISTER BRAIN'
                 WRITE(6,*)'PICK A NUMBER BETWEEN 1 AND 8'
                 WRITE(6,*)
                 GO TO 300
              ENDIF
250        CONTINUE

************************************
** WRITE NEW VALUES TO INPUT FILE **
************************************

           REWIND 98
           WRITE(98,*)ITMAX,N,(X(I),I=1,N),DTMAX,DTMIN,(TOL(I),
     +              I=1,N)

        ENDIF

******************************************************************************
*             LET'S CHOOSE AN INFLOW FUNCTION AND TIME INTERVAL              *
******************************************************************************


        WRITE(6,*)
        WRITE(6,*)
115     WRITE(6,*)'WHICH INFLOW FUNCTION WOULD YOU LIKE?'
        WRITE(6,*)'1=LINEAR'
        WRITE(6,*)'2=EXPONENTIAL'
        WRITE(6,*)'3=LOGARITHMIC'
        WRITE(6,*)'4=POWER'
        WRITE(6,*)'5=SINUSOIDAL'
        WRITE(6,*)'6=STEP'
        WRITE(6,*)'7=ZERO INFLOW'
        READ(5,*)INCHOICE

        WRITE(6,*)
        WRITE(6,*)
        IF(INCHOICE .EQ. 1)THEN
           WRITE(6,*)'YOU HAVE CHOSEN f(QI)= A*X+B'
           WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
        ELSE IF(INCHOICE .EQ. 2)THEN
           WRITE(6,*)'YOU HAVE CHOSEN f(QI)= B*EXP(A*X)'
           WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
```

```
         ELSE IF(INCHOICE .EQ. 3)THEN
            WRITE(6,*)'YOU HAVE CHOSEN f(QI)= B+A*log(X)'
            WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
         ELSE IF(INCHOICE .EQ. 4)THEN
            WRITE(6,*)'YOU HAVE CHOSEN f(QI)= B+A*(X**C)'
            WRITE(6,*)'ENTER VALUES FOR ''A'',''B'', AND ''C'''
         ELSE IF(INCHOICE .EQ. 5)THEN
            WRITE(6,*)'YOU HAVE CHOSEN f(QI)= B+A*SIN(C*X)'
            WRITE(6,*)'ENTER VALUES FOR ''A'',''B'', AND ''C'''
         ELSE IF(INCHOICE .EQ. 6)THEN
            WRITE(6,*)'YOU HAVE CHOSEN f(QI)= B+(A*B)'
            WRITE(6,*)'ENTER A VALUE FOR ''A'', AND ''B'''
         ELSE IF(INCHOICE .EQ. 7)THEN
            WRITE(6,*)'YOU HAVE CHOSEN ZERO INFLOW'
            WRITE(6,*)'GRAB YOUR CANTEEN AND HEAD FOR THE SHADE'
         ELSE
            WRITE(6,*)
            WRITE(6,*)
            WRITE(6,*)'NOT A VALID CHOICE MULLET-HEAD'
            WRITE(6,*)
            WRITE(6,*)
            GOTO 115
         ENDIF

         IF(INCHOICE .EQ. 4 .OR. INCHOICE .EQ. 5)THEN
            READ(5,*)A,B,C
         ELSE IF(INCHOICE .NE. 7)THEN
            READ(5,*)A,B
         ENDIF

         WRITE(6,*)
         WRITE(6,*)'ENTER STARTING TIME AND ENDING TIME'
         WRITE(6,*)'0 CORRESPONDS TO PRESENT, 2.0E6 IS 2 MILLION YRS AGO'
         WRITE(6,*)
         WRITE(6,*)
         WRITE(6,*)'MANAGEMENT ACCEPTS NO RESPONSIBILITY FOR PEOPLE WHO'
         WRITE(6,*)'RUN THE MODEL BACKWARD IN TIME'
         WRITE(6,*)
         WRITE(6,*)
         READ(5,*)TBEG, TEND
         WRITE(6,*)

**********************************
** CONSTANT PARAMETER OPTION **
**********************************

         CPARAM=.FALSE.
         WRITE(6,*)
         WRITE(6,*)'DO YOU WANT TO RUN THE PROGRAM WITH CONSTANT PARAMETE
     +RS ?'
         WRITE(6,*)'1=YES    0=NO'
         WRITE(6,*)
         READ(5,*)INPARAM

         IF(INPARAM .EQ. 1)THEN
            CPARAM=.TRUE.
            OPEN(UNIT=68, FILE='CONSTANT.INP', STATUS='OLD')
            READ(68,*) TEMPC,HUMC,EVAPC,DELAC,DELIC
            CLOSE(UNIT=68)

         END IF

**********************
** GRAPHICS STUFF **
**********************

         GRAF=.FALSE.

         WRITE(6,*)
         WRITE(6,*)'DO YOU WANT TO PLOT THIS RUN ON THE SCREEN'
         WRITE(6,*)'1=YES    0=NO'
         READ(5,*)IGRAF
         WRITE(6,*)

         IF(IGRAF .EQ. 1)THEN
            GRAF=.TRUE.
```

```
        ENDIF

*****************************************************************************
*    All we're doing here is reading data files, the good stuff is later   *
*****************************************************************************

        IF(.NOT. CPARAM)THEN

          WRITE(6,*)
          WRITE(6,*)
          WRITE(6,*)'READING DATA FILES, GOOD TIME TO GET MUNCHIES'
          WRITE(6,*)
90        WRITE(6,*)'WHICH TEMPERATURE CURVE DO YOU WANT?'
          WRITE(6,*)'1=SHACKLETON OCEAN O-18 CURVE'
          WRITE(6,*)'2=PHILLIPS SAN JUAN BASIN CURVE'
          WRITE(6,*)'3=OSTRACODE STRATIGRAPHY CURVE'
          READ(5,*)TEMPCURV

          IF(TEMPCURV .EQ. 1)THEN
            WRITE(6,*)'YOU HAVE CHOSEN THE SHACKLETON OCEAN CURVE'
            OPEN(UNIT=20,FILE='SHACK_SA.CLIM',STATUS='OLD')
            OPEN(UNIT=25,FILE='SHACK_SA.ISODAT',STATUS='OLD')
            OPEN(UNIT=30,FILE='HUM_SA.DAT',STATUS='OLD')

            DO 101 I=1,300
              READ(20,*,END=102)STIME(I),TEMP(I),EVAP(I)
101         CONTINUE

102         NPTS = I-1

            DO 103 I=1,NPTS
              READ(25,*)DELI(I),DELA(I)
103         CONTINUE

            DO 104 I=1,300
              READ(30,*,END=105) HTIME(I), HUM(I)
104         CONTINUE

105         NHPTS = I-1

            DO 106 I=1,NPTS
              STIME(I)=STIME(I)*1.0D3
106         CONTINUE

            DO 107 I=1,NHPTS
              HTIME(I)=HTIME(I)*1.0D3
107         CONTINUE

            CLOSE(UNIT=20)
            CLOSE(UNIT=25)
            CLOSE(UNIT=30)

          ELSE IF(TEMPCURV .EQ. 2)THEN
            WRITE(6,*)'YOU HAVE CHOSEN UNCLE FREDDYS SAN JUAN CURVE'
            OPEN(UNIT=35,FILE='SANJUAN_SA.CLIM',STATUS='OLD')
            OPEN(UNIT=40,FILE='SANJUAN_SA.ISODAT',STATUS='OLD')
            OPEN(UNIT=45,FILE='HUM_SA.DAT',STATUS='OLD')

            DO 120 I=1,300
              READ(35,*,END=121)STIME(I),TEMP(I),EVAP(I)
120         CONTINUE

121         NPTS = I-1

            DO 122 I=1,NPTS
              READ(40,*)DELI(I),DELA(I)
122         CONTINUE

            DO 123 I=1,300
              READ(45,*,END=124)HTIME(I),HUM(I)
123         CONTINUE

124         NHPTS = I-1

            DO 125 I=1,NPTS
```

```
                 STIME(I)=STIME(I)*1.0D3
 125          CONTINUE

              DO 126 I=1,NHPTS
                 HTIME(I)=HTIME(I)*1.0D3
 126          CONTINUE

              CLOSE(UNIT=35)
              CLOSE(UNIT=40)
              CLOSE(UNIT=45)

           ELSE IF(TEMPCURV .EQ. 3)THEN
              WRITE(6,*)'YOU HAVE CHOSEN THE OSTRACODE CURVE'
              OPEN(UNIT=50,FILE='COD_CLIM_ISO.DAT',STATUS='OLD')
              OPEN(UNIT=55,FILE='HUM_SA.DAT',STATUS='OLD')

              DO 130 I=1,300
                 READ(50,*,END=131)STIME(I),TEMP(I),EVAP(I),DELI(I),
     +           DELA(I)
 130          CONTINUE

 131          NPTS = I-1

              DO 132 I=1,300
                 READ(55,*,END=133)HTIME(I),HUM(I)
 132          CONTINUE

 133          NHPTS = I-1

              DO 134 I=1,NPTS
                 STIME(I)=STIME(I)*1.0D3
 134          CONTINUE

              DO 135 I=1,NHPTS
                 HTIME(I)=HTIME(I)*1.0D3
 135          CONTINUE

              CLOSE(UNIT=50)
              CLOSE(UNIT=55)

           ELSE
              WRITE(6,*)'NOT A VALID CHOICE KNUMB-KNUCKLES, TRY AGAIN!'
           GO TO 90

           ENDIF
        ENDIF

*****************************
** START THE BALL ROLLING **
*****************************

        CALL RKF(N,X,TBEG,TEND,TOL,DTMAX,DTMIN,ITMAX)

****************************
** FINISH PLOTTING STUFF **
****************************

        IF(GRAF)THEN
           CALL ENDPL(0)
           CALL DONEPL
        ENDIF

        WRITE(99,*)
        WRITE(96,*)
        WRITE(95,*)
        END

        SUBROUTINE RKF(N,X,TBEG,TEND,TOL,DTMAX,DTMIN,ITMAX)
*******************************************************************************
*       SOLVE A SYSTEM OF PARTIAL DIFFERENTIAL EQUATION OF THE FORM:          *
*                            F(T,X)= X'                                       *
*        BETWEEN T1,T2, GIVEN THE INITIAL CONDITION XO(T1)                    *
*******************************************************************************
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        COMMON/TRADE1/GRAF,CALTEMP,DELCAL,TDELI,TTEMP
        REAL XMIN,XMAX,YMIN,YMAX,YDEL(300),XPLT(300),YQI(300),
```

```
     +   YAREA(300),YTEMP(300)

         DIMENSION X(3),RK1(3),RK2(3),RK3(3),RK4(3),RK5(3),RK6(3),R(3)
         DIMENSION TERM(3),DEL(3),TOL(3)
         PARAMETER (VOLMAX=39.89809)
         LOGICAL FOUND,GUESS,PASS,GRAF,CPARAM,ONLY1,ZEROVOL,ZEROCHK
         OPEN(UNIT=99,FILE='DEL.OUT',STATUS='NEW',CARRIAGE CONTROL=
     +       'LIST')
         OPEN(UNIT=97,FILE='DIAG.OUT',STATUS='NEW',CARRIAGE CONTROL=
     +       'LIST')
         OPEN(UNIT=96,FILE='AREA.OUT',STATUS='NEW',CARRIAGE CONTROL=
     +       'LIST')
         OPEN(UNIT=95,FILE='QI.OUT',STATUS='NEW',CARRIAGE CONTROL=
     +       'LIST')
 10      FORMAT(A,D12.5)


         XMIN=TBEG
         XMAX=TEND
         TIME=TBEG
         STEP=DTMAX
         KOUNT=1
         ONLY1=.TRUE.
         ZEROVOL=.FALSE.
         ZEROCHK=.FALSE.

*************************************
** THE SOLVING ROUTINE BEGINS HERE **
*************************************

         WRITE(6,*)
         WRITE(6,*)
         WRITE(6,*)'START SOLVING DIFFERENTIAL EQUATIONS'
         WRITE(6,*)
         WRITE(6,*)

         DO 100 ITER=1,ITMAX
            IF(TIME.GT.TEND)THEN
               KNT=1
               T=TIME
               DO 200 I=1,N
                  RK1(I)=STEP*F(I,TIME,X,T,KNT,KOUNT,TBEG)
 200           CONTINUE

********************************************
** STORE INITIAL VALUES IN GRAPHICS ARRAY **
********************************************

               IF(ONLY1)THEN
                  YDEL(1)=DELCAL
                  XPLT(1)=TBEG
                  YAREA(1)=SAREA(X(1))
                  YQI(1)=FQI(0.D0)
                  YTEMP(1)=TTEMP
                  ONLY1=.FALSE.
               ENDIF

*************************************************************************
** "TERM" IS AN ARRAY WHICH STORES APPROXIMATIONS OF VOL AND DEL O-18 **
** WHICH WILL BE USED IN FINAL CALCULATIONS IF ERRORS WITHIN THE STEP **
** ARE LESS THAN THE GIVEN TOLERANCES.                                **
*************************************************************************

               T=TIME-STEP/4.D0
               KNT=2
               DO 300 I=1,N
                  TERM(I)=X(I)+ RK1(I)/4.D0
 300           CONTINUE
               IF(TERM(1) .LE. 0.D0)THEN
                  IF(STEP .EQ. DTMIN)THEN
                     ZEROVOL=.TRUE.
                     PASS=.TRUE.
                     GOTO 1375
                  ELSE
                     ZEROCHK=.TRUE.
                  ENDIF
```

```
            ENDIF
            DO 400 I=1,N
               RK2(I)=STEP*F(I,TIME,TERM,T,KNT,KOUNT,TBEG)
  400       CONTINUE
            T=TIME-3.D0*STEP/8.D0
            KNT=3
            DO 500 I=1,N
               TERM(I)=X(I)+(3.D0*RK1(I)+9.D0*RK2(I))/32.D0
  500       CONTINUE
            IF(TERM(1) .LE. 0.D0)THEN
               IF(STEP .EQ. DTMIN)THEN
                  ZEROVOL=.TRUE.
                  PASS=.TRUE.
                  GOTO 1375
               ELSE
                  ZEROCHK=.TRUE.
               ENDIF
            ENDIF
            DO 600 I=1,N
               RK3(I)=STEP*F(I,TIME,TERM,T,KNT,KOUNT,TBEG)
  600       CONTINUE
            T=TIME-12.D0*STEP/13.D0
            KNT=4
            DO 700 I=1,N
               TERM(I)=X(I) + (1932.D0*RK1(I)-7200.D0*RK2(I)+
     +                   7296.D0*RK3(I))/2197.D0
  700       CONTINUE
            IF(TERM(1) .LE. 0.D0)THEN
               IF(STEP .EQ. DTMIN)THEN
                  ZEROVOL=.TRUE.
                  PASS=.TRUE.
                  GOTO 1375
               ELSE
                  ZEROCHK=.TRUE.
               ENDIF
            ENDIF
            DO 800 I=1,N
               RK4(I)=STEP*F(I,TIME,TERM,T,KNT,KOUNT,TBEG)
  800       CONTINUE
            T=TIME-STEP
            KNT=5
            DO 900 I=1,N
               TERM(I)=X(I) +439.D0*RK1(I)/216.D0-
     +                8.D0*RK2(I)+3680.D0*RK3(I)/513.D0-
     +                845.D0*RK4(I)/4104.D0
  900       CONTINUE
            IF(TERM(1) .LE. 0.D0)THEN
               IF(STEP .EQ. DTMIN)THEN
                  ZEROVOL=.TRUE.
                  PASS=.TRUE.
                  GOTO 1375
               ELSE
                  ZEROCHK=.TRUE.
               ENDIF
            ENDIF
            DO 1000 I=1,N
               RK5(I)=STEP*F(I,TIME,TERM,T,KNT,KOUNT,TBEG)
 1000       CONTINUE
            T=TIME-STEP/2.D0
            KNT=6
            DO 1100 I=1,N
               TERM(I)=X(I)-8.D0*RK1(I)/27.D0+
     +                2.D0*RK2(I)-3544.D0*RK3(I)/2565.D0+
     +                1859.D0*RK4(I)/4104.D0-11.D0*RK5(I)/40.D0
 1100       CONTINUE
            IF(TERM(1) .LE. 0.D0)THEN
               IF(STEP .EQ. DTMIN)THEN
                  ZEROVOL=.TRUE.
                  PASS=.TRUE.
                  GOTO 1375
               ELSE
                  ZEROCHK=.TRUE.
               ENDIF
            ENDIF
            DO 1200 I=1,N
               RK6(I)=STEP*F(I,TIME,TERM,T,KNT,KOUNT,TBEG)
```

```
1200      CONTINUE
          IF(TERM(1) .LE. 0.D0)THEN
             IF(STEP .EQ. DTMIN)THEN
                ZEROVOL=.TRUE.
                PASS=.TRUE.
                GOTO 1375
             ELSE
                ZEROCHK=.TRUE.
             ENDIF
          ENDIF
          PASS=.TRUE.

**********************************************
** CALCULATE ERRORS RESULTING FROM STEP SIZE **
**********************************************

          DO 1300 I=1,N
             R(I)=DABS(RK1(I)/360.D0 -128.D0*RK3(I)/4275.D0-
     +             2197.D0*RK4(I)/75240.D0+RK5(I)/50.D0+
     +             2.D0*RK6(I)/55.D0)/STEP
             IF(R(I).GT.TOL(I)) PASS=.FALSE.
D            WRITE(97,*)'R(I)',R(I)
D            WRITE(97,*)'TOL(I)',TOL(I)
1300      CONTINUE

D         WRITE(97,10) 'STEP=',STEP
D         WRITE(97,10)'RK1(1)',RK1(1)
D         WRITE(97,10)'RK2(1)',RK2(1)
D         WRITE(97,10)'RK3(1)',RK3(1)
D         WRITE(97,10)'RK4(1)',RK4(1)
D         WRITE(97,10)'RK5(1)',RK5(1)
D         WRITE(97,10)'RK6(1)',RK6(1)
D         WRITE(97,10)'RK1(2)',RK1(2)
D         WRITE(97,10)'RK2(2)',RK2(2)
D         WRITE(97,10)'RK3(2)',RK3(2)
D         WRITE(97,10)'RK4(2)',RK4(2)
D         WRITE(97,10)'RK5(2)',RK5(2)
D         WRITE(97,10)'RK6(2)',RK6(2)

****************************************************************************
** MAKE SURE THE SOLVER ISN''T "STUCK" BECAUSE OF THE ERROR TOLERANCES **
****************************************************************************

          IF(R(1) .EQ. RIPREV)THEN
             IF(ZEROCHK)THEN
                PASS=.TRUE.
                ZEROVOL=.TRUE.
                GOTO 1375
             ELSE
                WRITE(6,*)
                WRITE(6,*)'THE CURRENT RUN IS "STUCK" BUT WE HAVE
     +FORCED IT TO MOVE ON'
                WRITE(6,*)'DESPITE THE GIVEN TOLERANCES'
                WRITE(6,*)
                PASS=.TRUE.
                GOTO 1375
             ENDIF
          ELSE
             RIPREV=R(1)
          ENDIF

          DO 1310 I=1,N
             IF(R(I) .EQ. 0.D0)R(I)=.1
1310      CONTINUE
          DELMIN=4.D0

*****************************************************
** 'DEL' IS A VARIABLE USED TO UPDATE THE STEP SIZE **
*****************************************************

          DO 1350 I = 1,N
             DEL(I)=0.84*(TOL(I)/R(I))**(1.D0/4.D0)
D            WRITE(97,*)' DEL(I)',DEL(I)
             DELMIN=DMIN1(DEL(I),DELMIN)
1350      CONTINUE
```

```
*********************************************************
** IF THE ERROR IS LESS THAN THE GIVEN TOLERANCES ... **
*********************************************************

 1375         IF(PASS)THEN
                IF(TIME-STEP.GE.TEND)THEN
                  KOUNT=KOUNT+1
                  TIME=TIME-STEP
                  IF(ZEROVOL)THEN
                    X(1)=0.D0
                    X(2)=TDELI
                    ZEROVOL=.FALSE.
                    ZEROCHK=.FALSE.
                  ELSE

***********************************
** CALCULATE VOLUME AND DEL O-18 **
***********************************

                    DO 1400 I=1,N
                      X(I)=X(I)+25.D0*RK1(I)/216.D0+
     +                   1408.D0*RK3(I)/2565.D0+
     +                   2197.D0*RK4(I)/4104.D0- RK5(I)/5.D0
D                        WRITE(97,*)'X(I)',X(I)
1400                CONTINUE
                  ENDIF

**********************************************
** CALCULATE DEL CALCITE FROM DEL WATER, X(2)**
**********************************************

                  DELCAL=FDCAL(CALTEMP,X(2))
D                 WRITE(97,*)'DELCAL',DELCAL

*****************************************************
** STORE VALUES IN ARRAYS FOR THE PLOTTING ROUTINE **
*****************************************************

                  QI=FQI(TBEG-TIME)
                  IF(QI .LE. 0.D0)QI=0.D0

                  IF(GRAF)THEN
                    YQI(KOUNT)=QI
                    YTEMP(KOUNT)=CALTEMP
                    YDEL(KOUNT)=DELCAL
                    YAREA(KOUNT)=SAREA(X(1))
                    XPLT(KOUNT)=TIME
                  ENDIF

**************************
** WRITE RESULTS TO FILE **
**************************

D                 WRITE(97,*)
D                 WRITE(97,*)' TOTAL ELAPSED TIME',TBEG-TIME
D                 WRITE(97,*)

                  AREA=SAREA(X(1))
                  WRITE(96,*)TIME,AREA
                  WRITE(99,*)TIME,DELCAL
                  WRITE(95,*)TIME,QI

D                 WRITE(97,*)KOUNT,TIME,STEP,X(1),DELCAL
                  STEP=DTMAX
                  GOTO 100

**********************************
** MAKE SURE THE SOLVER DOESN'T **
** OVERSTEP DESIGNATED END TIME **
**********************************

                ELSEIF(TIME-STEP.LT.TEND)THEN
                  DTMAX=TIME-TEND
                  STEP=DTMAX
                  GOTO 100
                ENDIF
```

```
*****************************************
** ADJUST THE SIZE OF THE TIME STEP **
*****************************************

            ELSEIF(DELMIN .LE. 0.1)THEN
               STEP=STEP*1.0D-1
            ELSEIF(DELMIN .GE. 4.D0)THEN
               STEP=4.D0*STEP
            ELSE
               STEP=DELMIN*STEP
            ENDIF
            IF(STEP.GT.DTMAX)STEP=DTMAX
            IF(STEP.LT.DTMIN)STEP=DTMIN
         ELSE
            WRITE(6,*)
            WRITE(6,*)'FINAL AREA',AREA
            WRITE(6,*)'FINAL DEL CALCITE',DELCAL
            WRITE(6,*)'FINAL DEL WATER',X(2)
            WRITE(6,*)'FINAL VOLUME',X(1)
            WRITE(6,*)
            IF(GRAF)THEN
               WRITE(6,*)'CALLING PLOTTING ROUTINE'
               CALL PLOTEM(XMIN,XMAX,XPLT,YDEL,YAREA,YQI,YTEMP,KOUNT)
            ENDIF
            WRITE(6,*)'FINISHED '
            RETURN
         ENDIF
100   CONTINUE
      IF(ITER .GT. ITMAX)WRITE(6,*)'MAX # OF ITERATIONS EXCEEDED'
      WRITE(6,*)
      WRITE(6,*)'FINAL AREA',AREA
      WRITE(6,*)'FINAL DEL CALCITE',DELCAL
      WRITE(6,*)'FINAL DEL WATER',X(2)
      WRITE(6,*)'FINAL VOLUME',X(1)
      WRITE(6,*)
      IF(GRAF)THEN
         WRITE(6,*)'CALLING PLOTTING ROUTINE'
         CALL PLOTEM(XMIN,XMAX,XPLT,YDEL,YAREA,YQI,YTEMP,KOUNT)
      ENDIF
      END

******************************************************************************
******************************************************************************
*    Below lies a chaotic convolution of esoteric enigmas that hopefully     *
*    accomplish the isotopic and lake level voodoo we set out toodoo.        *
*    Actually this part of the program calculates the derivatives of lake    *
*    volume and isotopic composition with respect to time.                   *
******************************************************************************
******************************************************************************
      DOUBLE PRECISION FUNCTION F(I,TIME,TERM,T,KNT,KOUNT,TBEG)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      LOGICAL FOUND,GUESS,PASS,GRAF,CPARAM,ONLY1,ZEROVOL,ZEROCHK
      COMMON/TRADE1/GRAF,CALTEMP,DELCAL,TDELI,TTEMP
      DIMENSION TERM(3)

      COMMON/FIRST/INCHOICE,A,B,STIME(300),TEMP(300),EVAP(300),
     + HUM(300),DELA(300),DELI(300),GUESS,C,TEMPC,HUMC,EVAPC,DELAC,
     + DELIC,CPARAM,TEMPCURV,NPTS,HTIME(300),NHPTS
      PARAMETER(VOLMAX=39.898D9)
      EXTERNAL FQI,FINDT,FINDHT,OINTERP,HUMTERP,SAREA,FEPS,DELE
      INTRINSIC DEXP
      IF(I.EQ.1)THEN

******************************************************************************
******************************************************************************
*   A function subroutine to calculate the lake level history of San Agustin *
*   lake                                                                     *
******************************************************************************
******************************************************************************
*   Check for overflow                                                       *
******************************************************************************

D        WRITE(97,*)
D        WRITE(97,*)
```

```
          VOL = TERM(1)
          IF(VOL .GT. VOLMAX)THEN
          WRITE(6,*)'THE LAKE IS TOO BIG'
          STOP
          ENDIF

10        FORMAT(A,D12.5)

*******************************
** CONSTANT PARAMETER OPTION **
*******************************

          IF(CPARAM)THEN
             TDELI=DELIC
             TTEMP=TEMPC
             TEVAP=EVAPC
             TDELA=DELAC
             THUM=HUMC
          ELSE

*********************************************************************
** DETERMINE VALUES OF NECESSARY PARAMETERS BY ASSIGNING VALUES OR  **
** INTERPOLATING BETWEEN GIVEN VALUES                               **
*********************************************************************

             CALL FINDT(T,INDEX,FOUND,STIME,NPTS)
             CALL FINDHT(NHPTS,T,HINDEX,HTIME)

             IF(FOUND)THEN
                TDELI = DELI(INDEX)
                TTEMP = TEMP(INDEX)
                TEVAP = EVAP(INDEX)
                TDELA = DELA(INDEX)
                THUM = HUM(HINDEX)
             ELSE
                CALL OINTERP(T,INDEX,TDELI,TTEMP,TEVAP,TDELA)
             ENDIF
             CALL HUMTERP(T,HINDEX,THUM)
          ENDIF

*****************************************************************
** "REMEMBER" TEMP AT END OF TIMESTEP TO CALCULATE DEL CALCITE **
*****************************************************************

          IF(KNT .EQ. 5)CALTEMP=TTEMP

**********************************************
** TRACK TOTAL ELAPSED TIME TO CALCULATE INFLOW**
**********************************************

          ETIME = TBEG-T
D         WRITE(97,*)'TOTAL ELAPSED TIME :',ETIME

**************************************************
** ALSO NEED TO KNOW DEL TIME WITHIN THE TIME-STEP **
**************************************************

          DTIME = TIME-T

***************************
** CALCULATE AREA OF LAKE **
***************************
          AREA = SAREA(VOL)

*******************************
** CALCULATE EVAPORATION (QE) **
*******************************

          QE = AREA*TEVAP

********************************************
**  CALCULATE BACK-CONDENSATION FLUX (QC) **
********************************************

          QC = THUM*QE
```

```
*************************************************
** CALCULATE FLUX TO GROUNDWATER (QGW)        **
*************************************************

           QGW = 0.22D0*AREA

****************************************
**  CALCULATE dV/dT IF VOL IS ZERO **
****************************************

           IF(TERM(1) .LE. 0.D0)THEN
              IF(GUESS)THEN
                 WRITE(6,*)
                 WRITE(6,*)'THE INITIAL INFLOW IS :',B
                 WRITE(6,*)
                 WRITE(96,*)TBEG,AREA
                 WRITE(95,*)TBEG,B
                 DELCAL=FDCAL(TTEMP,TERM(2))
                 WRITE(99,*)TBEG,DELCAL
                 GUESS=.FALSE.
              ENDIF

              QI=FQI(ETIME)
              F=QI

              IF(QI .LT. 0.D0)QI=0.D0

              IF(F .LE. 0.D0)THEN
                 DV_DT=0.D0
              ELSE
                 DV_DT=F
              ENDIF

****************************************************
**  CALCULATE dV/dT IF VOL IS LESS THAN VOLMAX **
****************************************************

           ELSE IF(TERM(1)  .LT. VOLMAX)THEN

****************************************************
** PROVIDE AN INITIAL GUESS FOR QI SO THE MODEL**
** STARTS UNDER STABLE CONDITIONS              **
****************************************************

              IF(GUESS)THEN
                 WRITE(6,*)
                 WRITE(6,*)'QI =  ',B
                 WRITE(6,*)'QE = ',QE
                 WRITE(6,*)'QC = ',QC
                 WRITE(6,*)'QGW = ',QGW
                 WRITE(6,*)'dV/dt =',B+QC-QE-QGW
                 WRITE(6,*)
                 WRITE(6,*)'STEADY STATE QI WOULD BE',QE-QC+QGW
                 WRITE(6,*)'WOULD YOU LIKE TO CHANGE B (QI) ?'
                 WRITE(6,*)'1=YES     0=NO'
                 READ(5,*)IQI
                 IF(IQI .EQ. 1)THEN
                    WRITE(6,*)'INPUT NEW VALUE FOR B'
                    READ(5,*)B
                 ENDIF
                 WRITE(96,*)TBEG,AREA
                 WRITE(95,*)TBEG,B
                 DELCAL=FDCAL(TTEMP,TERM(2))
                 WRITE(99,*)TBEG,DELCAL
                 GUESS=.FALSE.
              ENDIF

****************************
** CALCULATE INFLOW (QI) **
****************************

              QI = FQI(ETIME)

**********************
**  CALCULATE dV/dT **
**********************
```

```
              F=QI+QC-QE-QGW
              DV_DT=F
D             WRITE(97,10)' QI=',QI
D             WRITE(97,10)' QC=',QC
D             WRITE(97,10)' QE=',QE
D             WRITE(97,10)' AREA=',AREA
D             WRITE(97,10)' EVAP=',TEVAP
D             WRITE(97,10)'QGW=',QGW
           END IF

D          WRITE(97,10)' dV/dT =',F

*******************************************************************************
* A function subroutine to calculate the isotopic history of San Agustin Lake *
*******************************************************************************

        ELSE IF(I.EQ.2)THEN

        DELL = TERM(2)
D       WRITE(97,10)' THE CURRENT DEL OF THE LAKE IS',DELL
D       WRITE(97,10)' THE CURRENT VOLUME OF THE LAKE IS',VOL

************************
** CALCULATE dDEL/dt  **
************************

        IF(VOL .LE. 10.D0)THEN
           F=0.D0

        ELSE

*******************************************
** CALCULATE ISOTOPIC ENRICHMENT FACTOR **
*******************************************

           EPS = FEPS(TTEMP)
D          WRITE(97,*)' EPSILON :',EPS

*********************************************
** CALCULATE DEL OF THE BACK-CONDENSATION **
*********************************************

           DELC =EPS*(1.D0+(TDELA/1.D3))+TDELA
D          WRITE(97,*)'DEL OF THE BACK-COND',DELC

**************************************
** CALCULATE DEL OF THE EVAPORATION **
**************************************

           DDELE = DELE(DELL,EPS,THUM)
D          WRITE(97,*)'DEL OF THE EVAP',DDELE

*******************************************************************
** SET DEL OF THE GROUNDWATER OUTFLOW EQUAL TO DEL OF THE LAKE **
*******************************************************************

           DELGW = DELL

           F=(QI*TDELI+QC*DELC-QGW*DELGW-QE*DDELE-DELL*DV_DT)/VOL

        ENDIF

D       WRITE(97,10)' dDEL/dt =',F
        END IF
        RETURN
        END

*******************************************************************************
*******************************************************************************
**        A function subprogram to calculate the area of San Agustin Lake   **
*******************************************************************************
*******************************************************************************

        double precision function sarea(vol)
        implicit double precision (a-h,o-z)
```

```
      TVOL=VOL/1.0D9

      if(tvol .ge. 0.0d0 .and. tvol .lt. 0.199d0) then
         sarea = 741.209d0*tvol - 2139.724d0*tvol**3
      else if(tvol .ge. 0.199d0 .and. tvol .lt. 0.667d0) then
         sarea = 130.809d0 + 486.108d0*(tvol-0.199d0) - 1279.662d0*
     +   (tvol-0.199d0)**2 + 964.959d0*(tvol-0.199d0)**3
      else if(tvol .ge. 0.667d0 .and. tvol .lt. 3.861d0) then
         sarea = 176.992d0 - 77.704d0*(tvol-0.667d0) + 73.258d0*
     +   (tvol-0.667d0)**2 - 9.780d0*(tvol-0.667d0)**3
      else if(tvol .ge. 3.861d0 .and. tvol .lt. 9.075d0) then
         sarea = 357.551d0 + 90.925d0*(tvol-3.861d0) - 20.474d0*
     +   (tvol-3.861d0)**2 + 1.600d0*(tvol-3.861d0)**3
      else if(tvol .ge. 9.075d0 .and. tvol .lt. 16.310d0) then
         sarea = 501.798d0 + 7.896d0*(tvol-9.075d0) + 4.549d0*
     +   (tvol-9.075d0)**2 - 0.283d0*(tvol-9.075d0)**3
      else if(tvol .ge. 16.310d0 .and. tvol .lt. 25.810d0)then
         sarea = 689.920d0 + 29.309d0*(tvol-16.310d0) - 1.589d0*
     +   (tvol-16.310d0)**2 + 0.055d0*(tvol-16.310d0)**3
      else if(tvol .ge. 25.810d0 .and. tvol .lt. 38.089d0)then
         sarea = 872.108d0 + 14.010d0*(tvol-25.810d0) - 0.021d0*
     +   (tvol-25.810d0)**2 + 0.058d0*(tvol-25.810d0)**3
      else if(tvol .ge. 38.089d0 .and. tvol .lt. 39.898d0) then
         sarea = 1148.520d0 + 39.772d0*(tvol-38.089d0) + 2.119d0*
     +   (tvol-38.089d0)**2 - 0.391d0*(tvol-38.089d0)**3
      else  if(tvol .eq. 39.898d0)then
         sarea = 1225.064d0

      end if

      sarea=sarea*1.0d6
      return
      end

******************************************************************************
******************************************************************************
*  This is a function subprogram to calculate the isotopic enrichment factor  *
******************************************************************************
******************************************************************************

      double precision function feps(temp)
      implicit double precision (a-h, o-z)
      intrinsic dexp

**************************
**CONVERT TEMP TO KELVIN**
**************************

      tempk=temp+273.15d0
      feps =(dexp((1.534d0*(1.0d6/(tempk)**2)-3.206d0*(1.0d3/tempk)+
     +       2.644d0)/1.d3)-1.d0)*1.d3
      return
      end

******************************************************************************
******************************************************************************
*      A subroutine to linearly interpolate between points in data sets      *
*              containing San Agustin Lake parameters                         *
******************************************************************************
******************************************************************************

      subroutine ointerp(time,ndx,tdeli,ttemp,tevap,tdela)

      implicit double precision (a-h,o-z)
      LOGICAL FOUND,GUESS,PASS,GRAF,CPARAM,ONLY1,ZEROVOL,ZEROCHK
      COMMON/FIRST/INCHOICE,A,B,STIME(300),TEMP(300),EVAP(300),
     + HUM(300),DELA(300),DELI(300),GUESS,C,TEMPC,HUMC,EVAPC,DELAC,
     + DELIC,CPARAM,TEMPCURV,NPTS,HTIME(300),NHPTS

      tdeli=(deli(ndx+1)-deli(ndx))/(stime(ndx+1)-stime(ndx))*
     +         (time-stime(ndx))+deli(ndx)

      ttemp=(temp(ndx+1)-temp(ndx))/(stime(ndx+1)-stime(ndx))*
     +         (time-stime(ndx))+temp(ndx)
```

```
      tevap=(evap(ndx+1)-evap(ndx))/(stime(ndx+1)-stime(ndx))*
     +        (time-stime(ndx))+evap(ndx)

      tdela = (dela(ndx+1)-dela(ndx))/(stime(ndx+1)-stime(ndx))*
     +        (time-stime(ndx))+dela(ndx)

      return
      end

*******************************************************************************
*******************************************************************************
*     A SUBROUTINE TO CALCULATE HUMIDITY FOR LAKE SAN AGUSTIN               **
*******************************************************************************
*******************************************************************************


      SUBROUTINE HUMTERP(TIME,HINDX,THUM)

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND,GUESS,PASS,GRAF,CPARAM,ONLY1,ZEROVOL,ZEROCHK
      COMMON/FIRST/INCHOICE,A,B,STIME(300),TEMP(300),EVAP(300),
     + HUM(300),DELA(300),DELI(300),GUESS,C,TEMPC,HUMC,EVAPC,DELAC,
     + DELIC,CPARAM,TEMPCURV,NPTS,HTIME(300),NHPTS

      thum=(hum(hindx+1)-hum(hindx))/(htime(hindx+1)-htime(hindx))*
     +       (time-htime(hindx))+hum(hindx)

      RETURN
      END

*******************************************************************************
*******************************************************************************
** A function subprogram to calculate the relative isotopic enrichment of   **
** the evaporating water.                                                    **
*******************************************************************************
*******************************************************************************

      double precision function dele(dell,eps,hum)
      implicit double precision(a-h,o-z)
      parameter(c=6.8d-3)

      dele=(((1.D0+1.D-3*dell)*(1.D0-c))/((1.D0+1.D-3*eps)*(1.D0-c*
     +     hum))-1.D0)*1.D03

      return
      end

*******************************************************************************
*******************************************************************************
**          A function subprogram to calculate the inflow, QI(t)            **
*******************************************************************************
*******************************************************************************

      DOUBLE PRECISION FUNCTION FQI(X)
      implicit double precision(a-h,o-z)
      LOGICAL FOUND,GUESS,PASS,GRAF,CPARAM,ONLY1,ZEROVOL,ZEROCHK
      COMMON/FIRST/INCHOICE,A,B,STIME(300),TEMP(300),EVAP(300),
     + HUM(300),DELA(300),DELI(300),GUESS,C,TEMPC,HUMC,EVAPC,DELAC,
     + DELIC,CPARAM,TEMPCURV,NPTS,HTIME(300),NHPTS
      INTRINSIC DEXP,DLOG10,DSIN

      IF(INCHOICE .EQ. 1)THEN
         FQI=A*X+B
      ELSE IF(INCHOICE .EQ. 2)THEN
         FQI=B*DEXP(A*X)
      ELSE IF(INCHOICE .EQ. 3)THEN
         IF(X .LT. 1.D0)X=1.D0
         FQI=B+A*DLOG10(X)
      ELSE IF(INCHOICE .EQ. 4)THEN
         FQI=B+A*X**C
      ELSE IF(INCHOICE .EQ. 5)THEN
         FQI=B+A*DSIN(C*X)
      ELSE IF(INCHOICE .EQ. 6)THEN
         IF(X .LT. 1.D0)THEN
```

```
            FQI=B
         ELSE
            FQI=B+A*B
         ENDIF
      ELSE IF(INCHOICE .EQ. 7)THEN
         FQI=0.D0
      ENDIF
      IF(FQI .LT. 0.D0)FQI=0.D0
      RETURN
      END

******************************************************************************
******************************************************************************
*             A SUBROUTINE TO FIND THE TIME WITH A BINARY SEARCH             *
******************************************************************************
******************************************************************************

      SUBROUTINE FINDT(TM,INDEX,FOUND,STIME,NPTS)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND,GUESS,PASS,GRAF,CPARAM,ONLY1,ZEROVOL,ZEROCHK
      DIMENSION STIME(300)
      TFIRST=1
      TLAST=NPTS
      FOUND = .FALSE.
      DO 200 I = 1,100
         IF(TLAST .LT. TFIRST) THEN
            INDEX=TLAST
            GOTO 210
         ENDIF
         IF( TFIRST .LE. TLAST .AND. FOUND .EQ. .FALSE.) THEN
            MIDDLE = (TFIRST+TLAST)/2
            TEST = ABS(TM-STIME(MIDDLE))
            IF( TEST .LT. 1.0D-1) THEN
               FOUND = .TRUE.
               INDEX=MIDDLE
               GOTO 210
            ELSE IF(TM .LT. STIME(MIDDLE))THEN
               TLAST = MIDDLE-1
            ELSE
               TFIRST = MIDDLE+1
            END IF
         END IF
200   CONTINUE

210   RETURN
      END


******************************************************************************
**    A SUBROUTINE TO FIND THE HUMIDITY TIME INDEX WITH A BINARY SEARCH     **
******************************************************************************

      SUBROUTINE FINDHT(NHPTS,TM,HINDEX,TIME)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND
      PARAMETER(NUMA=300)
      DIMENSION TIME(NUMA)
      TFIRST=1
      TLAST=NHPTS
      FOUND=.FALSE.
      DO 200 I=1,100
         IF(TLAST .LT. TFIRST) THEN
            HINDEX=TLAST
            GOTO 210
         ENDIF
         IF(TFIRST .LE. TLAST .AND. .NOT. FOUND)THEN
            MIDDLE = (TFIRST+TLAST)/2
            TEST = ABS(TM-TIME(MIDDLE))
            IF(TEST .LT. 1.0D-1) THEN
               FOUND = .TRUE.
               HINDEX = MIDDLE
               GOTO 210
            ELSE IF(TM .LT. TIME(MIDDLE)) THEN
               TLAST = MIDDLE-1
            ELSE
               TFIRST = MIDDLE+1
```

```
             END IF
          END IF
 200   CONTINUE

 210   RETURN
       END


       SUBROUTINE PLOTEM(XMIN,XMAX,XPLT,YDEL,YAREA,YQI,YTEMP,KOUNT)
*************************************************************************
*      SET UP THE GRAPHICS WINDOW FOR PLOTTING                           *
*************************************************************************
       REAL XMIN,XMAX,YMIN,YMAX,XPLT(300),YAREA(300),YQI(300),
     +     YDEL(300),STIME(300),SAISO(300)

       WRITE(6,*)
       WRITE(6,*)'READING SAN AGUSTIN ISOTOPE DATA'
       WRITE(6,*)

       OPEN(UNIT=60,FILE='OVAGE.DAT',STATUS='OLD')

       DO 100 I=1,300
          READ(60,*,END=101)STIME(I),SAISO(I)
 100   CONTINUE

 101   NPTS=I-1

       CLOSE(UNIT=60)

       WRITE(6,*)'READ',NPTS,'POINTS FROM SAN AGUSTIN DATA'

       DO 150 I=1,NPTS
          STIME(I)=STIME(I)*1.0E3
 150   CONTINUE

       CALL UIS

       CALL PAGE(33.,22.)
       CALL COMPLX
       CALL HEIGHT(.35)

C$$$        FIND THE POSITION TO PUT THE GRAPH
       DO 300 I=1,3
          IF(I .EQ. 1)THEN
             CALL PHYSOR(4.,12.)
          ELSE IF(I .EQ. 2)THEN
             CALL PHYSOR(4.,7.)
          ELSE IF(I .EQ. 3)THEN
             CALL PHYSOR(4.,2.)
          ENDIF

          IF(I .EQ. 1)THEN
             CALL AREA2D (25.,8.)
          ELSE
             CALL AREA2D (25.,3.)
          END IF
          call gapwid(.001)

C$$$  PUT THE HEADING ON THE PLOT

          IF(I .EQ. 1)THEN
             Call YNAME ('DEL O-18$',100)
             Call XNAME ('TIME$',100)
          ELSEIF(I .EQ. 2)THEN
             Call YNAME ('AREA$',100)
             Call XNAME ('TIME$',100)
          ELSEIF(I .EQ. 3)THEN
             Call YNAME ('INFLOW $',100)
             Call XNAME ('TIME$',100)
          ENDIF

*****************************
** CALCULATE YMAX AND YMIN **
*****************************

          IF(I .EQ. 1)THEN
```

```
            YMIN=YDEL(1)
            YMAX=YDEL(1)
            DO 1500 IMM = 2,KOUNT
               IF(YDEL(IMM) .LT. YMIN)YMIN=YDEL(IMM)
               IF(YDEL(IMM) .GT. YMAX)YMAX=YDEL(IMM)
 1500       CONTINUE
            YMIN=NINT(YMIN)-0.5
            YMAX=NINT(YMAX)+0.5
         ELSE IF(I .EQ. 2)THEN
            YMIN=YAREA(1)
            YMAX=YAREA(1)
            DO 1600 IMM = 2,KOUNT
               IF(YAREA(IMM) .LT. YMIN)YMIN=YAREA(IMM)
               IF(YAREA(IMM) .GT. YMAX)YMAX=YAREA(IMM)
 1600       CONTINUE
            YMIN=YMIN-0.1D0*YMIN
            YMAX=YMAX+0.1D0*YMAX
         ELSE IF(I .EQ. 3)THEN
            YMIN=YQI(1)
            YMAX=YQI(1)
            DO 1700 IMM = 2,KOUNT
               IF(YQI(IMM) .LT. YMIN)YMIN=YQI(IMM)
               IF(YQI(IMM) .GT. YMAX)YMAX=YQI(IMM)
 1700       CONTINUE
            YMIN=YMIN-0.1D0*YMIN
            YMAX=YMAX+0.1D0*YMIN
         ENDIF

         CALL GRAF (XMIN,XMIN,XMAX,YMIN,YMAX,YMAX)

         CALL FRAME
         CALL THKCRV(.02)
         CALL MARKER(3)
         CALL SCLPIC(.7)

         IF(I .EQ. 1)THEN
            CALL CURVE(XPLT,YDEL,KOUNT,-1)
            CALL CURVE(STIME,SAISO,NPTS,0)
         ELSE IF(I .EQ. 2)THEN
            CALL CURVE(XPLT,YAREA,KOUNT,-1)
         ELSE IF(I .EQ. 3)THEN
            CALL CURVE(XPLT,YQI,KOUNT,-1)
         ENDIF

         CALL ENDGR(IPLOT)

  300 CONTINUE

      RETURN
      END

************************************************************
************************************************************
**    A function subprogram to  calculate del calcite **
************************************************************
************************************************************

      DOUBLE PRECISION FUNCTION FDCAL(TEMP,DELH2O)
      implicit double precision (a-h,o-z)

************************************
** CONVERT TEMP TO DEGREES KELVIN **
************************************

      TEMPK = TEMP+273.15D0

************************************************
** CALCULATE EPSILON FOR CALCITE AND WATER **
************************************************

      EPSCAL=(DEXP((2.78D0*(1.D6/TEMPK**2)-2.89D0)/1.D3)-1.D0)*1.D3

D     WRITE(97,*)' EPSCAL',EPSCAL
D     WRITE(97,*)'TEMPK',TEMPK

      FDCAL=EPSCAL+DELH2O*(EPSCAL/1.0D3+1.D0)
```

Appendix O

Steady-State Isotopic Model

```
      PROGRAM SS.FOR

      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL OVER,DELCHK,TEMPCHK,HUMCHK

      DOUBLE PRECISION OQI(150),TEMPSL(50),SUMAREA(36,100,60),
     +     DELDOL_S(36,100,60),HUM(60),ISO(500),STIME(500),
     +     STEMP(2000),WTIME(2000)

      PARAMETER(AMAX_O=0.694D9,AMAX_C=0.155D9,AMAX_S=0.994D9,
     +     CONST_K=6.8D-3,AMAX_P=0.727D9,AMAX_D=0.583D9)

      WRITE(6,*)
      WRITE(6,*)'ENTER THE ACTIVITY OF WATER IN SEARLES LAKE'

      READ(5,*)AW

      WRITE(6,*)
      WRITE(6,*)'ENTER STARTING VALUE FOR QI'
      READ(5,*)STARTQI

      WRITE(6,*)
      WRITE(6,*)'ENTER ENDING VALUE FOR QI'
      READ(5,*)ENDQI

      HS=-2.2D0

      HI=135.D0

      QIINC=(ENDQI-STARTQI)/100.D0

      TEMP_I=6.25D0

      DO 100 I=1,36
         TEMP_I=TEMP_I+0.25D0
**          TEMP_I=15.2D0
         TEMPO=TEMP_I
         OQI_I=STARTQI
         TEMPSL(I)=TEMPO+3.6D0

         DO 200 J=1,100
            OQI_I=OQI_I+QIINC
            OQI(J)=OQI_I

            DO 300 K=1,60
               HUM(K)=DBLE(30.D0+K)/100.D0

*****************************
** OWENS LAKE CALCULATIONS **
*****************************
*********************************************************************
** CALCULATE ALL PARAMETERS THAT ARE A FUNCTION OF TEMPERATURE **
*********************************************************************

** DEL OF THE INFLOW **

               ODELI=TEMPO*0.289874D0 - 20.74367D0

** DEL OF THE ATMOSPHERE **

               DELA=TEMPO*2.898861D-1 - 35.79326D0
```

```
**  DEL OF THE PRECIPITATION **

              ODELP=TEMPO*2.915851D-1 - 1.60108D1

**  PRECIPITATION **

              OPRECIP= -1.63537110-2*TEMPO + 4.07238D-1
              IF(OPRECIP .LT. 0.D0)OPRECIP=0.D0

**  EVAPORATION **

              OEVAP=1.46896D-1*TEMPO - 6.37164D-1
              IF(OEVAP .LT. 1.35D0)OEVAP=1.35D0

**  EPSILON , THE ISOTOPIC ENRICHMENT FACTOR **

              EPS=FEPS(TEMPO)
              ALPHA=1.D0 + EPS/1.D3

**  DEL OF THE BACK-CONDENSATION **

              ODELC=EPS*(1.D0+(DELA/1.D3))+DELA

*******************************************
** NOW CALCULATE THE REST OF THE STUFF **
*******************************************

** CALCULATE AREA OF OWENS **

              AREA_O=OQI(j)/(OEVAP*(1.D0-HUM(K))-OPRECIP)
              IF (AREA_O .GT. AMAX_O)THEN
                 AREA_O=AMAX_O
              ENDIF

** GROSS EVAPORATION FLUX, QE (M^3/YR) **

              OQE=OEVAP*AREA_O

** PRECIP FLUX, QP (M^3/YR) **

              OQP=OPRECIP*AREA_O

** BACK-CONDENSATION FLUX, QC **

              OQC=OQE*HUM(K)

** OVERFLOW FLUX **

              OQO=OQI(j)+OQP+OQC-OQE

              IF(OQO .LE. 0.D0)OQO=0.D0

** KINETIC ISOTOPIC ENRICHMENT FACTOR, EPS_K **

              EPS_K=(ALPHA*(1.D0-CONST_K*HUM(K))/(1.D0-CONST_K)-
     +             1.D0)*1.D3

** DEL OF THE LAKE **

              ODELL=(OQI(J)*ODELI+OQP*ODELP+OQE*(HUM(K)*ODELC+EPS_K))/
     +             (OQI(J)+HUM(K)*OQE+OQP)

** CONVERT DEL WATER TO DEL DOLOMITE **

              DELDOL_O=FDDOL(TEMPO,ODELL)
```

```
** FLUX OUT OF OWENS EQUALS FLUX INTO CHINA **

              CQI=OQO

** SET DEL OF OWENS EQUAL TO THE DEL OF THE INFLOW TO CHINA LAKE **

              CDELI=ODELL

******************************
** CHINA LAKE CALCULATIONS **
******************************
*******************************************************************
** CALCULATE ALL PARAMETERS THAT ARE A FUNCTION OF TEMPERATURE **
*******************************************************************
              DELA=ODELA+8.1D0
              TEMPC = TEMPO+3.6D0

** DEL OF THE PRECIPITATION **

              CDELP=TEMPC*2.89886D-1 - 1.47368D1

** PRECIPITATION **

              CPRECIP=TEMPC*(-1.62597D-2)+4.05687D-1
              IF(CPRECIP .LT. 0.D0)CPRECIP=0.D0

**  EVAPORATION **

              CEVAP=1.46896D-1*TEMPC - 6.37164D-1
              IF(CEVAP .LT. 1.35D0)CEVAP=1.35D0

**  EPSILON , THE ISOTOPIC ENRICHMENT FACTOR **

              EPS=FEPS(TEMPC)
              ALPHA=1.D0 + EPS/1.D3

**  DEL OF THE BACK-CONDENSATION **

              CDELC=EPS*(1.D0+(DELA/1.D3))+DELA

*******************************************
** NOW CALCULATE THE REST OF THE STUFF **
*******************************************

** CALCULATE AREA OF CHINA LAKE **

              AREA_C=CQI/(CEVAP*(1.D0-HUM(K))-CPRECIP)

              IF (AREA_C .GT. AMAX_C)THEN
                 AREA_C=AMAX_C
              ENDIF

** GROSS EVAPORATION FLUX, QE (M^3/YR) **

              CQE=CEVAP*AREA_C

** PRECIP FLUX, QP (M^3/YR) **

              CQP=CPRECIP*AREA_C

** BACK-CONDENSATION FLUX **

              CQC=CQE*HUM(K)

** OVERFLOW FLUX **
```

```
      CQO=CQI+CQP+CQC-CQE

      IF(CQO .LE. 0.D0)CQO=0.D0

** KINETIC ISOTOPIC ENRICHMENT FACTOR, EPS_K **

      EPS_K=(ALPHA*(1.D0-CONST_K*HUM(K))/(1.D0-CONST_K)-
     +        1.D0)*1.D3

** DEL OF THE LAKE **

      IF(CQI .GT. 0.D0)THEN

        CDELL=(CQI*CDELI+CQP*CDELP+CQE*(HUM(K)*CDELC+EPS_K))/
     +          (CQI+HUM(K)*CQE+CQP)

      ENDIF

** CONVERT DEL WATER TO DEL DOLOMITE **

      DELDOL_C=FDDOL(TEMPC,CDELL)

** FLUX OUT OF CHINA EQUALS FLUX INTO SEARLES **

      SQI=CQO

** SET DEL OF CHINA EQUAL TO THE DEL OF THE INFLOW TO SEARLES LAKE **

      SDELI=CDELL

********************************
** SEARLES LAKE CALCULATIONS **
********************************
*******************************************************************
** CALCULATE ALL PARAMETERS THAT ARE A FUNCTION OF TEMPERATURE **
*******************************************************************

      TEMPS=TEMPC

** DEL OF THE PRECIPITATION **

      SDELP=TEMPS*2.89886D-1 - 1.47368D1

** PRECIPITATION **

      SPRECIP=TEMPS*(-1.62597D-2)+4.05687D-1
      IF(SPRECIP .LT. 0.D0)SPRECIP=0.D0

**  EVAPORATION **

      SEVAP=1.46896D-1*TEMPS - 6.37164D-1
      IF(SEVAP .LT. 1.35D0)SEVAP=1.35D0

**  EPSILON , THE ISOTOPIC ENRICHMENT FACTOR **

      EPS=FEPS(TEMPS)
      ALPHA=1.D0 + EPS/1.D3

**  DEL OF THE BACK-CONDENSATION **

      SDELC=EPS*(1.D0+(DELA/1.D3))+DELA

** KINETIC ISOTOPIC ENRICHMENT FACTOR, EPS_K **

      EPS_K=(ALPHA*(1.D0-CONST_K*HUM(K))/(1.D0-CONST_K)-1.D0)
     +        *1.D3
```

```
*********************************
** CALCULATE AREA OF SEARLES **
*********************************

              AREA_S=SQI/(SEVAP*(1.D0-HUM(K)/AW)-SPRECIP)
              IF(AREA_S .LT. 10.D0)AREA_S=0.D0

              OVER=.FALSE.

              IF(AREA_S .GT. 0.994D9)THEN
                 AREA_S=0.994D9
                 OVER=.TRUE.
              ENDIF

** GROSS EVAPORATION FLUX **

              SQE=SEVAP*AREA_S*AW

** PRECIP FLUX **

              SQP=SPRECIP*AREA_S

** BACK-CONDENSATION FLUX **

              SQC=SQE*HUM(K)/AW

** OH MY, WHICH EQUATION DO I USE **

** SEARLES OVERFLOWING **

              IF(OVER)THEN
                 SQI=OQO
                 SDELI=ODELL

                 SDELL=(SQI*SDELI+SQP*SDELP+SQE*(HUM(K)*SDELC+
     +                AW*EPS_K))/(SQI+HUM(K)*SQE+SQP)
                 SQO=SQI+SQP+SQC-SQE
                 AREA_C=0.D0

** SEARLES NOT OVERFLOWING, SEARLES AND CHINA COALESCED **

              ELSE IF(AREA_S.LE.0.994D9.AND.AREA_S.GE.0.715D9)THEN
                 SQI=OQO
                 SDELI=ODELL
                 SDELL=(SQI/SQE)*SDELI+EPS_K+HUM(K)*SDELC/AW+(SQP/SQE)
     +                *SDELP

                 SQO=0.D0
                 AREA_C=0.D0

** SEARLES NOT OVERFLOWING, SEARLES AND CHINA NOT COALESCED **

              ELSE IF(AREA_S.LT.0.715D9.AND.AREA_S.GT.0.D0)THEN
                 SQI=CQO
                 SDELI=CDELL
                 SDELL=(SQI/SQE)*SDELI+EPS_K+HUM(K)*SDELC/AW+(SQP/SQE)
     +                *SDELP
                 SQO=0.D0
              ELSE
                 SQO=0.D0
              ENDIF

** CONVERT DEL WATER TO DEL DOLOMITE **

              IF(AREA_S .GT. 0.D0)THEN
```

```
              DELDOL_S(I,J,K)=FDDOL(TEMPS,SDELL)
           ELSE
                DELDOL_S(I,J,K)=99.D0
           ENDIF

****************************
** PANAMINT CALCULATIONS **
****************************

           IF(SQO .GT. 0.D0)THEN

              TEMPP=TEMPS+2.35D0


** EVAPORATION **

              PEVAP=1.46896D-1*TEMPP - 6.37164D-1
              IF(PEVAP .LT. 1.35D0)PEVAP=1.35D0

** PRECIPITATION **

              PPRECIP=TEMPP*(-1.62787D-2)+4.05912D-1
              IF(PPRECIP .LT. 0.D0)PPRECIP=0.D0

** CALCULATE AREA OF PANAMINT **

              PQI=SQO

              AREA_P=PQI/(PEVAP*(1.D0-HUM(K))-PPRECIP)

              IF(AREA_P .GT. AMAX_P)THEN
                 AREA_P=AMAX_P
                 PQE=PEVAP*AREA_P
                 PQC=PQE*HUM(K)
                 PQP=PPRECIP*AREA_P
                 PQO=PQI+PQC+PQP-PQE
              ELSE
                 PQO=0.D0
              ENDIF

*********************************
** DEATH VALLEY CALCULATIONS **
*********************************

              IF(PQO .GT. 0.D0)THEN

                 TEMPD = TEMPP+4.08D0

** EVAPORTAION **

                 DEVAP=1.46896D-1*TEMPD - 6.37164D-1
                 IF(DEVAP .LT. 1.35D0)DEVAP=1.35D0

** PRECIPITATION **

                 DPRECIP=TEMPD*(-1.6289D-2)+4.06423D-1
                 IF(DPRECIP .LT. 0.D0)DPRECIP=0.D0

** CALCULATE AREA OF LAKE MANLY **

                 DQI=PQO

                 AREA_D=DQI/(DEVAP*(1.D0-HUM(K))-DPRECIP)

                 IF(AREA_D .GT. AMAX_D)THEN
                    AREA_D=AMAX_D
```

```
                  DQE=DEVAP*AREA_D
                  DQC=DQE*HUM(K)
                  DQP=DPRECIP*AREA_D
                  DQO=DQAI+DQC+DQP-DQE
               ELSE
                  DQO=0.D0
               ENDIF
            ELSE
               AREA_D=0.D0
            ENDIF
         ELSE
            AREA_D=0.D0
            AREA_P=0.D0
         ENDIF

         SUMAREA(i,j,k)=(AREA_O+AREA_C+AREA_S+AREA_P+AREA_D)/1.D9
300     CONTINUE
200   CONTINUE
100 CONTINUE

** READ CYNDY'S ISOTOPE DATA AND TIMES **

    OPEN(UNIT=99,FILE='REALISO.DAT',STATUS='OLD')

    DO 400 I=1,500
       READ(99,*,END=401)STIME(I),ISO(I)
400 CONTINUE

401 NPTS=I-1

    CLOSE(UNIT=99)

    DO 500 I=1,NPTS
       STIME(I)=STIME(I)*1.D6
500 CONTINUE

** FIND TEMPERATURES CORRESPONDING TO TIMES **

    OPEN(UNIT=98,FILE='SEARLES.UF',STATUS='OLD',FORM=
   +     'UNFORMATTED')

    DO 600 I=1,2000
       READ(98,END=601)STEMP(I),GBG1,GBG2,GBG3
600 CONTINUE

601 NSPTS=I-1

    CLOSE(UNIT=98)

    OPEN(UNIT=97,FILE='OWENS.UF',STATUS='OLD',FORM=
   +     'UNFORMATTED')

    DO 700 I=1,NSPTS
       READ(97)WTIME(I),GBG1,GBG2,GBG3,GBG4
700 CONTINUE

    DO 725 I=1,NSPTS
       WTIME(I)=WTIME(I)*1.D6
725 CONTINUE

    OPEN(UNIT=96,FILE='SSQI.DAT',STATUS='UNKNOWN')

    OPEN(UNIT=92,FILE='STEMP.DAT',STATUS='UNKNOWN')

    DO 800 I=1,NPTS
       TEMPCHK=.FALSE.
```

```
        DELCHK=.FALSE.
        HUMCHK=.FALSE.

        CALL FINDT(NSPTS,STIME(I),INDEX,WTIME)
        CALL SINTERP(STIME(I),INDEX,TEMP,STEMP,WTIME)

** WRITE SEARLES TEMP TO FILE **

        WRITE(92,*)STIME(I)/1.D6,TEMP

** CALCULATE HUMIDITY FROM DEL **

        SHUM=DBLE(ANINT(HI+HS*ISO(I)))

** FIND CORRESPONDING HUMIDITY IN ARRAY **

        DO 900 I1=1,60
           IF(DABS(SHUM-HUM(I1)*100.D0) .LT. 0.01D0)THEN
              HUMCHK=.TRUE.
              GOTO 901
           ENDIF
 900    CONTINUE

 901    NK=I1

** DON'T ENTER THIS LOOP UNLESS THE PROGRAM FOUND A HUMIDITY MATCH **

        IF(HUMCHK)THEN

           DO 1000 II=1,36
           IF(TEMP .GT. TEMPSL(II) .AND. TEMP .LT. TEMPSL(II+1))THEN
                TEMPCHK=.TRUE.
                TEST1=DABS(TEMP-TEMPSL(II))
                TEST2=DABS(TEMP-TEMPSL(II+1))
                GOTO 1001
             ENDIF
 1000      CONTINUE

 1001      IF(TEST1 .GT. TEST2)THEN
             NI=II+1
           ELSE
             NI=II
           ENDIF

** LOCATE DELS IN ARRAY THAT CORRESPOND TO GIVEN DEL **

           CHCK=ISO(I)-DELDOL_S(NI,1,NK)

           IF(CHCK .LT. 0.D0 .AND. TEMPCHK)THEN
              DO 1100 IJ=1,100
                 CHCK=ISO(I)-DELDOL_S(NI,IJ,NK)
                 IF(CHCK .GT. 0.D0)THEN
                    NJ=IJ
                    DELCHK=.TRUE.
                    GOTO 1101
                 ENDIF
 1100         CONTINUE
           ENDIF

 1101      IF(DELCHK)THEN
              FQI=(OQI(NJ)-OQI(NJ-1))/(DELDOL_S(NI,NJ-1,NK)-
     +              DELDOL_S(NI,NJ,NK))*(DELDOL_S(NI,NJ-1,NK)
     +               -ISO(I))+OQI(NJ-1)

              FAREA=(SUMAREA(NI,NJ,NK)-SUMAREA(NI,NJ-1,NK))/
     +              (DELDOL_S(NI,NJ-1,NK)-DELDOL_S(NI,NJ,NK))*
```

```
     +            (DELDOL_S(NI,NJ-1,NK)-ISO(I))+SUMAREA(NI,NJ-1,NK)
               WRITE(96,*)STIME(I)/1.D6,FAREA,FQI
            ELSEIF(TEMPCHK)THEN
               WRITE(96,*)STIME(I)/1.D6,99,REAL(DELDOL_S(NI,1,NK)),
     +               REAL(DELDOL_S(NI,30,NK)),REAL(ISO(I))
            ELSE
               WRITE(96,*)STIME(I)/1.D6,98,TEMP
            ENDIF
         ENDIF

         IF(.NOT. HUMCHK)THEN
            WRITE(96,*)STIME(I)/1.D6,97,REAL(HUM(1)),REAL(HUM(50)),SHUM
         ENDIF

 800  CONTINUE

      CLOSE(UNIT=96)
      CLOSE(UNIT=92)

      END


*****************************************************************************
*****************************************************************************
*  This is a function subprogram to calculate the isotopic enrichment factor  *
*****************************************************************************
*****************************************************************************

        double precision function feps(temp)
        implicit double precision (a-h, o-z)

**************************
**CONVERT TEMP TO KELVIN**
**************************

        tempk=temp+273.15d0
        feps =(dexp((1.534d0*(1.0d6/(tempk)**2)-3.206d0*(1.0d3/tempk)+
     +       2.644d0)/1.d3)-1.d0)*1.d3
        return
        end




*****************************************************************
*****************************************************************
**    A function subprogram to  calculate del dolomite **
*****************************************************************
*****************************************************************

        DOUBLE PRECISION FUNCTION FDDOL(TEMP,DELH2O)
        implicit double precision (a-h,o-z)

**************************************
** CONVERT TEMP TO DEGREES KELVIN **
**************************************

        TEMPK = TEMP+273.15D0

************************************************
** CALCULATE EPSILON FOR DOLOMITE AND WATER **
************************************************

        EPSDOL=(DEXP((3.2D0*(1.D6/TEMPK**2)-4.3D0)/1.D3)-1.D0)*1.D3
```

```
      FDDOL=EPSDOL+DELH2O*(EPSDOL/1.0D3+1.D0)

      RETURN
      END


*****************************************************************************
*****************************************************************************
*        A SUBROUTINE TO FIND THE TIME INDEX WITH A BINARY SEARCH          *
*****************************************************************************
*****************************************************************************

      SUBROUTINE FINDT(NPTS,TM,INDEX,TIME)
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      LOGICAL FOUND
      PARAMETER(NUMA=2000)
      DIMENSION TIME(NUMA)
      TFIRST=1
      TLAST=NPTS
      FOUND = .FALSE.
      DO 200 I = 1,100
         IF(TLAST .LT. TFIRST) THEN
            INDEX=TLAST
            GOTO 210
         ENDIF
         IF( TFIRST .LE. TLAST .AND. .NOT. FOUND)THEN
            MIDDLE = (TFIRST+TLAST)/2
            TEST = ABS(TM-TIME(MIDDLE))
            IF( TEST .LT. 1.0D-1) THEN
               FOUND = .TRUE.
               INDEX=MIDDLE
               GOTO 210
            ELSE IF(TM .LT. TIME(MIDDLE))THEN
               TLAST = MIDDLE-1
            ELSE
               TFIRST = MIDDLE+1
            END IF
         END IF
200   CONTINUE

210   RETURN
      END


*****************************************************************************
*****************************************************************************
*     A subroutine to linearly interpolate between points in data sets     *
*                  containing Searles Lake parameters                      *
*****************************************************************************
*****************************************************************************

      subroutine sinterp(time,ndx,tstemp,STEMP,WTIME)

      implicit double precision (a-h,o-z)

      DIMENSION WTIME(2000),STEMP(2000)

      tstemp=(stemp(ndx+1)-stemp(ndx))/(wtime(ndx+1)-wtime(ndx))*
     +     (time-wtime(ndx))+stemp(ndx)

      return
      end
```

Appendix P

Bathymetric Data for Lakes in the Paleo-Owens River

```
*******************************************************************************
*                                                                             *
*   This file contains routing data for the Owens River Lake System           *
*   Data for Owens and Searles are from Smith( 1979 )                         *
*   Data for China Lake from Jannik (1989)                                    *
*   Data for Panamint is from Smith, R.S.U. (      )                          *
*   Data from Death Valley is from Jannik (1989)                              *
*                                                                             *
*******************************************************************************
```

CUMULATIVE DEPTH (m)	CUMULATIVE AREA (x10^9 m^2)	LAKE AREA (x10^9 m^2)	ADDED VOLUME (x10^9 m^3)	CUMULATIVE LAKE VOL. (x10^9 m^3)	K AREA/VOLUME
0.0	0.0	0.0	0.0	0.0	0
2.4	0.20	0.20	0.16	0.16	1.25
14.0	0.29	0.29	2.99	3.15	0.092
64.0	0.694	0.694	26.87	30.02	0.023
****************** OWENS OVERFLOW ***************************************					
68.0	0.721	0.027	0.036	0.036	0.75
72.0	0.849	0.155	0.660	0.696	.223
****************** CHINA OVERFLOW ***************************************					
97.0	1.094	0.245	2.04	2.04	.120
129.0	1.149	0.300	8.71	10.75	.028
160.0	1.199	0.350	10.07	20.82	.017
192.0	1.291	0.442	12.64	33.46	.013
219.0	1.382	0.533	13.14	46.60	.011
250.0	1.564	0.715	19.27	65.87	.011
***** China coalesces with Searles *****					
272.0	1.688	0.994	18.71	85.28	.012
****************** SEARLES OVERFLOW *************************************					
290.0	1.806	0.118	0.71	0.71	.166
310.0	1.863	0.175	2.91	3.62	.048
321.0	1.877	0.189	2.00	5.62	.034
351.0	1.930	0.242	6.45	12.07	.020
382.0	1.977	0.289	8.22	20.29	.014
412.0	2.017	0.329	9.26	29.55	.011
443.0	2.057	0.369	10.81	40.36	.009
478.0	2.116	0.428	13.93	54.29	.008
498.0	2.176	0.488	9.15	63.44	.008
513.0	2.212	0.524	7.59	71.03	.007
533.0	2.256	0.568	10.92	81.95	.007
559.0	2.326	0.638	15.67	97.62	.007
574.0	2.415	0.727	10.23	107.85	.007
***************** PANAMINT OVERFLOW ************************************					
613.0	2.465	0.050	0.65	0.65	
786.0	2.998	0.583	46.35	47.0	

Appendix Q

Model Inputs for Lake San Agustin

AGE (ka)	TEMPERATURE (Celsius)	EVAPORATION (mm/yr)	HUMIDITY	OXYGEN-18 OF INFLOW	OXYGEN-18 ATMOSPHERE
15.36	4.76	0.97	0.30	-14.01	-26.07
15.49	5.60	1.03	0.34	-14.12	-25.48
15.77	5.32	1.01	0.39	-14.35	-25.68
16.01	6.16	1.06	0.43	-14.55	-25.09
16.31	5.88	1.05	0.35	-14.78	-25.29
16.45	5.88	1.05	0.34	-14.58	-25.29
16.59	5.32	1.01	0.45	-14.38	-25.68
16.66	5.04	0.99	0.42	-14.29	-25.87
17.00	5.60	1.03	0.48	-13.80	-25.48
17.13	5.88	1.05	0.52	-13.92	-25.29
17.21	6.72	1.10	0.48	-13.99	-24.71
17.30	6.44	1.08	0.48	-14.07	-24.90
17.41	5.32	1.01	0.48	-14.17	-25.68
17.54	5.60	1.03	0.41	-14.29	-25.48
17.62	5.32	1.01	0.40	-14.35	-25.68
17.68	5.32	1.01	0.33	-14.41	-25.68
17.82	4.48	0.95	0.37	-14.54	-26.26
17.95	3.92	0.92	0.35	-14.81	-26.65
18.09	3.08	0.86	0.43	-14.49	-27.24
18.23	1.68	0.77	0.34	-14.16	-28.22
18.30	6.44	1.08	0.51	-14.00	-24.90
18.43	6.72	1.10	0.40	-13.68	-24.71
18.57	6.72	1.10	0.48	-13.37	-24.71
18.67	6.72	1.10	0.36	-13.13	-24.71
18.77	6.72	1.10	0.40	-12.88	-24.71
18.88	6.72	1.10	0.36	-12.63	-24.71
18.93	6.16	1.06	0.40	-12.31	-25.09
19.07	6.16	1.06	0.47	-12.38	-25.09
19.17	4.62	0.96	0.45	-12.44	-26.16
19.30	3.92	0.92	0.47	-12.51	-26.65
19.41	6.44	1.08	0.61	-12.56	-24.90
19.52	6.72	1.10	0.34	-12.62	-24.71
19.67	4.48	0.95	0.36	-12.70	-26.26
19.72	5.32	1.01	0.31	-12.73	-25.68
19.83	6.72	1.10	0.42	-12.79	-24.71
19.93	6.44	1.08	0.37	-12.84	-24.90
20.04	6.72	1.10	0.40	-12.90	-24.71
20.15	3.92	0.92	0.45	-12.95	-26.65
20.36	6.72	1.10	0.39	-13.06	-24.71
20.47	3.64	0.90	0.50	-13.12	-26.85
20.57	1.96	0.79	0.43	-13.17	-28.02
20.67	1.12	0.73	0.36	-13.23	-28.61
20.78	1.96	0.79	0.43	-13.29	-28.02
20.83	1.96	0.79	0.34	-13.31	-28.02
20.94	2.24	0.81	0.48	-13.35	-27.83

20.99	1.40	0.75	0.44	-13.39	-28.41
21.10	4.20	0.94	0.37	-13.48	-26.46
21.20	3.08	0.86	0.39	-13.57	-27.24
21.31	1.12	0.73	0.39	-13.65	-28.61
21.42	1.68	0.77	0.42	-13.74	-28.22
21.49	2.24	0.81	0.42	-13.81	-27.83
21.60	1.96	0.79	0.45	-13.89	-28.02
21.73	1.68	0.77	0.40	-14.00	-28.22
21.83	5.88	1.05	0.39	-14.09	-25.29
21.94	6.72	1.10	0.35	-14.18	-24.71
22.05	5.60	1.03	0.38	-14.27	-25.48
22.31	5.04	0.99	0.44	-14.48	-25.87
22.37	4.76	0.97	0.46	-14.53	-26.07
22.42	5.88	1.05	0.60	-14.57	-25.29
22.49	4.76	0.97	0.53	-14.63	-26.07
22.57	2.52	0.82	0.55	-14.70	-27.63
22.63	3.64	0.90	0.55	-14.74	-26.85
22.68	4.48	0.95	0.49	-14.79	-26.26
22.73	4.76	0.97	0.46	-14.83	-26.07
22.97	5.04	0.99	0.29	-15.05	-25.87
23.05	5.60	1.03	0.42	-15.04	-25.48
23.10	6.44	1.08	0.41	-15.03	-24.90
23.21	5.88	1.05	0.39	-15.02	-25.29
23.32	6.72	1.10	0.54	-15.00	-24.71
23.37	6.44	1.08	0.41	-15.00	-24.90
23.49	6.44	1.08	0.36	-14.98	-24.90
23.58	6.72	1.10	0.42	-14.96	-24.71
23.63	6.02	1.05	0.48	-14.96	-25.19
23.73	6.16	1.06	0.44	-14.94	-25.09
23.84	5.88	1.05	0.33	-14.92	-25.29
23.95	5.60	1.03	0.45	-14.91	-25.48
24.05	6.44	1.08	0.45	-14.89	-24.90
24.21	6.44	1.08	0.41	-14.87	-24.90
24.27	6.16	1.06	0.40	-14.86	-25.09
24.37	6.16	1.06	0.43	-14.84	-25.09
24.47	5.32	1.01	0.44	-14.83	-25.68
24.58	4.76	0.97	0.48	-14.81	-26.07
24.63	3.92	0.92	0.44	-14.81	-26.65
24.74	4.48	0.95	0.53	-14.79	-26.26
24.84	3.92	0.92	0.66	-14.77	-26.65
24.90	4.48	0.95	0.64	-14.77	-26.26
25.00	4.48	0.95	0.64	-14.75	-26.26
25.11	5.04	0.99	0.54	-14.73	-25.87
25.16	5.32	1.01	0.48	-14.73	-25.68
25.22	6.16	1.06	0.46	-14.72	-25.09
25.27	6.16	1.06	0.40	-14.71	-25.09
25.37	3.92	0.92	0.64	-14.69	-26.65
25.42	5.88	1.05	0.66	-14.69	-25.29
25.53	6.30	1.07	0.69	-14.67	-25.00

25.63	5.88	1.05	0.57	-14.65	-25.29
25.74	6.44	1.08	0.50	-14.64	-24.90
25.79	6.44	1.08	0.40	-14.63	-24.90
25.85	6.44	1.08	0.46	-14.62	-24.90
25.90	6.44	1.08	0.47	-14.62	-24.90
26.01	3.92	0.92	0.61	-14.60	-26.65
26.06	4.20	0.94	0.55	-14.60	-26.46
26.17	4.48	0.95	0.72	-14.59	-26.26
26.27	4.48	0.95	0.65	-14.59	-26.26
26.37	4.48	0.95	0.60	-14.58	-26.26
26.42	4.20	0.94	0.60	-14.58	-26.46
26.48	3.92	0.92	0.56	-14.58	-26.65
26.58	2.52	0.82	0.47	-14.57	-27.63
26.69	4.20	0.94	0.48	-14.57	-26.46
26.84	6.44	1.08	0.40	-14.56	-24.90
26.90	2.80	0.84	0.43	-14.56	-27.43
27.01	4.20	0.94	0.41	-14.56	-26.46
27.12	4.76	0.97	0.35	-14.55	-26.07
27.17	6.16	1.06	0.35	-14.55	-25.09
27.22	6.16	1.06	0.37	-14.55	-25.09
27.27	6.44	1.08	0.41	-14.54	-24.90
27.32	6.58	1.09	0.44	-14.54	-24.80
27.46	5.60	1.03	0.45	-14.54	-25.48
27.53	6.16	1.06	0.43	-14.53	-25.09
27.64	6.44	1.08	0.40	-14.53	-24.90
27.75	6.44	1.08	0.47	-14.52	-24.90
27.80	6.16	1.06	0.41	-14.52	-25.09
27.91	5.32	1.01	0.48	-14.52	-25.68
27.96	4.20	0.94	0.47	-14.51	-26.46
28.07	5.60	1.03	0.48	-14.51	-25.48
28.17	1.96	0.79	0.43	-14.50	-28.02
28.27	5.60	1.03	0.41	-14.50	-25.48
28.38	5.88	1.05	0.44	-14.49	-25.29
28.43	6.72	1.10	0.30	-14.49	-24.71
28.54	5.88	1.05	0.42	-14.49	-25.29
28.67	1.26	0.74	0.48	-14.48	-28.51
28.80	1.12	0.73	0.52	-14.48	-28.61
28.91	1.26	0.74	0.52	-14.47	-28.51
29.02	1.12	0.73	0.50	-14.47	-28.61
29.12	1.12	0.73	0.46	-14.46	-28.61
29.17	1.26	0.74	0.52	-14.46	-28.51
29.22	1.26	0.74	0.57	-14.46	-28.51
29.33	1.12	0.73	0.49	-14.45	-28.61
29.43	1.26	0.74	0.45	-14.45	-28.51
29.58	1.12	0.73	0.45	-14.44	-28.61
29.78	1.12	0.73	0.52	-14.43	-28.61
29.94	1.26	0.74	0.54	-14.42	-28.51
30.58	1.12	0.73	0.48	-14.40	-28.61
30.71	1.12	0.73	0.49	-14.39	-28.61

34.76	1.12	0.73	0.56	-14.21	-28.61
34.81	1.12	0.73	0.46	-14.21	-28.61
34.87	1.12	0.73	0.50	-14.21	-28.61
34.95	1.12	0.73	0.60	-14.18	-28.61
35.06	1.12	0.73	0.46	-14.22	-28.61
35.16	1.12	0.73	0.67	-14.26	-28.61
35.26	1.12	0.73	0.65	-14.31	-28.61
35.37	1.12	0.73	0.67	-14.35	-28.61

Age (Ka)	Relative Humidity

15.36288	0.3043299
15.49394	0.3410800
15.77243	0.3917599
16.00997	0.4318697
16.31304	0.3515799
16.45229	0.3421998
16.59153	0.4450300
16.65706	0.4165399
17.00108	0.4757600
17.13214	0.5209799
17.20586	0.4811499
17.30415	0.4809399
17.41063	0.4771600
17.54169	0.4145799
17.61541	0.4047098
17.68094	0.3333099
17.82018	0.3675399
17.95124	0.3508098
18.09049	0.4307499
18.22973	0.3412898
18.29974	0.5090098
18.43274	0.4009998
18.56573	0.4841599
18.66706	0.3611698
18.77472	0.4028900
18.88238	0.3640399
18.93304	0.4022598
19.06604	0.4721198
19.16736	0.4453800
19.30036	0.4745698
19.40802	0.6065199
19.51568	0.3400300
19.67400	0.3601198
19.72467	0.3061500
19.83233	0.4209497
19.93366	0.3706200
20.04132	0.3963797
20.14898	0.4521000
20.35797	0.3901498
20.46563	0.5040398
20.56696	0.4317999
20.67462	0.3627098
20.78228	0.4267597
20.83294	0.3438799
20.94060	0.4832499
20.99127	0.4367700
21.09893	0.3741899
21.20026	0.3932297
21.30792	0.3920400
21.41558	0.4189899
21.49158	0.4205298
21.59924	0.4499300
21.73223	0.4032400
21.83356	0.3867898
21.94122	0.3472400
22.04888	0.3820999
22.30853	0.4373300
22.36553	0.4586799
22.41619	0.6010599
22.49219	0.5337899
22.57452	0.5468800
22.62518	0.5452700
22.68218	0.4933999
22.73284	0.4560900
22.96716	0.2884400
23.04949	0.4192700
23.10016	0.4072299
23.20782	0.3861599
23.31548	0.5395298
23.36614	0.4066699
23.48647	0.3586500
23.57513	0.4192700
23.63213	0.4763899
23.73346	0.4392200
23.84112	0.3256099
23.94878	0.4513998

24.05011	0.4474800
24.20843	0.4108698
24.26543	0.3999498
24.36676	0.4250798
24.47442	0.4398499
24.58208	0.4767399
24.63274	0.4373999
24.74040	0.5322499
24.84173	0.6556599
24.89873	0.6448100
25.00006	0.6442498
25.10772	0.5377798
25.15838	0.4770198
25.21538	0.4598699
25.26604	0.3960998
25.37370	0.6436899
25.42437	0.6574799
25.53203	0.6891899
25.63336	0.5680900
25.74102	0.4980898
25.79168	0.3967998
25.84868	0.4576299
25.89934	0.4730999
26.00700	0.6109300
26.05767	0.5493999
26.16533	0.7194999
26.26666	0.6472600
26.37432	0.6008499
26.42498	0.5982599
26.48198	0.5568900
26.58331	0.4662399
26.69097	0.4765999
26.83663	0.3980598
26.89996	0.4309599
27.00762	0.4125500
27.11528	0.3506699
27.16594	0.3465397
27.21661	0.3720200
27.27360	0.4061098
27.32427	0.4425099
27.45726	0.4518900
27.53326	0.4326398
27.64092	0.4025400
27.74858	0.4675698
27.79924	0.4146500
27.90690	0.4789798
27.95757	0.4679198
28.06523	0.4827600
28.16656	0.4299099
28.27422	0.4101698
28.38188	0.4404800
28.43254	0.2978899
28.54020	0.4184997
28.66686	0.4788399
28.79986	0.5246198
28.90752	0.5173399
29.01518	0.5006800
29.11651	0.4609900
29.17350	0.5209100
29.22417	0.5690000
29.33183	0.4938900
29.43316	0.4468498
29.57900	0.4518900
29.78147	0.5193698
29.93980	0.5403700
30.57943	0.4772298
30.71242	0.4879398
34.76433	0.5628397
34.81499	0.4640000
34.87199	0.5017300
34.94799	0.5978398
35.05565	0.4638598
35.15697	0.6703600
35.26464	0.6525098
35.37230	0.6676298

4.130000	15.36288	33.08100	-0.5588126	4.760000
4.290000	15.49394	32.55600	-0.8503196	5.600000
4.630000	15.77243	31.83200	-1.622758	5.320000
4.920000	16.00997	31.25900	-1.962254	6.160000
5.290000	16.31304	32.40600	-0.9236690	5.880000
5.460000	16.45229	32.54000	-0.7939924	5.880000
5.630000	16.59153	31.07100	-2.359086	5.320000
5.710000	16.65706	31.47800	-2.037337	5.040000
6.130000	17.00108	30.63200	-2.712072	5.600000
6.290000	17.13214	29.98600	-3.265541	5.880000
6.380000	17.20586	30.55500	-2.501568	6.720000
6.500000	17.30415	30.55800	-2.569559	6.440000
6.630000	17.41063	30.61200	-2.803204	5.320000
6.790000	17.54169	31.50600	-1.866348	5.600000
6.880000	17.61541	31.64700	-1.801761	5.320000
6.960000	17.68094	32.66700	-0.8148321	5.320000
7.130000	17.82018	32.17800	-1.504843	4.480000
7.290000	17.95124	32.41700	-1.419429	3.920000
7.460000	18.09049	31.27500	-2.743968	3.080000
7.630000	18.22973	32.55300	-1.879657	1.680000
7.710000	18.29974	30.15700	-2.957670	6.440000
7.920000	18.43274	31.70000	-1.393295	6.720000
8.130000	18.56573	30.51200	-2.543190	6.720000
8.290000	18.66706	32.26900	-0.8425466	6.720000
8.460000	18.77472	31.67300	-1.419430	6.720000
8.630000	18.88238	32.22800	-0.8822318	6.720000
8.710000	18.93304	31.68200	-1.552882	6.160000
8.920000	19.06604	30.68400	-2.518732	6.160000
9.080000	19.16736	31.06600	-2.544323	4.620000
9.290000	19.30036	30.64900	-3.129484	3.920000
9.460000	19.40802	28.76400	-4.305891	6.440000
9.630000	19.51568	32.57100	-0.5502355	6.720000
9.880000	19.67400	32.28400	-1.402303	4.480000
9.960000	19.72467	33.05500	-0.4394116	5.320000
10.13000	19.83233	31.41500	-1.669153	6.720000
10.29000	19.93366	32.13400	-1.044222	6.440000
10.46000	20.04132	31.76600	-1.329412	6.720000
10.63000	20.14898	30.97000	-2.819005	3.920000
10.96000	20.35797	31.85500	-1.243268	6.720000
11.13000	20.46563	30.22800	-3.609753	3.640000
11.29000	20.56696	31.26000	-3.054865	1.960000
11.46000	20.67462	32.24700	-2.325596	1.120000
11.63000	20.78228	31.33200	-2.985260	1.960000
11.71000	20.83294	32.51600	-1.840659	1.960000
11.88000	20.94060	30.52500	-3.691121	2.240000
11.96000	20.99127	31.18900	-3.273060	1.400000
12.13000	21.09893	32.08300	-1.669504	4.200000
12.29000	21.20026	31.81100	-2.225648	3.080000
12.46000	21.30792	31.82800	-2.730565	1.120000
12.63000	21.41558	31.44300	-2.952642	1.680000
12.75000	21.49158	31.42100	-2.824870	2.240000
12.92000	21.59924	31.00100	-3.305247	1.960000
13.13000	21.73223	31.66800	-2.735146	1.680000
13.29000	21.83356	31.90300	-1.410429	5.880000
13.46000	21.94122	32.46800	-0.6499322	6.720000
13.63000	22.04888	31.97000	-1.417361	5.600000
14.04000	22.30853	31.18100	-2.324688	5.040000
14.13000	22.36553	30.87600	-2.692014	4.760000
14.21000	22.41619	28.84200	-4.372610	5.880000
14.33000	22.49219	29.80300	-3.730074	4.760000
14.46000	22.57452	29.61600	-4.495824	2.520000
14.54000	22.62518	29.63900	-4.179409	3.640000
14.63000	22.68218	30.38000	-3.244171	4.480000
14.71000	22.73284	30.91300	-2.656218	4.760000
15.08000	22.96716	33.30800	-0.2668005	5.040000
15.21000	23.04949	31.43900	-1.931182	5.600000
15.29000	23.10016	31.61100	-1.550408	6.440000
15.46000	23.20782	31.91200	-1.401719	5.880000
15.63000	23.31548	29.72100	-3.308816	6.720000
15.71000	23.36614	31.61900	-1.542666	6.440000
15.90000	23.48647	32.30500	-0.8787170	6.440000
16.04000	23.57513	31.43900	-1.645925	6.720000
16.13000	23.63213	30.62300	-2.613440	6.020000
16.29000	23.73346	31.15400	-2.063873	6.160000
16.46000	23.84112	32.77700	-0.5646440	5.880000
16.63000	23.94878	30.98000	-2.375331	5.600000

16.79000	24.05011	31.03600	-2.106926	6.440000
17.04000	24.20843	31.55900	-1.600737	6.440000
17.13000	24.26543	31.71500	-1.520944	6.160000
17.29000	24.36676	31.35600	-1.868379	6.160000
17.46000	24.47442	31.14500	-2.287484	5.320000
17.63000	24.58208	30.61800	-2.941612	4.760000
17.71000	24.63274	31.18000	-2.615887	3.920000
17.88000	24.74040	29.82500	-3.781059	4.480000
18.04000	24.84173	28.06200	-5.631698	3.920000
18.13000	24.89873	28.21700	-5.336587	4.480000
18.29000	25.00006	28.22500	-5.328846	4.480000
18.46000	25.10772	29.74600	-3.713060	5.040000
18.54000	25.15838	30.61400	-2.801268	5.320000
18.63000	25.21538	30.85900	-2.349370	6.160000
18.71000	25.26604	31.77000	-1.467716	6.160000
18.88000	25.37370	28.23300	-5.466302	3.920000
18.96000	25.42437	28.03600	-5.152589	5.880000
19.13000	25.53203	27.58300	-5.484382	6.300000
19.29000	25.63336	29.31300	-3.916815	5.880000
19.46000	25.74102	30.31300	-2.806684	6.440000
19.54000	25.79168	31.76000	-1.406198	6.440000
19.63000	25.84868	30.89100	-2.247263	6.440000
19.71000	25.89934	30.67000	-2.461160	6.440000
19.88000	26.00700	28.70100	-5.013640	3.920000
19.96000	26.05767	29.58000	-4.090648	4.200000
20.13000	26.16533	27.15000	-6.368767	4.480000
20.29000	26.26666	28.18200	-5.370445	4.480000
20.46000	26.37432	28.84500	-4.729080	4.480000
20.54000	26.42498	28.88200	-4.765821	4.200000
20.63000	26.48198	29.47300	-4.266943	3.920000
20.79000	26.58331	30.76800	-3.381990	2.520000
20.96000	26.69097	30.62000	-3.084659	4.200000
21.19000	26.83663	31.74200	-1.423619	6.440000
21.29000	26.89996	31.27200	-2.820670	2.800000
21.46000	27.00762	31.53500	-2.199583	4.200000
21.63000	27.11528	32.41900	-1.199259	4.760000
21.71000	27.16594	32.47800	-0.7825233	6.160000
21.79000	27.21661	32.11400	-1.134799	6.160000
21.88000	27.27360	31.62700	-1.534922	6.440000
21.96000	27.32427	31.10700	-2.002742	6.580000
22.17000	27.45726	30.97300	-2.382105	5.600000
22.29000	27.53326	31.24800	-1.972901	6.160000
22.46000	27.64092	31.67800	-1.485563	6.440000
22.63000	27.74858	30.74900	-2.384699	6.440000
22.71000	27.79924	31.50500	-1.724180	6.160000
22.88000	27.90690	30.58600	-2.828360	5.320000
22.96000	27.95757	30.74400	-2.964715	4.200000
23.13000	28.06523	30.53200	-2.808837	5.600000
23.29000	28.16656	31.28700	-3.028763	1.960000
23.46000	28.27422	31.56900	-1.805387	5.600000
23.63000	28.38188	31.13600	-2.152668	5.880000
23.71000	28.43254	33.17300	3.2455627E-02	6.720000
23.88000	28.54020	31.45000	-1.848804	5.880000
24.08000	28.66686	30.58800	-3.891504	1.260000
24.29000	28.79986	29.93400	-4.561129	1.120000
24.46000	28.90752	30.03800	-4.423102	1.260000
24.63000	29.01518	30.27600	-4.230585	1.120000
24.79000	29.11651	30.84300	-3.682574	1.120000
24.88000	29.17350	29.98700	-4.472396	1.260000
24.96000	29.22417	29.30000	-5.136412	1.260000
25.13000	29.33183	30.37300	-4.136833	1.120000
25.29000	29.43316	31.04500	-3.449792	1.260000
25.52000	29.57900	30.97300	-3.556928	1.120000
25.84000	29.78147	30.00900	-4.488640	1.120000
26.09000	29.93980	29.70900	-4.741096	1.260000
27.10000	30.57943	30.61100	-3.906804	1.120000
27.31000	30.71242	30.45800	-4.054679	1.120000
40.13000	34.76433	29.38800	-5.088842	1.120000
40.21000	34.81499	30.80000	-3.724134	1.120000
40.30000	34.87199	30.26100	-4.245081	1.120000
40.42000	34.94799	28.88800	-5.572095	1.120000
40.59000	35.05565	30.80200	-3.722201	1.120000
40.75000	35.15697	27.85200	-6.573398	1.120000
40.92000	35.26464	28.10700	-6.326938	1.120000
41.09000	35.37230	27.89100	-6.535703	1.120000

15.36288	4.76	0.971829	-14.00705	-26.06740
15.49394	5.60	1.026992	-14.11795	-25.48307
15.77243	5.32	1.008604	-14.35360	-25.67770
16.00997	6.16	1.063767	-14.55459	-25.09377
16.31304	5.88	1.045380	-14.78137	-25.28843
16.45229	5.88	1.045380	-14.58244	-25.28843
16.59153	5.32	1.008604	-14.38353	-25.67770
16.65706	5.04	0.990217	-14.28991	-25.87256
17.00108	5.60	1.026992	-13.80097	-25.48307
17.13214	5.88	1.045380	-13.91893	-25.28843
17.20586	6.72	1.100542	-13.98527	-24.70501
17.30415	6.44	1.082155	-14.07373	-24.89934
17.41063	5.32	1.008604	-14.16957	-25.67770
17.54169	5.60	1.026992	-14.28752	-25.48307
17.61541	5.32	1.008604	-14.35387	-25.67770
17.68094	5.32	1.008604	-14.41285	-25.67770
17.82018	4.48	0.953442	-14.53816	-26.26246
17.95124	3.92	0.916666	-14.81458	-26.65266
18.09049	3.08	0.861504	-14.48735	-27.23862
18.22973	1.68	0.769566	-14.16013	-28.21691
18.29974	6.44	1.082155	-13.99561	-24.89934
18.43274	6.72	1.100542	-13.68306	-24.70501
18.56573	6.72	1.100542	-13.37054	-24.70501
18.66706	6.72	1.100542	-13.13241	-24.70501
18.77472	6.72	1.100542	-12.87941	-24.70501
18.88238	6.72	1.100542	-12.62641	-24.70501
18.93304	6.16	1.063767	-12.31485	-25.09377
19.06604	6.16	1.063767	-12.38467	-25.09377
19.16736	4.62	0.962635	-12.43786	-26.16493
19.30036	3.92	0.916666	-12.50769	-26.65266
19.40802	6.44	1.082155	-12.56421	-24.89934
19.51568	6.72	1.100542	-12.62073	-24.70501
19.67400	4.48	0.953442	-12.70385	-26.26246
19.72467	5.32	1.008604	-12.73045	-25.67770
19.83233	6.72	1.100542	-12.78697	-24.70501
19.93366	6.44	1.082155	-12.84017	-24.89934
20.04132	6.72	1.100542	-12.89669	-24.70501
20.14898	3.92	0.916666	-12.95321	-26.65266
20.35797	6.72	1.100542	-13.06293	-24.70501
20.46563	3.64	0.898279	-13.11946	-26.84791
20.56696	1.96	0.787953	-13.17265	-28.02104
20.67462	1.12	0.732790	-13.22918	-28.60883
20.78228	1.96	0.787953	-13.28570	-28.02104
20.83294	1.96	0.787953	-13.31229	-28.02104
20.94060	2.24	0.806341	-13.35099	-27.82528
20.99127	1.40	0.751178	-13.39280	-28.41287
21.09893	4.20	0.935054	-13.48162	-26.45751
21.20026	3.08	0.861504	-13.56521	-27.23862
21.30792	1.12	0.732790	-13.65403	-28.60883
21.41558	1.68	0.769566	-13.74285	-28.21691
21.49158	2.24	0.806341	-13.80555	-27.82528
21.59924	1.96	0.787953	-13.89437	-28.02104
21.73223	1.68	0.769566	-14.00409	-28.21691
21.83356	5.88	1.045380	-14.08769	-25.28843
21.94122	6.72	1.100542	-14.17651	-24.70501
22.04888	5.60	1.026992	-14.26533	-25.48307
22.30853	5.04	0.990217	-14.47954	-25.87256
22.36553	4.76	0.971829	-14.52656	-26.06740
22.41619	5.88	1.045380	-14.56836	-25.28843
22.49219	4.76	0.971829	-14.63106	-26.06740
22.57452	2.52	0.824728	-14.69898	-27.62963
22.62518	3.64	0.898279	-14.74077	-26.84791
22.68218	4.48	0.953442	-14.78780	-26.26246
22.73284	4.76	0.971829	-14.82959	-26.06740
22.96716	5.04	0.990217	-15.05493	-25.87256
23.04949	5.60	1.026992	-15.04258	-25.48307
23.10016	6.44	1.082155	-15.03498	-24.89934
23.20782	5.88	1.045380	-15.01883	-25.28843
23.31548	6.72	1.100542	-15.00268	-24.70501
23.36614	6.44	1.082155	-14.99508	-24.89934
23.48647	6.44	1.082155	-14.97703	-24.89934
23.57513	6.72	1.100542	-14.96373	-24.70501
23.63213	6.02	1.054573	-14.95518	-25.19110
23.73346	6.16	1.063767	-14.93998	-25.09377
23.84112	5.88	1.045380	-14.92383	-25.28843
23.94878	5.60	1.026992	-14.90768	-25.48307

24.05011	6.44	1.082155	-14.89248	-24.89934
24.20843	6.44	1.082155	-14.86874	-24.89934
24.26543	6.16	1.063767	-14.86019	-25.09377
24.36676	6.16	1.063767	-14.84499	-25.09377
24.47442	5.32	1.008604	-14.82884	-25.67770
24.58208	4.76	0.971829	-14.81269	-26.06740
24.63274	3.92	0.916666	-14.80509	-26.65266
24.74040	4.48	0.953442	-14.78894	-26.26246
24.84173	3.92	0.916666	-14.77374	-26.65266
24.89873	4.48	0.953442	-14.76519	-26.26246
25.00006	4.48	0.953442	-14.74999	-26.26246
25.10772	5.04	0.990217	-14.73384	-25.87256
25.15838	5.32	1.008604	-14.72624	-25.67770
25.21538	6.16	1.063767	-14.71769	-25.09377
25.26604	6.16	1.063767	-14.71009	-25.09377
25.37370	3.92	0.916666	-14.69395	-26.65266
25.42437	5.88	1.045380	-14.68635	-25.28843
25.53203	6.30	1.072961	-14.67020	-24.99656
25.63336	5.88	1.045380	-14.65500	-25.28843
25.74102	6.44	1.082155	-14.63885	-24.89934
25.79168	6.44	1.082155	-14.63125	-24.89934
25.84868	6.44	1.082155	-14.62270	-24.89934
25.89934	6.44	1.082155	-14.61510	-24.89934
26.00700	3.92	0.916666	-14.59969	-26.65266
26.05767	4.20	0.935054	-14.59744	-26.45751
26.16533	4.48	0.953442	-14.59265	-26.26246
26.26666	4.48	0.953442	-14.58815	-26.26246
26.37432	4.48	0.953442	-14.58336	-26.26246
26.42498	4.20	0.935054	-14.58111	-26.45751
26.48198	3.92	0.916666	-14.57858	-26.65266
26.58331	2.52	0.824728	-14.57408	-27.62963
26.69097	4.20	0.935054	-14.56929	-26.45751
26.83663	6.44	1.082155	-14.56282	-24.89934
26.89996	2.80	0.843116	-14.56000	-27.43407
27.00762	4.20	0.935054	-14.55522	-26.45751
27.11528	4.76	0.971829	-14.55043	-26.06740
27.16594	6.16	1.063767	-14.54818	-25.09377
27.21661	6.16	1.063767	-14.54593	-25.09377
27.27360	6.44	1.082155	-14.54340	-24.89934
27.32427	6.58	1.091349	-14.54114	-24.80211
27.45726	5.60	1.026992	-14.53523	-25.48307
27.53326	6.16	1.063767	-14.53186	-25.09377
27.64092	6.44	1.082155	-14.52707	-24.89934
27.74858	6.44	1.082155	-14.52229	-24.89934
27.79924	6.16	1.063767	-14.52003	-25.09377
27.90690	5.32	1.008604	-14.51525	-25.67770
27.95757	4.20	0.935054	-14.51300	-26.45751
28.06523	5.60	1.026992	-14.50821	-25.48307
28.16656	1.96	0.787953	-14.50371	-28.02104
28.27422	5.60	1.026992	-14.49892	-25.48307
28.38188	5.88	1.045380	-14.49414	-25.28843
28.43254	6.72	1.100542	-14.49189	-24.70501
28.54020	5.88	1.045380	-14.48710	-25.28843
28.66686	1.26	0.741984	-14.48147	-28.51080
28.79986	1.12	0.732790	-14.47556	-28.60883
28.90752	1.26	0.741984	-14.47078	-28.51080
29.01518	1.12	0.732790	-14.46599	-28.60883
29.11651	1.12	0.732790	-14.46149	-28.60883
29.17350	1.26	0.741984	-14.45896	-28.51080
29.22417	1.26	0.741984	-14.45670	-28.51080
29.33183	1.12	0.732790	-14.45192	-28.60883
29.43316	1.26	0.741984	-14.44742	-28.51080
29.57900	1.12	0.732790	-14.44093	-28.60883
29.78147	1.12	0.732790	-14.43194	-28.60883
29.93980	1.26	0.741984	-14.42490	-28.51080
30.57943	1.12	0.732790	-14.39647	-28.60883
30.71242	1.12	0.732790	-14.39056	-28.60883
34.76433	1.12	0.732790	-14.21047	-28.60883
34.81499	1.12	0.732790	-14.20822	-28.60883
34.87199	1.12	0.732790	-14.20569	-28.60883
34.94799	1.12	0.732790	-14.17890	-28.60883
35.05565	1.12	0.732790	-14.22258	-28.60883
35.15697	1.12	0.732790	-14.26369	-28.60883
35.26464	1.12	0.732790	-14.30737	-28.60883
35.37230	1.12	0.732790	-14.35105	-28.60883

Appendix R

Numerical Modeling Results for the Paleo-Owens River System

```
***********************************************************
*                                                         *
*       MODEL GENERATED SEARLES LAKE AREA OUTPUT          *
*                                                         *
***********************************************************
```

AGE(Ka)	LAKE AREA (billion m2)	AGE(Ka)	LAKE AREA (billion m2)
1168.84094	1.33629	1132.60901	1.86377
1168.11597	1.11342	1131.88403	1.89080
1167.39099	1.24004	1131.16003	1.91960
1166.66699	1.35668	1130.43506	1.94906
1165.94202	1.47638	1129.70996	1.96312
1165.21704	1.58879	1128.98596	1.97660
1164.49304	1.72907	1128.26099	1.99088
1163.76794	1.76433	1127.53601	2.01913
1163.04297	1.80910	1126.81201	2.04763
1162.31897	1.84602	1126.08704	2.04185
1161.59399	1.84787	1125.36206	2.03095
1160.87000	1.81372	1124.63794	1.98461
1160.14502	1.77625	1123.91296	1.94172
1159.42004	1.73898	1123.18799	1.93057
1158.69604	1.69507	1122.46399	1.91882
1157.97095	1.37545	1121.73901	1.89014
1157.24597	1.34998	1121.01501	1.85855
1156.52197	1.32451	1120.29004	1.81665
1155.79700	1.29903	1119.56494	1.77540
1155.07300	1.27357	1118.84094	1.73411
1154.34802	1.24809	1118.11597	1.64626
1153.62305	1.22262	1117.39099	1.38935
1152.89905	1.19715	1116.66699	1.22713
1152.17395	1.17168	1115.94202	1.06420
1151.44897	1.14621	1115.21704	0.93491
1150.72498	1.12073	1114.49304	0.90058
1150.00000	1.09526	1113.76794	0.86857
1149.27502	1.06979	1113.04395	0.86638
1148.55103	1.04432	1112.31897	0.86419
1147.82605	1.01885	1111.59399	0.86200
1147.10205	0.99338	1110.87000	0.85981
1146.37695	0.96790	1110.14502	0.85762
1145.65198	0.94243	1109.42004	0.91429
1144.92798	0.91696	1108.69604	0.96524
1144.20300	0.89149	1107.97095	1.01484
1143.47803	0.86602	1107.24597	1.06738
1142.75403	1.00725	1106.52197	1.10129
1142.02905	1.09714	1105.79700	1.12954
1141.30396	1.17876	1105.07300	1.16768
1140.57996	1.30084	1104.34802	1.21278
1139.85498	1.40805	1103.62305	1.27826
1139.13000	1.51011	1102.89905	1.34569
1138.40601	1.60966	1102.17395	1.40653
1137.68103	1.70210	1101.44897	1.46873
1136.95703	1.73140	1100.72498	1.53191
1136.23206	1.74756	1100.00000	1.59502
1135.50696	1.76384	1099.62500	1.72020
1134.78296	1.78757	1099.16394	1.73712
1134.05798	1.81228	1098.70300	1.75858
1133.33301	1.83723	1098.24097	1.78132

1097.78003	1.80370
1097.31897	1.82158
1096.85803	1.80015
1096.39600	1.77209
1095.93506	1.74437
1095.47400	1.71622
1095.01196	1.68605
1094.55103	1.60141
1094.08997	1.54084
1093.62805	1.52631
1093.16699	1.51598
1092.70605	1.53141
1092.24500	1.55480
1091.78296	1.62945
1091.32202	1.70479
1090.86096	1.71233
1090.39905	1.71979
1089.93799	1.72982
1089.47705	1.73846
1089.01501	1.73573
1088.55396	1.73086
1088.09302	1.71899
1087.63098	1.70446
1087.17004	1.67912
1086.70898	1.60974
1086.24805	1.58369
1085.78601	1.59864
1085.32495	1.60597
1084.86401	1.49353
1084.40198	1.45817
1083.94104	1.43743
1083.47998	1.41716
1083.01904	1.39712
1082.55701	1.37779
1082.09595	1.35962
1081.63501	1.33834
1081.17297	1.32770
1080.71204	1.31885
1080.25098	1.31028
1079.78894	1.30035
1079.32800	1.29096
1078.86694	1.29874
1078.40601	1.31140
1077.94397	1.33176
1077.48303	1.35583
1077.02197	1.38837
1076.56006	1.41708
1076.09900	1.46665
1075.63794	1.52784
1075.17603	1.54831
1074.71497	1.55659
1074.25403	1.55268
1073.79199	1.54770
1073.33105	1.53631
1072.87000	1.52561
1072.40906	1.52729
1071.94702	1.55646
1071.48596	1.62198
1071.02502	1.70312
1070.56299	1.73465
1070.I0205	1.76690
1069.64099	1.80025
1069.18005	1.83324
1068.71802	1.85037
1068.25696	1.83661
1067.79602	1.80893
1067.33398	1.77517
1066.87305	1.74375
1066.41199	1.71782
1065.94995	1.70836
1065.48901	1.69910
1065.02795	1.68953
1064.56702	1.68531
1064.10498	1.68314
1063.64404	1.67882
1063.18298	1.69397
1062.72095	1.69700
1062.26001	1.69413
1061.79895	1.69157
1061.33704	1.68949
1060.87598	1.68518
1060.41504	1.64827
1059.95300	1.59879
1059.49194	1.54793
1059.03101	1.48379
1058.56995	1.42375
1058.10803	1.39255
1057.64697	1.36781
1057.18604	1.34973
1056.72400	1.32741
1056.26294	1.30402
1055.80200	1.29162
1055.33997	1.28901
1054.87903	1.28914
1054.41797	1.28428
1053.95703	1.29723
1053.49500	1.33106
1053.03406	1.34875
1052.57300	1.36007
1052.11096	1.36544
1051.65002	1.35599
1051.18896	1.35913
1050.72803	1.37353
1050.26599	1.40155
1049.80505	1.43123
1049.34399	1.44910
1048.88196	1.45618
1048.42102	1.40219
1047.95996	1.32817
1047.49805	1.26086
1047.03699	1.21114
1046.57605	1.17511
1046.11499	1.13325
1045.65295	1.08860
1045.19202	1.08134
1044.73096	1.12302
1044.26904	1.20202
1043.80798	1.31151
1043.34705	1.43494
1042.88501	1.57216

1042.42395	1.66596	1014.74597	1.61127
1041.96301	1.65848	1014.28497	1.58038
1041.50098	1.63924	1013.82300	1.57223
1041.04004	1.62241	1013.36200	1.57405
1040.57898	1.60246	1012.90100	1.56871
1040.11804	1.59620	1012.44000	1.56755
1039.65601	1.59201	1011.97803	1.63233
1039.19495	1.59152	1011.51703	1.70254
1038.73401	1.58971	1011.05603	1.71172
1038.27197	1.58754	1010.59399	1.71658
1037.81104	1.58728	1010.13300	1.72688
1037.34998	1.57939	1009.67200	1.73888
1036.88904	1.57118	1009.21100	1.74905
1036.42700	1.51872	1008.74902	1.75854
1035.96594	1.46043	1008.28802	1.76247
1035.50500	1.47912	1007.82703	1.76499
1035.04297	1.49791	1007.36499	1.76161
1034.58203	1.51730	1006.90399	1.71557
1034.12097	1.53404	1006.44299	1.67164
1033.65906	1.53785	1005.98102	1.53929
1033.19800	1.54470	1005.52002	1.40233
1032.73706	1.58045	1005.05902	1.28553
1032.27600	1.63064	1004.59698	1.18217
1031.81396	1.62685	1004.13599	1.10242
1031.35303	1.59953	1003.67499	1.07432
1030.89197	1.56530	1003.21399	1.04046
1030.43005	1.51958	1002.75201	1.00681
1029.96899	1.48416	1002.29102	0.97859
1029.50806	1.48819	1001.83002	1.00066
1029.04602	1.51556	1001.36798	1.04304
1028.58496	1.51781	1000.90698	1.07587
1028.12402	1.50617	1000.44598	1.09841
1027.66199	1.48807	999.98450	1.11775
1027.20105	1.47141	999.52319	1.14409
1026.73999	1.45917	999.06189	1.17398
1026.27905	1.44796	998.60059	1.20492
1025.81702	1.47422	998.13928	1.26150
1025.35596	1.50024	997.67798	1.50268
1024.89502	1.51289	997.21667	1.73507
1024.43298	1.52216	996.75537	1.81192
1023.97198	1.53367	996.29413	1.88811
1023.51099	1.54017	995.83282	1.94631
1023.04901	1.53927	995.37152	1.85934
1022.58801	1.55186	994.91022	1.77254
1022.12701	1.59966	994.44891	1.68510
1021.66602	1.65435	993.98761	1.46104
1021.20398	1.69694	993.52631	1.24670
1020.74298	1.71030	993.06500	1.22645
1020.28198	1.72350	992.60370	1.24749
1019.82001	1.73970	992.14240	1.26978
1019.35901	1.75361	991.68109	1.29044
1018.89801	1.72521	991.21979	1.33901
1018.43701	1.70048	990.75848	1.42991
1017.97498	1.69051	990.29718	1.54188
1017.51398	1.63969	989.83588	1.56616
1017.05298	1.56971	989.37457	1.56139
1016.59100	1.56462	988.91333	1.59855
1016.13000	1.58246	988.45203	1.66932
1015.66901	1.60433	987.99072	1.58210
1015.20697	1.62662	987.52942	1.19765

987.06812	0.93412	947.00000	1.26354
986.60681	0.92927	946.00000	1.44588
986.14551	0.92479	945.00000	1.57102
985.68420	0.92205	944.00000	1.45130
985.22290	0.91756	943.00000	1.29193
984.76160	0.92822	942.00000	1.13085
984.30029	0.94889	941.00000	0.97996
983.83899	0.97007	940.00000	0.95362
983.37769	0.99822	939.00000	0.96636
982.91638	1.02455	938.00000	0.98091
982.45508	1.05042	937.00000	0.99508
981.99377	1.07378	936.00000	0.99727
981.53247	1.09417	935.00000	0.98628
981.07123	1.09502	934.00000	0.97528
980.60992	1.24623	933.00000	0.96429
980.14862	1.51715	932.00000	0.95330
979.68732	1.64379	931.00000	0.94231
979.22601	1.69929	930.00000	0.93132
978.76471	1.72924	929.00000	1.40920
978.30341	1.75985	928.00000	1.73243
977.84210	1.79131	927.00000	1.83188
977.38080	1.81914	926.00000	1.82874
976.91949	1.84576	925.00000	1.82009
976.45819	1.85745	924.00000	1.79719
975.99689	1.88355	923.00000	1.76645
975.53558	1.91675	922.00000	1.73568
975.07428	1.95061	921.00000	1.70460
974.61298	1.98415	920.00000	1.67000
974.15167	2.00762	919.00000	1.56508
973.69043	2.03042	918.00000	1.46774
973.22913	2.04900	917.00000	1.38297
972.76782	2.06414	916.00000	1.30828
972.30652	2.09175	915.00000	1.23690
971.84521	2.11975	914.00000	1.17682
971.38391	2.16063	913.00000	1.16087
970.92261	2.22236	912.00000	1.15645
970.46130	1.59349	911.00000	1.15198
970.00000	1.13807	910.00000	1.14808
969.00000	0.92388	909.00000	1.15233
968.00000	0.92273	908.00000	1.15952
967.00000	0.92159	907.00000	1.16554
966.00000	0.92045	906.00000	1.17180
965.00000	0.91931	905.00000	1.16678
964.00000	0.91817	904.00000	0.90957
963.00000	0.91703	903.00000	0.90378
962.00000	0.91589	902.00000	0.89800
961.00000	0.91475	901.00000	0.89221
960.00000	0.91361	900.00000	0.88642
959.00000	0.91247	899.00000	1.35450
958.00000	0.91133	898.00000	1.81868
957.00000	0.91019	897.00000	1.85526
956.00000	0.90905	896.00000	1.80539
955.00000	0.90791	895.00000	1.76252
954.00000	0.90677	894.00000	1.71689
953.00000	0.90563	893.00000	1.61285
952.00000	0.98214	892.00000	1.41078
951.00000	0.99682	891.00000	1.27513
950.00000	1.00844	890.00000	1.15950
949.00000	1.06306	889.00000	1.17448
948.00000	1.13018	888.00000	1.27626

887.00000	1.37561	827.00000	1.98407
886.00000	1.49459	826.00000	2.00731
885.00000	1.57395	825.00000	2.02979
884.00000	1.50371	824.00000	2.04802
883.00000	1.43039	823.00000	2.07005
882.00000	1.36695	822.00000	2.09940
881.00000	1.33352	821.00000	2.12809
880.00000	1.48787	820.00000	2.15721
879.00000	1.65388	819.00000	2.19895
878.00000	1.73625	818.00000	2.25040
877.00000	1.76231	817.00000	2.29968
876.00000	1.78755	816.00000	2.35017
875.00000	1.81369	815.00000	2.39808
874.00000	1.83111	814.00000	2.43207
873.00000	1.79422	813.00000	2.46238
872.00000	1.77004	812.00000	2.49292
871.00000	1.74603	811.00000	2.52982
870.00000	1.72289	810.00000	2.56530
869.00000	1.69926	809.00000	2.49338
868.00000	1.66380	808.00000	2.42013
867.00000	1.62684	807.00000	2.30098
866.00000	1.61441	806.00000	2.16467
865.00000	1.60714	805.00000	1.99487
864.00000	1.60065	804.00000	1.82331
863.00000	1.59303	803.00000	1.73135
862.00000	1.60244	802.00000	1.68372
861.00000	1.61303	801.00000	1.53054
860.00000	1.60825	800.00000	1.30196
859.00000	1.59609	799.00000	1.05045
858.00000	1.58499	798.00000	1.02527
857.00000	1.57503	797.00000	1.00009
856.00000	1.56637	796.00000	0.97491
855.00000	1.56386	795.00000	0.94973
854.00000	1.50187	794.00000	0.92454
853.00000	1.41784	793.00000	0.89936
852.00000	1.34017	792.00000	0.87418
851.00000	1.25857	791.00000	0.84900
850.00000	1.18484	790.00000	1.11049
849.00000	1.14152	789.00000	1.36912
848.00000	1.14469	788.00000	1.71834
847.00000	1.14745	787.00000	1.78420
846.00000	1.15674	786.00000	1.81425
845.00000	1.16714	785.00000	1.83082
844.00000	1.19334	784.00000	1.84608
843.00000	1.21499	783.00000	1.87949
842.00000	1.23024	782.00000	1.83710
841.00000	1.24739	781.00000	1.77123
840.00000	1.27341	780.00000	1.70462
839.00000	1.40047	779.00000	1.53049
838.00000	1.55630	778.00000	1.22978
837.00000	1.70948	777.00000	0.85884
836.00000	1.74849	776.00000	0.86052
835.00000	1.78635	775.00000	0.86220
834.00000	1.78654	774.00000	0.86388
833.00000	1.78211	773.00000	0.86556
832.00000	1.76568	772.00000	0.86724
831.00000	1.77870	771.00000	0.86892
830.00000	1.83299	770.00000	0.87060
829.00000	1.88981	769.00000	0.87228
828.00000	1.94574	768.00000	0.87396

767.00000	0.87564	707.00000	1.77153
766.00000	0.87732	706.00000	1.81840
765.00000	0.87900	705.00000	1.87154
764.00000	0.88068	704.00000	1.92929
763.00000	0.88235	703.00000	1.94582
762.00000	0.88403	702.00000	1.88239
761.00000	0.88571	701.00000	1.84266
760.00000	0.88739	700.00000	1.80336
759.00000	0.88907	699.00000	1.75728
758.00000	0.89075	698.00000	1.71094
757.00000	0.89243	697.00000	1.64557
756.00000	0.89411	696.00000	1.53228
755.00000	0.89579	695.00000	1.43457
754.00000	0.89747	694.00000	1.37887
753.00000	0.89915	693.00000	1.29738
752.00000	0.90083	692.00000	1.12273
751.00000	1.12280	691.00000	1.13099
750.00000	1.31568	690.00000	1.16473
749.00000	1.54697	689.00000	1.20102
748.00000	1.63828	688.00000	1.23180
747.00000	1.68847	687.00000	1.25267
746.00000	1.70816	686.00000	1.27294
745.00000	1.71135	685.00000	1.29388
744.00000	1.71535	684.00000	1.33001
743.00000	1.73794	683.00000	1.36600
742.00000	1.75848	682.00000	1.40343
741.00000	1.75966	681.00000	1.43428
740.00000	1.76019	680.00000	1.46482
739.00000	1.75305	679.00000	1.13159
738.00000	1.74813	678.00000	1.12271
737.00000	1.74417	677.00000	1.08826
736.00000	1.73172	676.00000	1.04087
735.00000	1.71401	675.00000	1.01677
734.00000	1.69595	674.00000	1.00056
733.00000	1.69836	673.00000	0.98393
732.00000	1.70680	672.00000	0.97599
731.00000	1.70664	671.00000	0.96806
730.00000	1.70226	670.00000	0.96012
729.00000	1.69738	669.00000	0.95218
728.00000	1.69444	668.00000	0.94425
727.00000	1.66136	667.00000	0.93631
726.00000	1.65326	666.00000	1.10303
725.00000	1.67088	665.00000	1.18662
724.00000	1.70003	664.00000	1.33980
723.00000	1.66686	663.00000	1.49461
722.00000	1.57174	662.00000	1.62206
721.00000	1.46191	661.00000	1.71256
720.00000	1.37848	660.00000	1.29361
719.00000	1.35292	659.00000	1.35799
718.00000	1.29469	658.00000	1.67275
717.00000	1.19980	657.00000	1.75003
716.00000	1.04477	656.00000	1.78365
715.00000	0.89955	655.00000	1.84501
714.00000	0.89835	654.00000	1.94521
713.00000	0.89716	653.00000	1.69936
712.00000	1.03085	652.00000	1.24169
711.00000	1.12561	651.00000	1.23675
710.00000	1.28335	650.00000	1.26276
709.00000	1.51671	649.00000	1.44959
708.00000	1.70789	648.00000	1.37401

647.00000	1.15691
646.00000	1.19853
645.00000	1.24325
644.00000	1.27055
643.00000	1.17685
642.00000	1.35931
641.00000	1.58673
640.00000	1.29367
639.00000	1.00624
638.00000	1.17818
637.00000	1.53506
636.00000	1.69689
635.00000	1.72058
634.00000	1.74802
633.00000	1.76694
632.00000	1.79831
631.00000	1.82899
630.00000	1.83799
629.00000	1.85924
628.00000	1.92408
627.00000	1.89173
626.00000	1.70687
625.00000	1.56838
624.00000	1.56432
623.00000	1.70350
622.00000	1.82332
621.00000	1.86723
620.00000	2.01904
619.00000	2.27848
618.00000	2.06315
617.00000	1.63473
616.00000	1.47541
615.00000	1.79871
614.00000	1.82217
613.00000	1.82553
612.00000	1.80383
611.00000	1.79677
610.00000	1.91387
609.00000	2.00226
608.00000	1.85137
607.00000	1.80849
606.00000	1.83644
605.00000	1.77979
604.00000	1.67567
603.00000	1.54588
602.00000	1.50387
601.00000	1.64752
600.00000	1.49404
599.00000	0.91035
598.00000	0.91170
597.00000	0.91306
596.00000	0.91441
595.00000	0.91576
594.00000	0.91711
593.00000	0.91846
592.00000	0.91982
591.00000	0.92117
590.00000	0.92252
589.00000	0.92387
588.00000	0.92523
587.00000	0.92658
586.00000	0.92793
585.00000	0.92928
584.00000	0.93064
583.00000	0.93199
582.00000	0.93334
581.00000	0.93469
580.00000	0.93605
579.00000	0.93740
578.00000	0.93875
577.00000	0.94010
576.00000	0.94146
575.00000	0.94281
574.00000	0.94416
573.00000	0.94551
572.00000	0.94687
571.00000	0.94822
570.00000	0.94957
569.00000	0.95092
568.00000	0.95228
567.00000	0.95363
566.00000	0.95498
565.00000	0.95633
564.00000	0.95768
563.00000	0.95904
562.00000	0.96039
561.00000	0.96174
560.00000	0.96309
559.00000	0.96445
558.00000	0.96580
557.00000	0.96715
556.00000	0.96850
555.00000	0.96986
554.00000	0.97121
553.00000	0.97256
552.00000	0.97391
551.00000	0.97527
550.00000	0.97662
549.00000	0.97797
548.00000	0.97932
547.00000	0.98068
546.00000	0.98203
545.00000	0.98338
544.00000	0.98473
543.00000	0.98609
542.00000	0.98744
541.00000	0.98879
540.00000	0.99014
539.00000	0.99149
538.00000	0.99285
537.00000	0.99420
536.00000	0.99555
535.00000	0.99690
534.00000	0.99826
533.00000	0.99961
532.00000	1.00096
531.00000	1.00231
530.00000	1.00367
529.00000	1.00502
528.00000	1.00637

527.00000	1.00772	467.00000	1.10179
526.00000	1.00908	466.00000	1.09768
525.00000	1.01043	465.00000	1.09650
524.00000	1.01178	464.00000	1.09632
523.00000	1.01313	463.00000	1.09630
522.00000	1.01449	462.00000	1.09629
521.00000	1.01584	461.00000	1.09627
520.00000	1.01719	460.00000	1.09626
519.00000	1.01854	459.00000	1.09624
518.00000	1.01990	458.00000	1.09623
517.00000	1.02125	457.00000	1.09621
516.00000	1.02260	456.00000	1.09620
515.00000	1.02395	455.00000	1.09618
514.00000	1.02531	454.00000	1.09617
513.00000	1.02666	453.00000	1.09615
512.00000	1.02801	452.00000	1.09614
511.00000	1.02936	451.00000	1.09612
510.00000	1.03071	450.00000	1.09611
509.00000	1.03207	449.00000	1.09609
508.00000	1.03342	448.00000	1.09608
507.00000	1.03477	447.00000	1.16518
506.00000	1.03612	446.00000	1.29316
505.00000	1.03748	445.00000	1.62534
504.00000	1.03883	444.00000	1.84369
503.00000	1.04018	443.00000	1.83643
502.00000	1.04153	442.00000	1.82977
501.00000	1.04289	441.00000	1.81811
500.00000	1.04424	440.00000	1.80030
499.00000	1.04559	439.00000	1.79294
498.00000	1.04694	438.00000	1.79481
497.00000	1.04830	437.00000	1.78433
496.00000	1.04965	436.00000	1.77480
495.00000	1.05100	435.00000	1.77615
494.00000	1.05235	434.00000	1.80999
493.00000	1.05371	433.00000	1.87249
492.00000	1.05506	432.00000	1.93753
491.00000	1.05641	431.00000	1.97924
490.00000	1.05776	430.00000	2.00826
489.00000	1.05912	429.00000	2.02566
488.00000	1.06047	428.00000	2.04089
487.00000	1.06182	427.00000	2.34558
486.00000	1.06317	426.00000	2.31326
485.00000	1.06453	425.00000	2.11393
484.00000	1.06588	424.00000	1.91368
483.00000	1.06723	423.00000	1.87254
482.00000	1.06858	422.00000	1.87201
481.00000	1.06993	421.00000	1.83441
480.00000	1.07129	420.00000	1.75668
479.00000	1.07264	419.00000	1.68077
478.00000	1.07399	418.00000	1.46513
477.00000	1.07534	417.00000	1.20263
476.00000	1.07670	416.00000	0.86154
475.00000	1.07805	415.00000	0.86157
474.00000	1.07940	414.00000	0.86159
473.00000	1.08075	413.00000	0.86162
472.00000	1.10004	412.00000	0.86164
471.00000	1.10067	411.00000	0.86167
470.00000	1.10604	410.00000	0.86169
469.00000	1.11369	409.00000	0.86172
468.00000	1.10742	408.00000	0.86174

407.00000	0.86177	347.00000	0.86700
406.00000	0.86179	346.00000	0.86712
405.00000	0.86182	345.00000	0.86725
404.00000	0.86184	344.00000	0.86738
403.00000	0.86187	343.00000	0.86750
402.00000	0.86189	342.00000	0.86762
401.00000	0.86192	341.00000	0.86775
400.00000	0.86195	340.00000	0.86787
399.00000	0.86197	339.00000	0.86800
398.00000	0.86200	338.00000	0.86813
397.00000	0.86202	337.00000	0.86825
396.00000	0.86205	336.00000	0.86838
395.00000	0.86207	335.00000	0.86850
394.00000	0.86210	334.00000	0.86862
393.00000	0.86212	333.00000	0.86875
392.00000	0.86215	332.00000	0.86888
391.00000	0.86217	331.00000	0.86900
390.00000	0.86220	330.00000	0.86913
389.00000	0.86222	329.00000	0.86925
388.00000	0.86225	328.00000	0.86937
387.00000	0.86227	327.00000	0.86950
386.00000	0.86230	326.00000	0.86962
385.00000	0.86232	325.00000	0.86975
384.00000	0.86235	324.00000	0.86988
383.00000	0.86237	323.00000	0.87000
382.00000	0.86240	322.00000	0.87012
381.00000	0.86242	321.00000	0.87025
380.00000	0.86245	320.00000	0.87037
379.00000	0.86247	319.00000	0.87050
378.00000	0.86250	318.00000	0.87063
377.00000	0.86252	317.00000	0.87075
376.00000	0.86255	316.00000	0.87088
375.00000	1.12677	315.00000	0.87100
374.00000	1.39916	314.00000	0.87112
373.00000	1.73438	313.00000	0.87125
372.00000	1.80471	312.00000	0.87138
371.00000	2.12322	311.00000	0.87150
370.00000	2.04026	310.00000	0.87163
369.00000	2.02203	309.00000	0.87175
368.00000	2.00456	308.00000	0.87187
367.00000	1.96763	306.10001	0.87200
366.00000	1.96933	304.29999	0.87900
365.00000	1.98608	302.60001	0.88700
364.00000	1.99398	300.89999	0.89500
363.00000	1.97190	299.10001	0.89700
362.00000	1.93707	297.39999	0.89700
361.00000	1.82026	295.70001	0.91000
360.00000	1.71411	293.89999	0.91800
359.00000	1.48524	292.20001	0.92300
358.00000	1.15557	290.39999	0.93600
357.00000	0.98073	288.70001	0.96200
356.00000	0.91293	287.00000	1.00000
355.00000	0.86600	285.20001	1.25600
354.00000	0.86612	283.50000	1.17900
353.00000	0.86625	281.70001	1.07700
352.00000	0.86638	280.00000	1.05900
351.00000	0.86650	278.29999	1.04600
350.00000	0.86663	276.50000	1.02600
349.00000	0.86675	274.79999	1.01300
348.00000	0.86687	273.00000	1.00000

271.29999	1.00000	170.00000	1.39557
269.60001	1.00000	169.00000	1.36081
267.79999	1.00300	168.00000	1.41378
266.10001	1.00500	167.00000	1.27255
264.29999	1.01300	166.00000	1.28466
262.60001	1.02100	165.20000	1.20500
260.89999	1.02100	163.50000	1.20500
259.10001	1.02300	161.70000	1.25600
257.39999	1.02300	160.00000	1.35900
255.70000	1.02800	158.30000	1.38500
253.89999	1.03600	156.50000	1.35900
252.20000	1.04900	154.80000	1.35900
250.39999	1.05100	153.00000	1.38500
248.70000	1.06700	151.30000	1.46200
247.00000	1.07400	149.60001	1.48700
245.20000	1.07700	147.80000	1.46200
243.50000	1.10300	146.10001	1.61500
241.70000	1.23100	144.30000	1.87200
240.00000	1.20500	142.60001	2.30800
238.30000	1.10300	140.89999	2.41000
236.50000	1.02600	139.10001	2.56400
234.80000	1.07700	137.39999	2.53800
233.00000	1.05100	135.70000	2.38500
231.30000	1.15400	133.89999	2.20500
229.60001	1.10300	132.20000	1.84600
227.80000	1.35900	130.39999	1.23100
226.10001	1.28200	128.70000	1.02600
224.30000	1.10300	127.00000	0.96200
222.60001	1.12800	125.20000	0.94900
220.89999	1.17900	123.50000	0.94900
219.10001	1.12800	122.00000	0.97400
217.30000	1.23100	121.00000	1.21050
215.70000	1.15400	120.00000	1.24601
213.89999	1.35900	119.00000	1.16644
212.20000	1.25600	118.00000	1.23946
210.39999	1.17900	117.00000	1.40012
208.70000	1.17900	116.00000	1.60825
207.00000	1.41000	115.00000	1.73118
205.20000	1.25600	114.00000	1.57447
203.50000	1.12800	113.00000	1.39845
201.70000	1.10300	112.00000	1.25046
200.00000	1.10300	111.00000	1.16409
198.30000	1.20500	110.00000	1.20748
196.50000	1.17900	109.00000	1.31564
194.80000	1.12800	108.00000	1.33385
193.00000	1.12800	107.00000	1.32984
191.30000	1.24400	106.00000	1.37800
189.60001	1.16700	105.00000	1.28915
187.80000	1.14100	104.00000	1.25864
186.10001	1.12800	103.00000	1.25669
184.30000	1.15400	102.00000	1.27154
182.60001	1.17900	101.00000	1.23943
180.89999	1.20500	100.00000	1.21929
179.10001	1.23100	99.00000	1.23676
177.39999	1.25600	98.00000	1.23833
175.70000	1.28200	97.00000	1.23616
174.00000	1.19125	96.00000	1.25684
173.00000	1.21211	95.00000	1.16326
172.00000	1.81136	94.00000	1.12402
171.00000	1.90584	93.00000	1.14252

92.00000	1.17488
91.00000	1.25330
90.00000	1.28554
89.00000	1.15596
88.00000	1.18291
87.00000	1.20480
86.00000	1.22846
85.00000	1.22825
84.00000	1.12947
83.00000	1.20049
82.00000	1.30278
81.00000	1.49143
80.00000	1.57190
79.00000	1.41404
78.00000	1.18825
77.00000	1.11011
76.00000	1.15544
75.00000	1.17252
74.00000	1.18375
73.00000	1.24372
72.00000	1.23881
71.00000	1.27308
70.00000	1.18596
69.00000	1.20700
68.00000	1.16694
67.00000	1.41557
66.00000	1.28777
65.00000	1.23321
64.00000	1.27915
63.00000	1.25798
62.00000	1.25301
61.00000	1.27147
60.00000	1.29562
59.00000	1.42248
58.00000	1.50003
57.00000	1.50052
56.00000	1.46633
55.00000	1.46731
54.00000	1.49022
53.00000	1.38205
52.00000	1.30369
51.00000	1.23953
50.00000	1.39136
49.00000	1.39013
48.00000	1.16366
47.00000	1.18657
46.00000	1.14461
45.00000	1.17679
44.00000	1.19390
43.00000	1.58209
42.00000	1.43835
41.00000	1.34496
40.00000	1.27524
39.00000	1.23071
38.00000	1.13585
37.00000	1.17388
36.00000	1.41659
35.00000	1.61829
34.00000	1.37929
33.00000	1.10187

32.00000	1.10260
31.00000	0.94870
30.00000	1.10260
29.00000	0.92310
28.00000	1.23080
27.00000	0.94870
26.00000	1.07690
25.00000	1.00000
24.00000	1.09724
23.97704	1.11725
23.77651	1.24454
23.52565	1.45461
23.26404	1.80744
22.94391	1.89317
22.89023	1.88802
22.63175	1.57091
22.37991	1.35440
22.16490	1.29846
21.91037	1.27935
21.65904	1.29503
21.40858	1.33231
21.15756	1.38464
20.90424	1.46937
20.65408	1.55716
20.33248	1.62809
20.03248	1.72850
19.73248	1.75466
19.43248	1.77996
19.13248	1.80037
18.83248	1.83485
18.53247	1.86841
18.23248	1.90351
18.18307	1.88607
17.89537	1.60187
17.69325	1.25424
17.44064	1.20299
17.18839	1.34701
17.13939	1.38044
16.88514	1.52050
16.63298	1.56380
16.61737	1.56845
16.45292	1.56401
16.45272	1.56402
16.45257	1.56401
16.45237	1.56401
16.45227	1.56400
16.45217	1.56402
16.45202	1.56401
16.45182	1.56402
16.45167	1.56401
16.45147	1.56402
16.45132	1.56401
16.45112	1.56401
16.45102	1.56400
16.45092	1.56402
16.45077	1.56401
16.45057	1.56401
16.45047	1.56400
16.45037	1.56402
16.45022	1.56401

16.45002	1.56401	16.44312	1.56401
16.44992	1.56400	16.44302	1.56400
16.44982	1.56402	16.44292	1.56401
16.44967	1.56401	16.44282	1.56400
16.44947	1.56401	16.44272	1.56401
16.44937	1.56400	16.44262	1.56400
16.44927	1.56402	16.44252	1.56401
16.44912	1.56401	16.44242	1.56400
16.44892	1.56401	16.44232	1.56401
16.44882	1.56400	16.44222	1.56400
16.44872	1.56402	16.44212	1.56401
16.44857	1.56401	16.44202	1.56400
16.44837	1.56401	16.44192	1.56401
16.44827	1.56400	16.44182	1.56400
16.44817	1.56401	16.44172	1.56401
16.44807	1.56400	16.44162	1.56400
16.44797	1.56402	16.44152	1.56401
16.44782	1.56401	16.44142	1.55741
16.44762	1.56401	16.27588	1.55239
16.44752	1.56400	16.11927	1.56398
16.44742	1.56401	16.11912	1.56399
16.44732	1.56400	16.11901	1.56398
16.44722	1.56401	16.11887	1.56400
16.44712	1.56400	16.11876	1.56398
16.44702	1.56402	16.11862	1.56399
16.44687	1.56401	16.11852	1.56398
16.44667	1.56401	16.11836	1.56400
16.44657	1.56400	16.11827	1.56398
16.44647	1.56401	16.11811	1.56399
16.44637	1.56400	16.11802	1.56398
16.44627	1.56401	16.11786	1.56400
16.44617	1.56400	16.11777	1.56399
16.44607	1.56401	16.11767	1.56400
16.44597	1.56400	16.11757	1.56399
16.44587	1.56401	16.11746	1.56845
16.44577	1.56400	15.87679	1.57620
16.44567	1.56402	15.62145	1.52433
16.44552	1.56401	15.55913	1.51672
16.44532	1.56401	15.30802	1.57275
16.44522	1.56400	15.02974	1.62108
16.44512	1.56401	14.92974	1.63023
16.44502	1.56400	14.86333	1.63732
16.44492	1.56401	14.51391	1.82362
16.44482	1.56400	14.18691	1.87175
16.44472	1.56401	14.08081	1.87111
16.44462	1.56400	13.81114	1.78485
16.44452	1.56401	13.64609	1.76943
16.44442	1.56400	13.31282	1.84683
16.44432	1.56401	13.28954	1.81793
16.44422	1.56400	13.22888	1.68504
16.44412	1.56401	13.22888	1.68504
16.44402	1.56400	12.92888	1.58160
16.44392	1.56401	12.67225	1.22473
16.44382	1.56400	12.65710	1.21099
16.44372	1.56401	12.40160	1.09737
16.44362	1.56400	12.14571	1.07340
16.44352	1.56401	12.11413	1.07350
16.44342	1.56400	11.85165	1.07948
16.44332	1.56401	11.82003	1.07942
16.44322	1.56400	11.56616	1.07746

11.37816	1.06777
11.12610	0.97574
10.94504	0.89928
10.92230	0.89087
10.67030	1.02754
10.63348	1.09456
10.58780	1.10564
10.42636	1.09450
10.00000	0.89000
9.00000	0.70000
8.00000	0.51000
7.00000	0.35000
6.00000	0.29490
5.00000	0.29490
4.00000	0.29490
3.00000	0.29490
2.00000	0.29490
1.00000	0.29490
0.00000	0.29490

Appendix S

Numerical Modeling Results for Lake San Agustin

Age (years)	Simulated $\delta^{18}O$(‰)
35375.00000000000	27.8909690159664 7
35305.00000000000	28.03701042666295
35205.00000000000	27.95625599383913
35107.00000000000	28.95670554088119
35005.00000000000	30.50873393599122
34908.00000000000	29.51145048854726
34805.00000000000	30.40544841553854
34764.00000000000	29.43674162918297
30712.00000000000	30.45765682960249
30656.55826198840	30.38876926181886
30602.93626276118	30.41346448719444
30574.82318221561	30.40583681447390
30366.07397259690	30.09329444949259
30306.07397259690	30.07120975471781
30206.07397259690	30.00832995154679
30106.07397259690	29.93506865610509
30006.07397259690	29.85109936388982
29900.00000000000	29.90609613308040
29800.00000000000	30.05290086095476
29701.00000000000	30.48907430976071
29601.00000000000	30.74197975215367
29501.00000000000	30.85944103204341
29403.00000000000	30.73290544243641
29302.00000000000	29.84019797602263
29204.00000000000	29.64024519238361
29106.00000000000	30.71740052771159
29006.00000000000	30.14848036942374
28906.00000000000	30.05897286736907
28806.00000000000	29.98959725975933
28700.00000000000	30.33980132475599
28600.00000000000	31.00541201667049
28500.00000000000	31.84202081879457
28402.82381478206	31.30890964049210
28300.02340965525	31.20842325319120
28202.02979980405	31.06683105307418
28103.32587359785	30.85948780591462
28002.03646687412	30.72516331669548
27907.00000000000	30.67039271673339
27800.00000000000	31.23900949995899
27702.91423817554	31.01808758599006
27603.89545016307	31.17006795695904
27502.29410541787	30.99316520657012
27401.00953829803	30.97961324126607
27300.41307845174	31.11853666505149
27202.48900329773	31.46169549272518
27102.96613347247	31.50767127765853
27000.80473995902	31.12275717068509
26901.19354879338	30.92869770075360
26800.42593245354	31.09404490592057
26701.93583323706	30.69522774024152
26599.74391173445	30.32641039675569
26507.68295533355	29.58767317859448
26374.00000000000	28.76435197159219
26304.00000000000	28.53348230260141
26204.00000000000	27.30936016813539
26105.00000000000	28.56000179087092
26007.00000000000	28.76243098361640
25907.00000000000	30.47639125907830
25809.00000000000	31.43165657534109
25702.00000000000	29.65237178265933
25603.00000000000	29.00049468600574
25502.00000000000	28.53316386946717
25402.00000000000	29.33359962000214
25304.22215978120	30.80501111720442
25206.00000000000	30.93070822886380
25108.00000000000	29.68119926713508
25008.00000000000	28.23850199687353
24908.00000000000	28.55144560828433
24802.00000000000	28.84423257020196
24700.00000000000	30.49859670902067
24603.00000000000	30.88460785730851
24502.00000000000	31.05605778887816
24402.00000000000	31.46885061207294
24302.00000000000	31.65561980511418
24205.00000000000	31.63236062615015

24105.00000000000	31.21496391102439
24000.00000000000	30.88531965459536
23900.00000000000	31.46694626849873
23801.00000000000	32.31656504790596
23701.00000000000	30.86416157107589
23602.00000000000	30.91512366642686
23505.00000000000	32.40640255663234
23406.00000000000	31.61263369389348
23305.00000000000	29.79900506200793
23207.00000000000	31.86164250790677
23107.00000000000	31.57119609619666
23004.31875477187	32.49974897026688
22909.11221664144	31.35583010189274
22799.11221664144	31.08134873453995
22709.11221664144	30.65436954540162
22605.00000000000	29.51479092633391
22504.00000000000	29.77418070012259
22406.00000000000	29.22160053931794
22306.00000000000	31.99810542638231
22206.00000000000	31.92871036026292
22106.00000000000	32.00716521948537
22006.00000000000	32.24484208626552
21901.00000000000	32.28418836976416
21801.00000000000	31.84897172940070
21702.00000000000	31.26382454114382
21602.00000000000	31.05894202555745
21509.00000000000	31.34379082107037
21406.00000000000	31.38253595476781
21308.00000000000	31.79853067682184
21208.00000000000	31.78591366556800
21100.00000000000	32.05305800740646
21009.00000000000	31.31688596657247
20901.00000000000	31.15343562844687
20803.00000000000	32.03901632581754
20702.00000000000	31.88419142478239
20605.00000000000	31.56853349645321
20507.00000000000	30.53412807873303
20406.00000000000	31.12021510785970
20308.00000000000	31.70481898632795
20208.00000000000	31.09523238057446
20109.00000000000	31.11862694089421
20001.00000000000	31.96517025255232
19904.00000000000	31.98534095180043
19802.00000000000	31.80180667454779
19705.00000000000	32.79185634252038
19605.00000000000	32.39891165658816
19501.25577154217	31.63366928910762
19408.00000000000	28.76264988980780
19300.00000000000	30.67189240003814
19200.00000000000	30.97790898742844
19107.00000000000	30.76162250310416
19006.00000000000	31.23235836954468
18903.00000000000	31.99196844439079
18802.00000000000	31.81807060543076
18705.00000000000	32.00715526696400
18607.00000000000	31.24787428462690
18506.00000000000	30.96618221783805
18403.00000000000	31.54502635758562
18303.00000000000	30.10827495869471
18200.00000000000	32.71509474853879
18100.00000000000	31.29883965819202
18000.00000000000	32.10601019156081
17900.00000000000	32.00476162054410
17800.00000000000	32.14105116885915
17700.00000000000	32.41284101898976
17601.00000000000	31.33879685941388
17502.00000000000	31.10124078465495
17401.00000000000	30.50667055183976
17304.00000000000	30.54411546421397
17206.00000000000	30.60816254634003
17102.00000000000	30.19275115108345
17002.00000000000	30.62939787733013
16901.00000000000	30.99637540206216
16801.00000000000	31.21643840614452
16701.00000000000	31.37274206290026
16607.00000000000	31.15137897500094

16502.00000000000	32.19233896159792
16402.00000000000	32.03977117192515
16305.34352587836	31.93941301850739
16205.34352587836	31.44818428536938
16105.34352587836	31.33993484406991
16000.00000000000	31.26447829252951
15900.00000000000	31.50378412919848
15800.00000000000	31.68196911871902
15700.00000000000	31.89149957235354
15600.00000000000	32.09886322082661
15500.00000000000	32.34020461706251
15400.00000000000	32.73935749420003
15363.00000000000	33.06668799234927

Age (years)	Area (m²)
_DUB1:[SYS1.ISOTOPE.PEGGY.MODEL]AREAOUT.DAT;2	22-MAR-1991 16:26 Page 1
35375.00000000000	680645703.1406250
35275.00000000000	674443099.1719718
35175.00000000000	698024949.0409381
35077.00000000000	578431197.8413554
34975.00000000000	468571219.5970658
34878.00000000000	316371144.7487890
34775.00000000000	161772619.5276147
34765.00000000000	158973591.8497205
34764.00000000000	158729241.2853238
30712.00000000000	186710365.9509690
30610.28125113584	185250215.8018397
30553.11395106157	187234634.2334977
30376.07397259690	245363611.3052019
30276.07397259690	279976216.6501175
30176.07397259690	314810138.2749086
30076.07397259690	350025328.7903858
29976.07397259690	385338235.7004805
29870.00000000000	404051533.5884676
29771.00000000000	398511982.1956313
29661.00000000000	329702838.6510387
29561.00000000000	266855539.3557654
29461.00000000000	216108259.8325549
29363.00000000000	191615994.3414383
29262.00000000000	233976040.6590218
29164.00000000000	257136656.4259794
29066.00000000000	221331121.5505786
28966.00000000000	228930683.0505632
28866.00000000000	232156047.5664130
28760.00000000000	230344058.3065872
28660.00000000000	210815013.4356997
28560.00000000000	164141128.2680661
28455.91420672506	147121349.1726574
28402.82381478206	159160175.7247359
28353.25344647414	170180380.9228434
28300.02340965525	161022280.1399965
28250.96981152586	159425142.2476723
28202.02979980405	168492489.6279936
28151.70744462619	176106187.1463915
28103.32587359785	171722952.2986736
28051.70647933103	170697264.1856188
28002.03646687412	179729011.9637806
27950.58689343433	187483358.7826831
27896.18081949227	186945826.4811628
27851.39851963470	165138937.1096579
27800.00000000000	141618235.8151677
27749.76827778204	147995438.4794517
27702.91423817554	148876311.9780689
27652.39866837271	149040818.3272108
27603.89545016307	151539671.4285697
27551.89288151102	155433566.3493392
27502.29410541787	160090920.0852071
27448.20965788650	165186540.3507500
27401.00953829803	161850040.2333188
27351.22665217336	158312871.1119056
27300.41307845174	153532411.1108761
27253.46239618992	148393759.6915952
27202.48900329773	144327801.4808772
27150.01615962909	143613106.5596101
27102.96613347247	153182812.6804472
27048.58779942876	161327349.7306014
27000.80473995902	168891611.3315433
26948.54872735334	176769514.5994397
26901.19354879338	183875450.0094975
26849.52163156433	157083801.3074242
26800.42593245354	155290461.9693025
26746.99816292478	167518786.7170019
26701.93583323706	178412621.4923457
26652.74762879446	167721802.5007396
26599.74391173445	157900530.0685722
26549.00712551221	159853633.5725078
26497.68295533355	178206723.1051355
26464.46513072198	187625431.0784375
26386.96232403116	203119269.5497410
26304.00000000000	369410417.1764829
26204.00000000000	515422079.7784703

26105.00000000000	744465246.6455248
25997.00000000000	787670751.2350705
25907.00000000000	756821378.4922718
25809.00000000000	591971604.6671610
25752.00000000000	521609340.3077203
25702.00000000000	540787696.0016306
25603.00000000000	633674612.6103119
25502.00000000000	774433966.2993105
25402.00000000000	773749633.5312074
25304.22215978120	651716945.4457584
25206.00000000000	499045183.6088848
25108.00000000000	495547479.8569586
25008.00000000000	569400360.3458562
24908.00000000000	701275146.1789804
24802.00000000000	769804668.9375053
24700.00000000000	666486075.1075461
24603.00000000000	520860722.7236761
24502.00000000000	527754429.2842281
24402.00000000000	514484958.3506629
24302.00000000000	501033589.2262040
24205.00000000000	495985085.7256667
24105.00000000000	494769695.5571100
24000.00000000000	498674304.8995535
23900.00000000000	505985151.3814947
23801.00000000000	483772259.9451430
23701.00000000000	489030518.3744908
23602.00000000000	497199925.9302889
23505.00000000000	454837620.6741665
23406.00000000000	378844893.1386111
23305.00000000000	464823631.2087201
23207.00000000000	452719957.1340341
23107.00000000000	376275050.3118588
23004.31875477187	271611390.2134641
22909.11221664144	206903859.5593370
22809.11221664144	302931478.4666668
22709.11221664144	410848842.6566359
22605.00000000000	498018781.0195768
22504.00000000000	577784896.3703658
22406.00000000000	789352488.5497211
22306.00000000000	739725810.3830367
22206.00000000000	677955368.6684228
22096.00000000000	642425287.8984896
22006.00000000000	613253187.8050106
21901.00000000000	574435732.2307705
21771.00000000000	656754764.0451968
21702.00000000000	717423981.1281065
21602.00000000000	723816999.9970900
21509.00000000000	709253799.5158042
21406.00000000000	702944373.9873050
21308.00000000000	680977254.1912496
21200.00000000000	602724571.0743114
21100.00000000000	545404686.7772548
21009.00000000000	557100046.2211658
20901.00000000000	607131105.0695794
20803.00000000000	544089056.2935697
20702.00000000000	575086126.5817978
20605.00000000000	561659996.8620679
20507.00000000000	593734037.8650007
20406.00000000000	600472804.7756432
20298.00000000000	521684450.7206977
20208.00000000000	538446802.1781470
20109.00000000000	561621806.5473592
20001.00000000000	513483729.9417412
19904.00000000000	497631379.3318452
19802.00000000000	496846915.4286691
19705.00000000000	432039609.6877251
19605.00000000000	336343634.4217594
19501.25577154217	247942982.4411032
19398.00000000000	523053611.2553179
19300.00000000000	651501650.1429870
19200.00000000000	620619242.8190217
19097.00000000000	628067006.3709010
19006.00000000000	636233999.2149485
18903.00000000000	561062944.3254260
18802.00000000000	515555358.5432683
18705.00000000000	507762246.7340190

18607.00000000000	498164938.4624175
18506.00000000000	514632927.0855709
18403.00000000000	496028547.4953394
18300.00000000000	502264147.1715940
18200.00000000000	492199901.2961829
18090.00000000000	455729047.9245656
18000.00000000000	373830887.2249080
17900.00000000000	263057381.4199342
17800.00000000000	189999705.0309148
17700.00000000000	159713241.6045634
17591.00000000000	155245552.9807510
17502.00000000000	155602191.7775321
17401.00000000000	163610695.2690444
17304.00000000000	219916025.8982226
17206.00000000000	324728842.2350990
17102.00000000000	483488803.0665680
17001.00000000000	499094332.5633777
16901.00000000000	508410565.2563979
16801.00000000000	504444949.3179047
16701.00000000000	497850651.7912587
16607.00000000000	491559710.1863959
16492.00000000000	446694219.2412362
16402.00000000000	340124937.3332794
16305.34352587836	232883800.0144815
16205.34352587836	233422609.6088024
16105.34352587836	240474600.7863375
16000.00000000000	247778784.3551018
15900.00000000000	234040257.0778466
15800.00000000000	219377726.2537336
15700.00000000000	202178612.2809464
15600.00000000000	183937882.8057421
15500.00000000000	167156409.6331781
15400.00000000000	155780981.4769961
15363.00000000000	154900391.9227704

Age (years)	Inflow (m^3 yr^{-1})
35375.00000000000	383570000.0000000
35305.00000000000	383570000.0000000
35205.00000000000	401570000.0000000
35107.00000000000	345970000.0000000
35005.00000000000	273170000.0000000
34908.00000000000	225170000.0000000
34805.00000000000	109570000.0000000
30704.04072589621	109274000.0000000
30602.93626276118	109274000.0000000
30506.07397259690	122400684.9325574
30406.07397259690	140400684.9325574
30306.07397259690	158400684.9325574
30206.07397259690	176400684.9325574
30106.07397259690	194400684.9325574
30006.07397259690	212400684.9325574
29900.00000000000	224294000.0000000
29800.00000000000	224294000.0000000
29701.00000000000	200294000.0000000
29601.00000000000	170294000.0000000
29501.00000000000	140294000.0000000
29403.00000000000	116894000.0000000
29302.00000000000	121794000.0000000
29204.00000000000	148994000.0000000
29106.00000000000	131394000.0000000
29006.00000000000	131394000.0000000
28906.00000000000	131394000.0000000
28806.00000000000	131394000.0000000
28700.00000000000	126394000.0000000
28600.00000000000	121394000.0000000
28500.00000000000	126394000.0000000
28402.82381478206	137076381.4782060
28300.02340965525	126796340.9655249
28250.96981152586	121890981.1525857
28202.02979980405	116996979.9804047
28103.32587359785	118721300.7441828
28002.03646687412	127330900.3156998
27902.92359875070	134593719.7501409
27800.00000000000	114009000.0000000
27702.91423817554	118517576.1824459
27603.89545016307	124709000.0000000
27502.29410541787	124709000.0000000
27401.00953829803	124709000.0000000
27300.41307845174	124709000.0000000
27202.48900329773	124709000.0000000
27102.96613347247	124709000.0000000
27000.80473995902	124709000.0000000
26901.19354879338	124709000.0000000
26800.42593245354	124709000.0000000
26701.93583323706	124709000.0000000
26604.37962372834	107384924.7456683
26507.68295533355	118172408.9332910
26406.96232403116	138316535.1937678
26304.00000000000	249909000.0000000
26204.00000000000	399909000.0000000
26105.00000000000	458409000.0000000
26007.00000000000	560409000.0000000
25907.00000000000	535409000.0000000
25809.00000000000	448909000.0000000
25702.00000000000	436309000.0000000
25603.00000000000	465109000.0000000
25502.00000000000	488609000.0000000
25402.00000000000	448609000.0000000
25304.22215978120	409497863.9124820
25206.00000000000	382209000.0000000
25108.00000000000	362609000.0000000
25008.00000000000	402609000.0000000
24908.00000000000	442609000.0000000
24802.00000000000	465809000.0000000
24700.00000000000	424849000.0000000
24603.00000000000	364549000.0000000
24502.00000000000	368549000.0000000
24402.00000000000	373549000.0000000
24302.00000000000	378549000.0000000
24205.00000000000	380399000.0000000
24105.00000000000	380399000.0000000
24000.00000000000	379899000.0000000

23900.00000000000	378899000.0000000
23801.00000000000	376309000.0000000
23701.00000000000	371309000.0000000
23602.00000000000	367859000.0000000
23505.00000000000	346859000.0000000
23406.00000000000	301159000.0000000
23305.00000000000	344409000.0000000
23207.00000000000	339509000.0000000
23107.00000000000	289509000.0000000
23004.31875477187	197955256.6806127
22909.11221664144	177547280.8472084
22809.11221664144	232547280.8472084
22709.11221664144	287547280.8472084
22605.00000000000	323809000.0000000
22504.00000000000	378309000.0000000
22406.00000000000	546309000.0000000
22306.00000000000	486309000.0000000
22206.00000000000	486309000.0000000
22106.00000000000	486309000.0000000
22006.00000000000	486309000.0000000
21901.00000000000	486309000.0000000
21801.00000000000	486309000.0000000
21702.00000000000	481809000.0000000
21602.00000000000	466809000.0000000
21509.00000000000	466359000.0000000
21406.00000000000	463359000.0000000
21308.00000000000	433959000.0000000
21200.00000000000	433959000.0000000
21100.00000000000	423959000.0000000
21009.00000000000	396859000.0000000
20901.00000000000	400459000.0000000
20803.00000000000	403259000.0000000
20702.00000000000	384259000.0000000
20605.00000000000	398759000.0000000
20507.00000000000	449959000.0000000
20406.00000000000	474559000.0000000
20308.00000000000	469559000.0000000
20208.00000000000	459559000.0000000
20109.00000000000	461659000.0000000
20001.00000000000	469259000.0000000
19904.00000000000	460709000.0000000
19802.00000000000	449309000.0000000
19705.00000000000	392809000.0000000
19605.00000000000	302809000.0000000
19501.25577154217	263992839.6819154
19408.00000000000	525109000.0000000
19300.00000000000	525109000.0000000
19200.00000000000	525109000.0000000
19107.00000000000	570109000.0000000
19006.00000000000	576859000.0000000
18903.00000000000	517659000.0000000
18802.00000000000	484659000.0000000
18705.00000000000	463109000.0000000
18607.00000000000	453609000.0000000
18506.00000000000	438609000.0000000
18403.00000000000	420359000.0000000
18300.00000000000	420359000.0000000
18200.00000000000	308359000.0000000
18100.00000000000	308359000.0000000
18000.00000000000	254359000.0000000
17900.00000000000	194359000.0000000
17800.00000000000	142959000.0000000
17700.00000000000	125959000.0000000
17601.00000000000	122729000.0000000
17502.00000000000	122729000.0000000
17401.00000000000	127729000.0000000
17304.00000000000	176229000.0000000
17206.00000000000	264429000.0000000
17102.00000000000	374029000.0000000
17001.00000000000	394229000.0000000
16901.00000000000	388229000.0000000
16801.00000000000	382229000.0000000
16701.00000000000	376229000.0000000
16607.00000000000	373589000.0000000
16502.00000000000	346589000.0000000
16402.00000000000	281589000.0000000

16305.34352587836	192589000.0000000
16205.34352587836	192589000.0000000
16105.34352587836	192589000.0000000
16000.00000000000	191519000.0000000
15900.00000000000	180819000.0000000
15800.00000000000	170119000.0000000
15700.00000000000	159419000.0000000
15600.00000000000	148719000.0000000
15500.00000000000	138019000.0000000
15400.00000000000	127319000.0000000
15363.00000000000	123360000.0000000