

USER'S MANUAL

**EROSION DUE TO HIGH FREQUENCY STORM EVENTS
(18 Selected Coastal Counties of Florida)**

November 22, 1995

Prepared For:

**Division of Beaches and Shores
Department of Environmental Protection
Tallahassee, FL 32399**

Prepared By:

**Subarna B. Malakar
and
Robert G. Dean
Department of Coastal and Oceanographic Engineering
University of Florida
Gainesville, FL 32611**

Table of Contents

1	INTRODUCTION	3
2	THE CCCLa EROSION MODEL	3
2.1	DESCRIPTION OF CCCLa	3
2.2	INPUT DATA	4
2.3	DATA FILES	5
2.4	PLOTS OF PARAMETER DATA	6
3	SAMPLE RUN	18
3.1	INPUT PROFILE FILE: SURVEY.DAT	18
3.2	OUTPUT FILE: CCCLa.OUT	19
3.3	OUTPUT FILE: SURVEY.OUT	20
	APPENDIX A	A-1
	APPENDIX B	B-1

USER'S MANUAL

EROSION DUE TO HIGH FREQUENCY STORM EVENTS

1 INTRODUCTION

This User's Manual presents a method of calculating dune erosion due to high frequency storm events for the 24 selected coastal counties of Florida. High frequency is defined herein as having return periods less than or equal to 25 years. Computations are performed for return periods of 25 years, 15 years, 10 years, and 5 years. The background associated with this methodology and the erosion model (CCCL) used are described in "High Frequency Storm Hydrograph Development and Erosion Model Calibration" (1993).

2 THE CCCLa EROSION MODEL

2.1 DESCRIPTION OF CCCLa

The CCCLa erosion model is an application version of the CCCLa erosion model, the erosion model which DEP uses, however the current CCCLa erosion model has undergone some modifications in the course of this study period. The primary difference between the CCCL and the CCCLa models is in the amount of input data required to run the two erosion models. For both models at least 5 parameters must be specified, namely: duration, storm surge height, astronomical tide, wave height and the scale parameter "A" associated with the survey profile.

To run the CCCL erosion model one needs to supply time series data of storm surge heights, astronomical tidal amplitudes and wave heights for each time step for the duration of the erosion model simulation. To run the CCCLa erosion model, mean values for each of the time series data have been determined for the East Coast and the West Coast of Florida such that running the CCCLa erosion model with the mean values will be equivalent to running the CCCL erosion model. Mean values for the time series data for the Panhandle Coast of Florida have been presented here, except for the mean breaking wave heights which

are determined by running the equivalent erosion model CCCLa iteratively given the erosion distances computed from running the CCCL model for at least the 7 severest storm events. Analysis of the erosion distances computed at or above the average survey profile berm elevations indicate little or no erosion considering the present high frequency events for the Panhandle Coast of Florida. Thus breaking wave heights could not be determined above for the Panhandle Coast of Florida. In the CCCLa model, an erosion factor of 1.0 is used.

2.2 INPUT DATA

All the input parameters required to run the CCCLa erosion model have been determined for the East Coast and the West Coast of Florida from the analysis of the field data available to compute erosion distances for a given profile data, for the high frequency return periods. Erosion distances computed for counties: Nassau, Duval, St. Johns, Flagler are at 8 feet elevation, and for counties: Volusia, Brevard, Indian River, St. Lucie, Martin, Palm Beach, Broward and Dade are at 7 feet elevation. Erosion distances computed for counties: Gulf, Franklin, Pinellas, Manatee, Sarasota, Charlotte, Lee and Collier are at 5 feet elevation. (Erosion distances computed for counties: Escambia, Okaloosa, Walton and Bay are at 6 feet elevation.) The scale parameter "A" is computed using the "least squares fit" method using the survey profile data up to the water depth of 12 feet.

The only required data to run the CCCLa erosion model is the survey profile data with offshore survey depths to at least 12 feet, in the formerly DNR data format, that is, it should have a header card with the first 3 characters of the country name in capitals starting column 1, followed by a series of profile data in the DNR format. Next the file containing the profile data must be copied to SURVEY.DAT.

For reference, the list of abbreviations used for the counties on the east coast of Florida are:

NAS, DUV, STJ, FLA

VOL, BRE, IND, STL, MAR, PAL, BRO, DAD

PIN, MAN, SAR, CHA, LEE, COL
GUL, FRA

ESC, OKA, WAL, BAY

Some of the parameters set in the CCCLa erosion model are: time step for erosion computation, $dt=0.05$ hour, erosion contour level, $clev=8$ feet for Nassau, Duval, St Johns and Flagler counties; $clev=7$ feet for counties between Volusia and Dade; $clev=6$ feet for counties between Escambia and Bay; $clev=5$ feet for Gulf, Franklin and counties between Pinellas and Collier, inclusive, and the duration of model run, $dero=18.6$ hours for east and west coast of Florida, and $dero=12.4$ hours for the Panhandle Coast of Florida.

2.3 DATA FILES

There are two files provided for each of the 12 counties on the east coast of Florida, one has the following data: mean astronomical tidal amplitude, and total mean peak storm surge and breaking wave height for the high frequency return periods; the other file has the scale parameter "A" associated with the survey profile. These data are referenced to DNR range monuments of the respective counties such that the values required to run the erosion model CCCLa for a given profile will be interpolated knowing the county and the monument range number.

2.4 PLOTS OF PARAMETER DATA

After smoothing the original computed parameters critical data points have been selected to define the smooth curves for these plots.

- a. Figure 1 shows the distribution of the mean tidal amplitude for the east coast of Florida.
- b. Figure 2 shows the distribution of the total mean peak storm surge for the east coast of Florida.
- c. Figure 3 shows the distribution of the breaking wave height for the east coast of Florida.
- d. Figure 4 shows the distribution for the east coast of Florida of the scale parameter "A" associated with the survey profile.
- e. Figure 5 shows the distribution of the mean tidal amplitude for the west coast of Florida.
- f. Figure 6 shows the distribution of the total mean peak storm surge for the west coast of Florida.
- g. Figure 7 shows the distribution of the breaking wave height for the west coast of Florida.
- h. Figure 8 shows the distribution for the west coast of Florida of the scale parameter "A" associated with the survey profile.
- i. Figure 9 shows the distribution of the mean tidal amplitude for the Panhandle coast of Florida.
- j. Figure 10 shows the distribution of the total mean peak storm surge for the Panhandle coast of Florida.
- k. Figure 4 shows the distribution for the east coast of Florida of the scale parameter "A" associated with the survey profile.

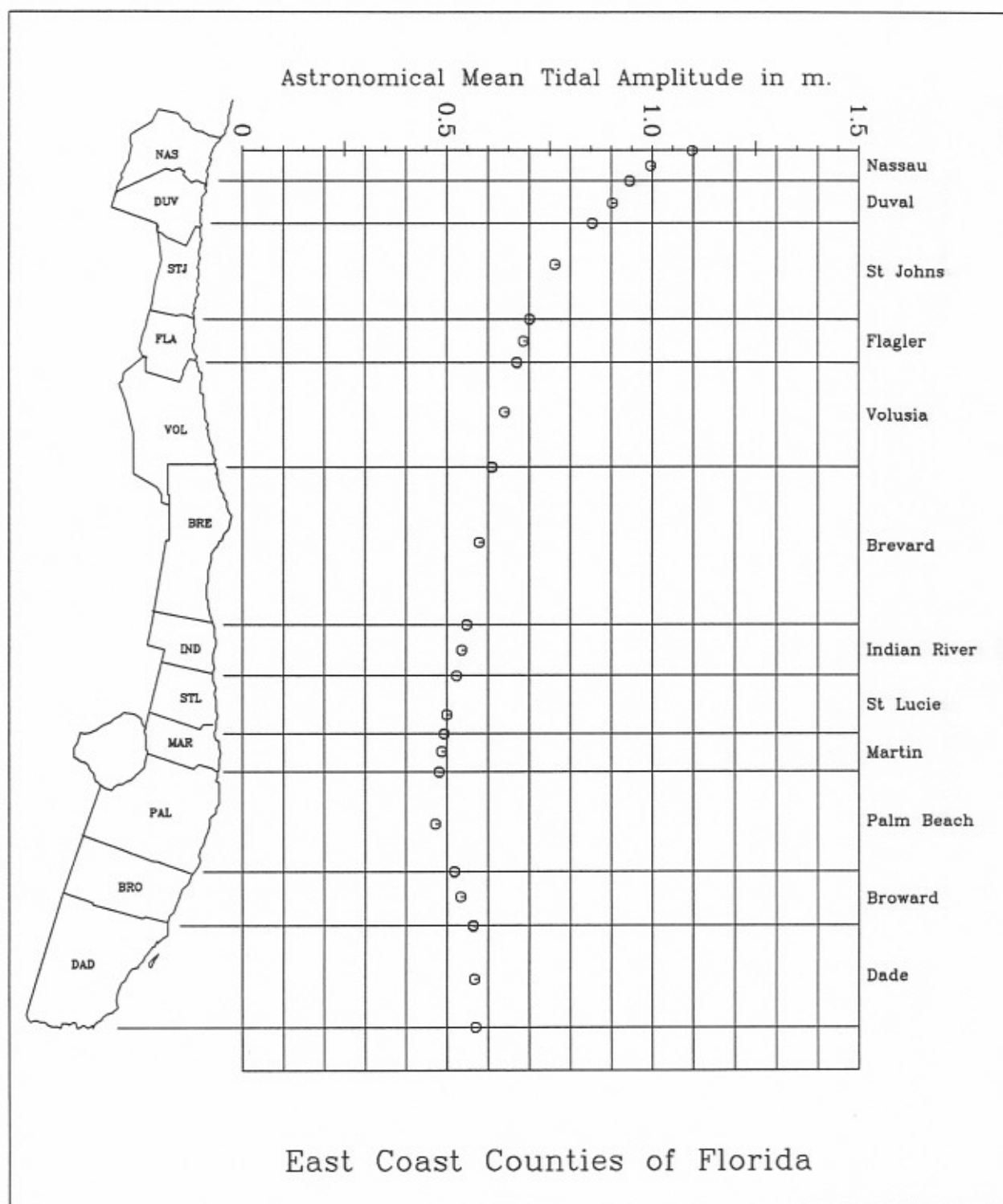


Figure 1. Variation of Mean Astronomical Tidal Amplitude Along the East Coast of Florida.

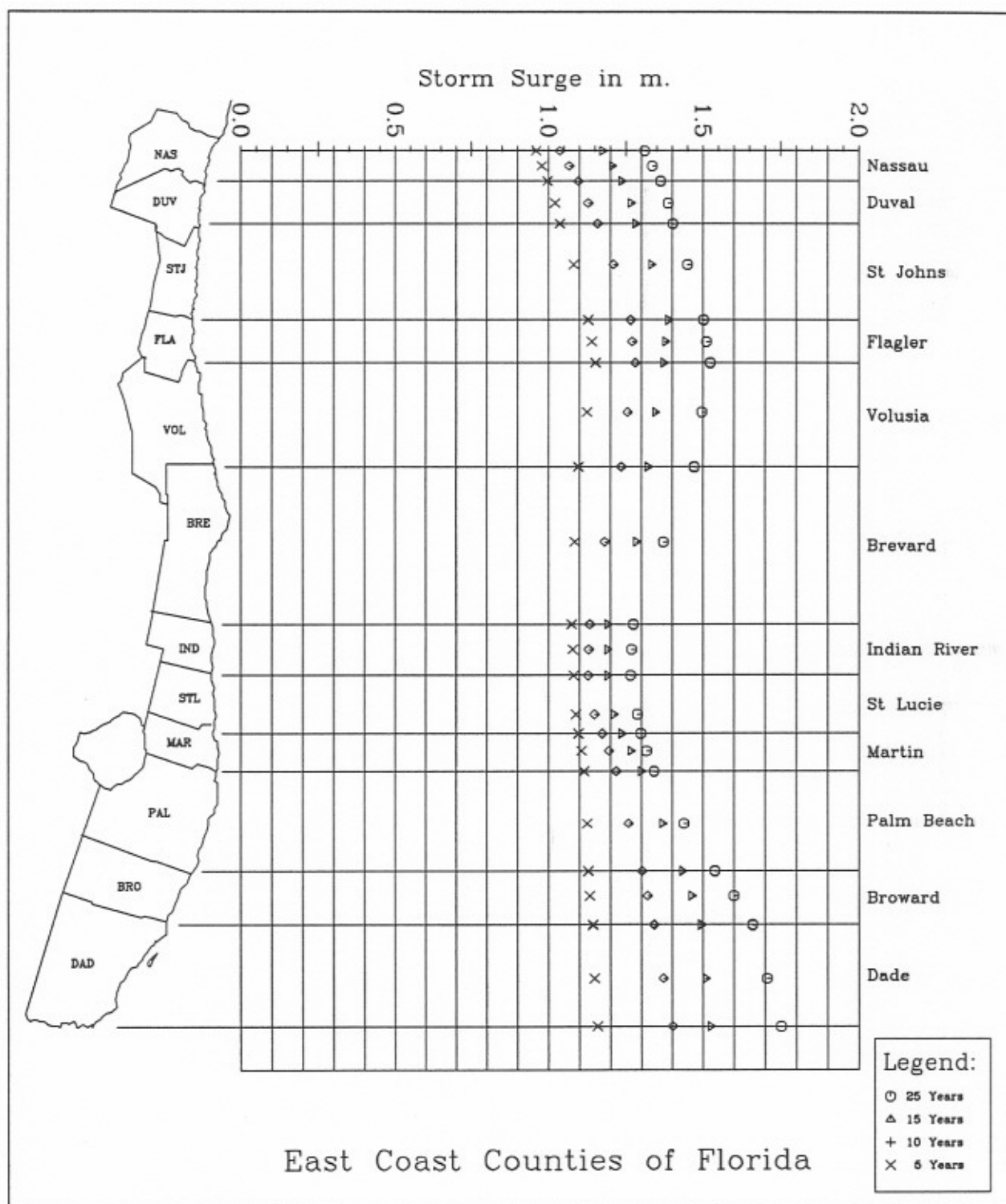


Figure 2. Variation of Equivalent Storm Surge with Return Period Along the East Coast of Florida.

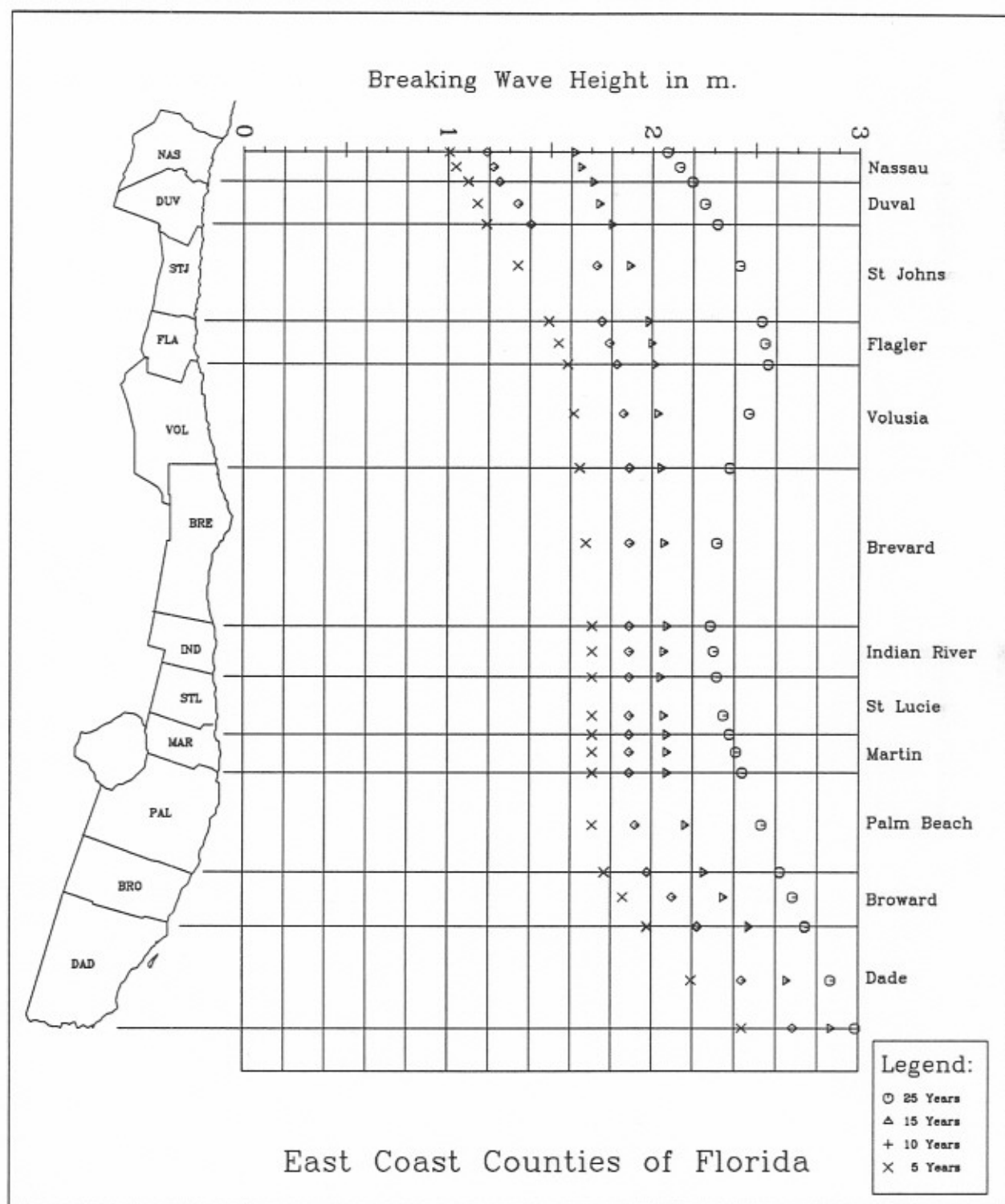


Figure 3. Variation of Equivalent Breaking Wave Height with Return Period Along the East Coast of Florida.

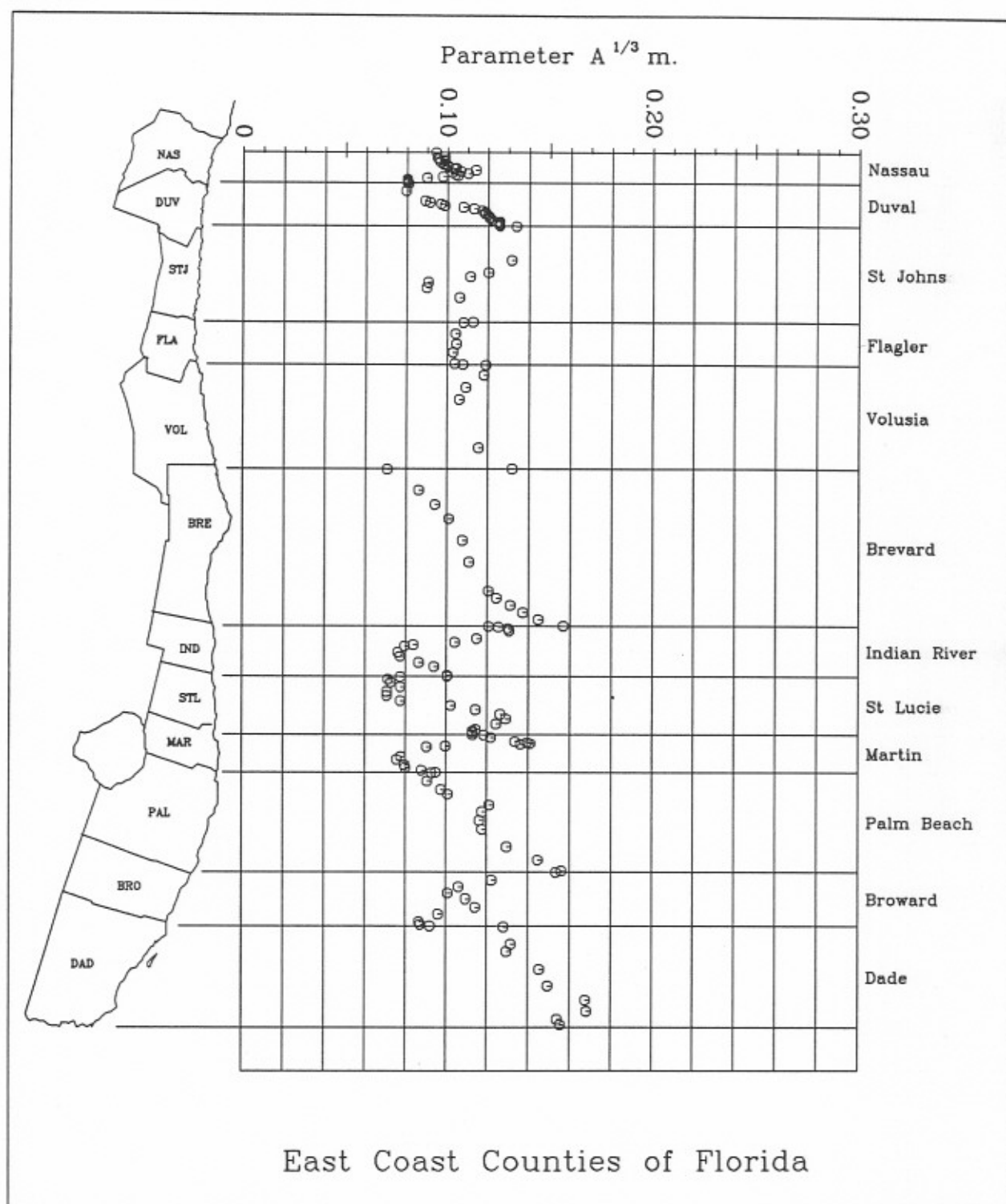


Figure 4. Variation of Scale Parameter "A" Along the East Coast of Florida.

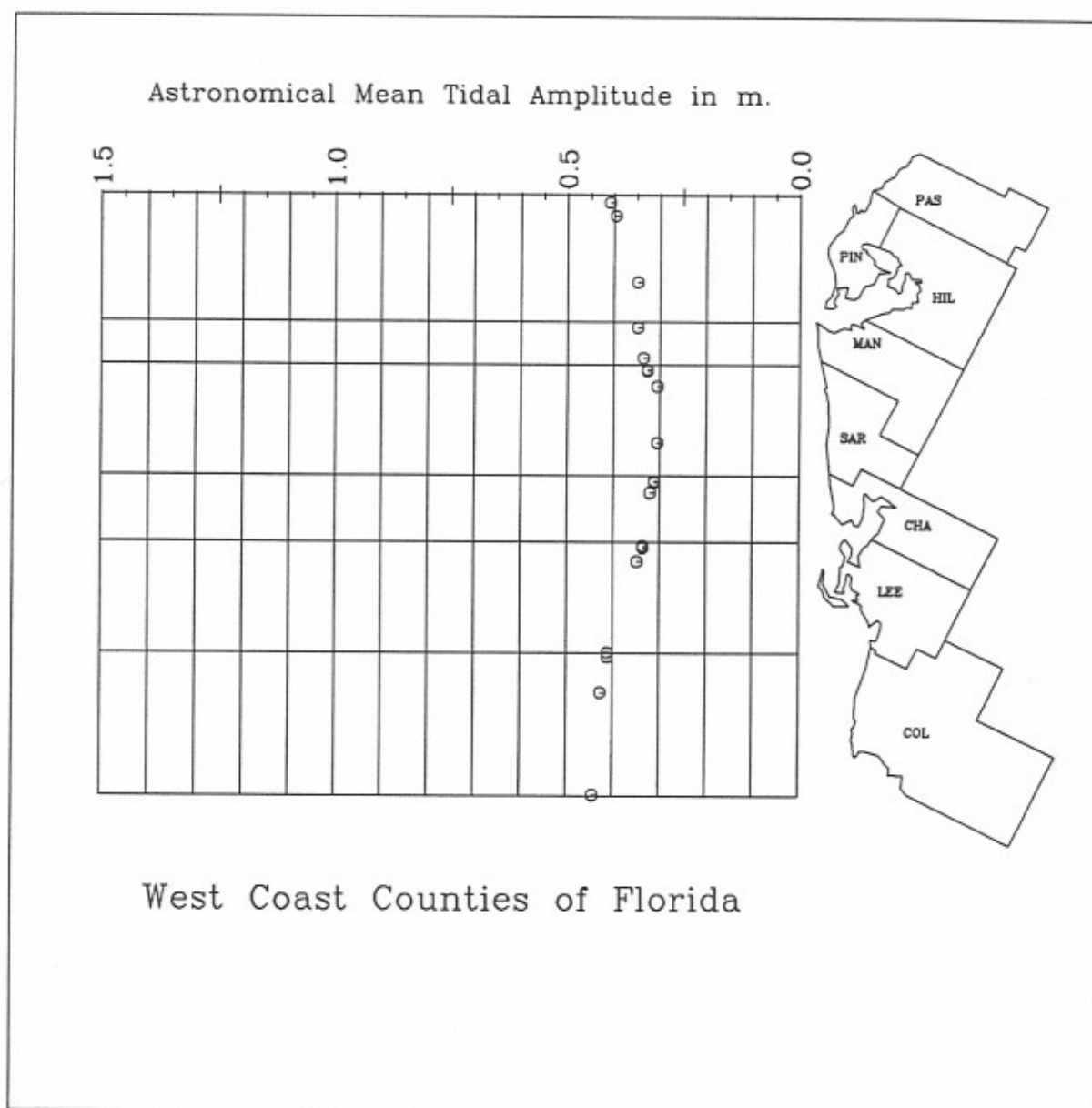


Figure 5. Variation of Mean Astronomical Tidal Amplitude Along the West Coast of Florida.

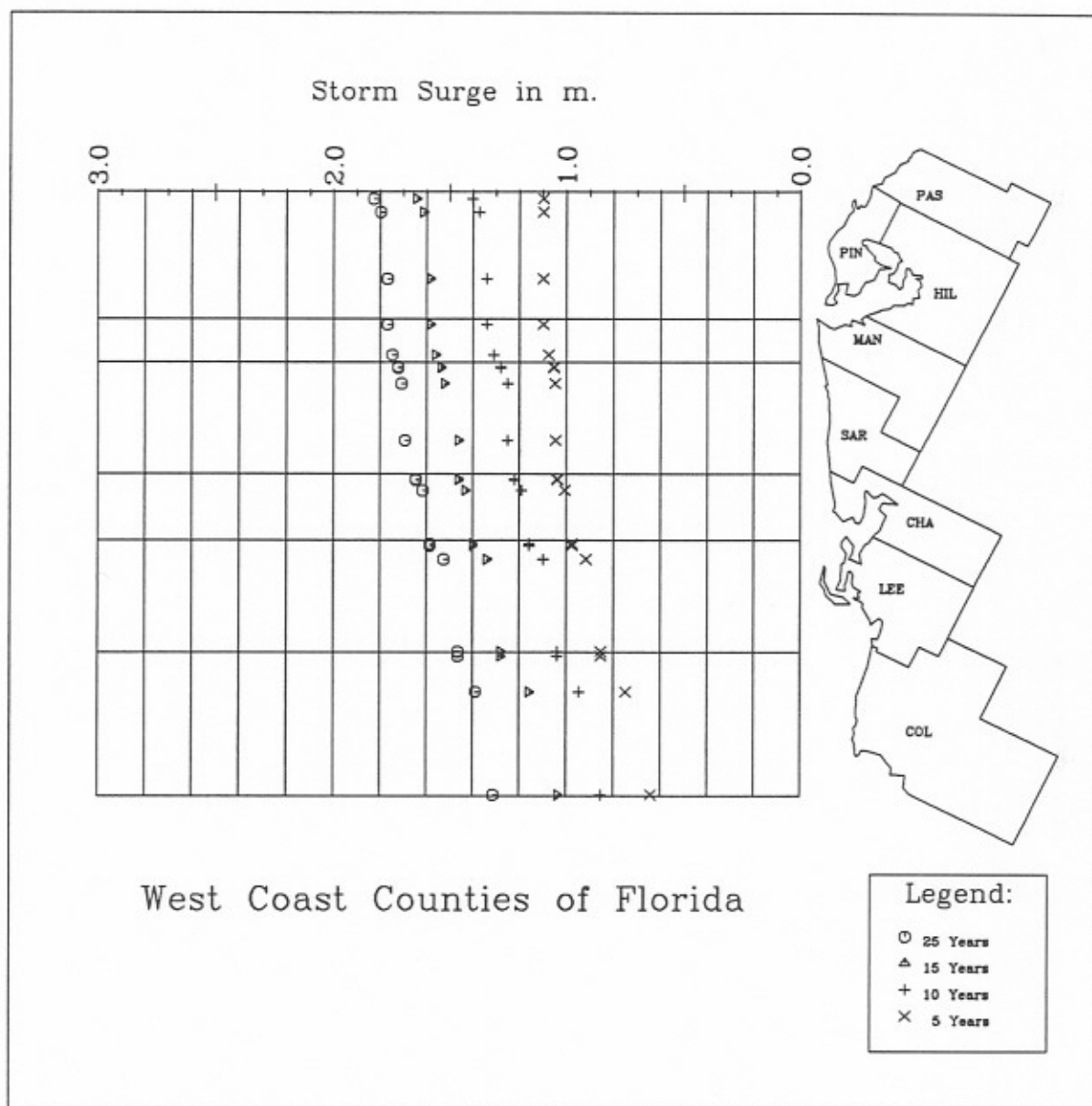


Figure 6. Variation of Equivalent Storm Surge With Return Period Along the West Coast of Florida.

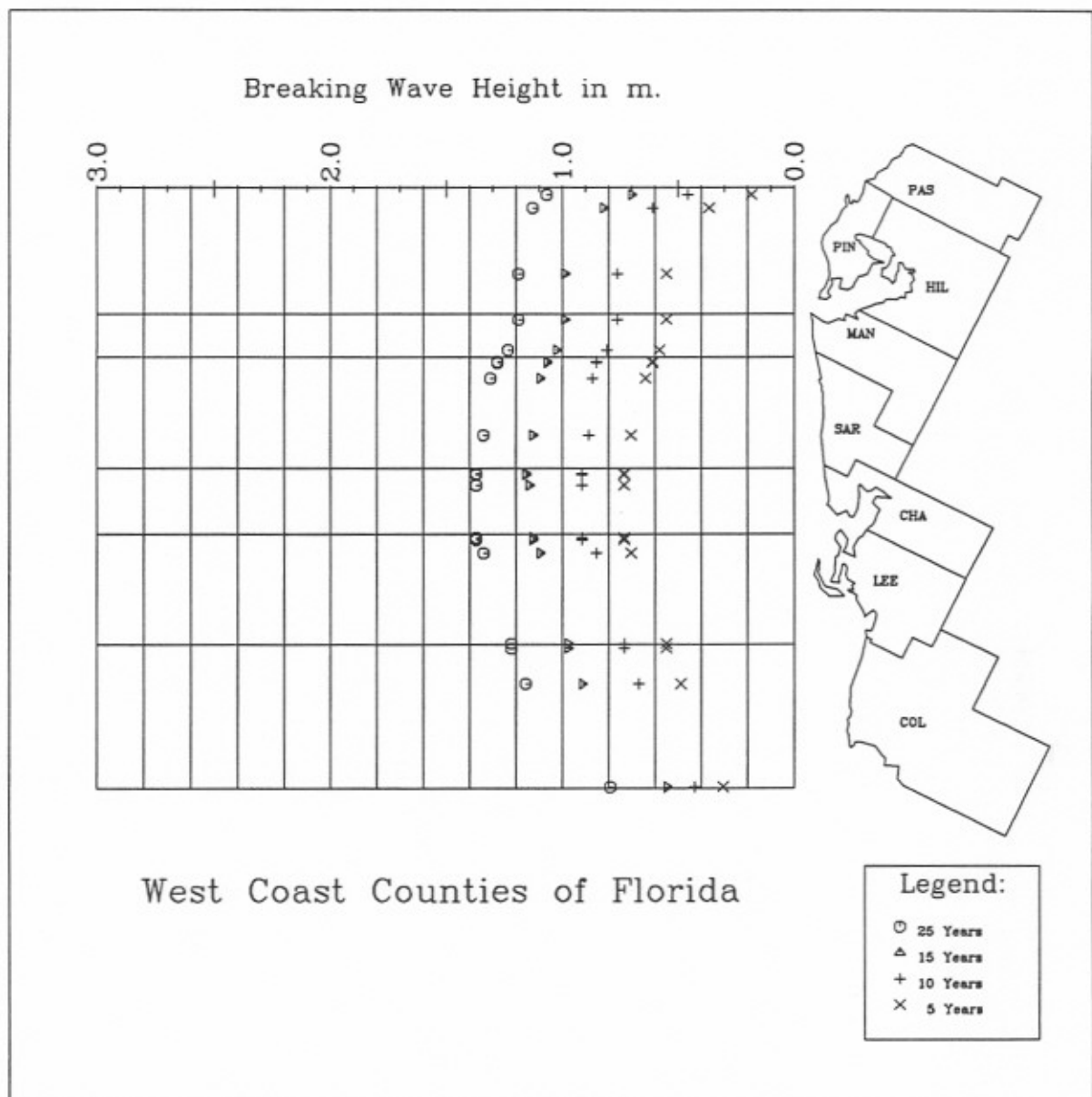


Figure 7. Variation of Equivalent Breaking Wave Height With Return Period Along the East Coast of Florida.

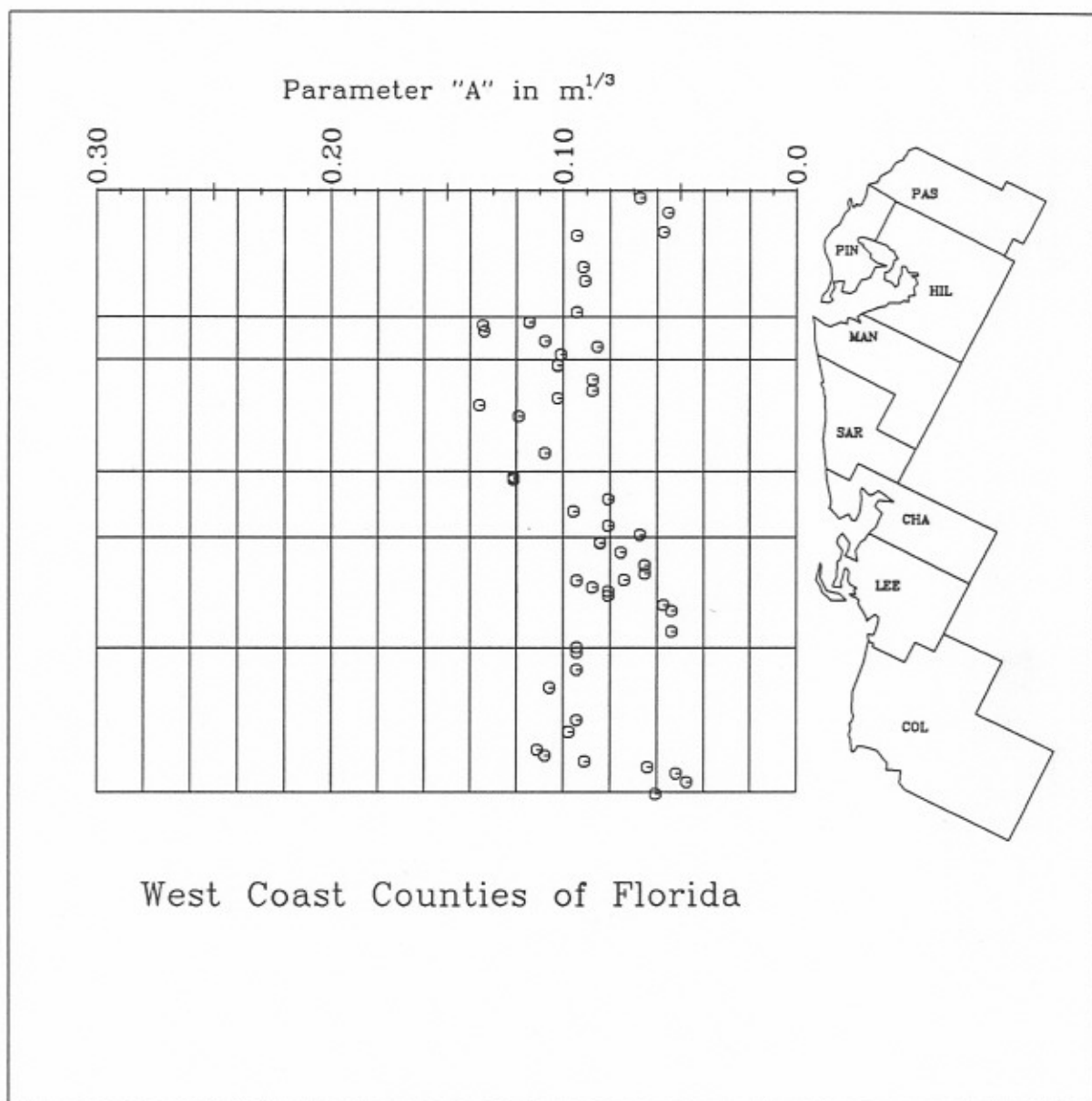


Figure 8. Variation of Scale Parameter "A", Along the West Coast of Florida.

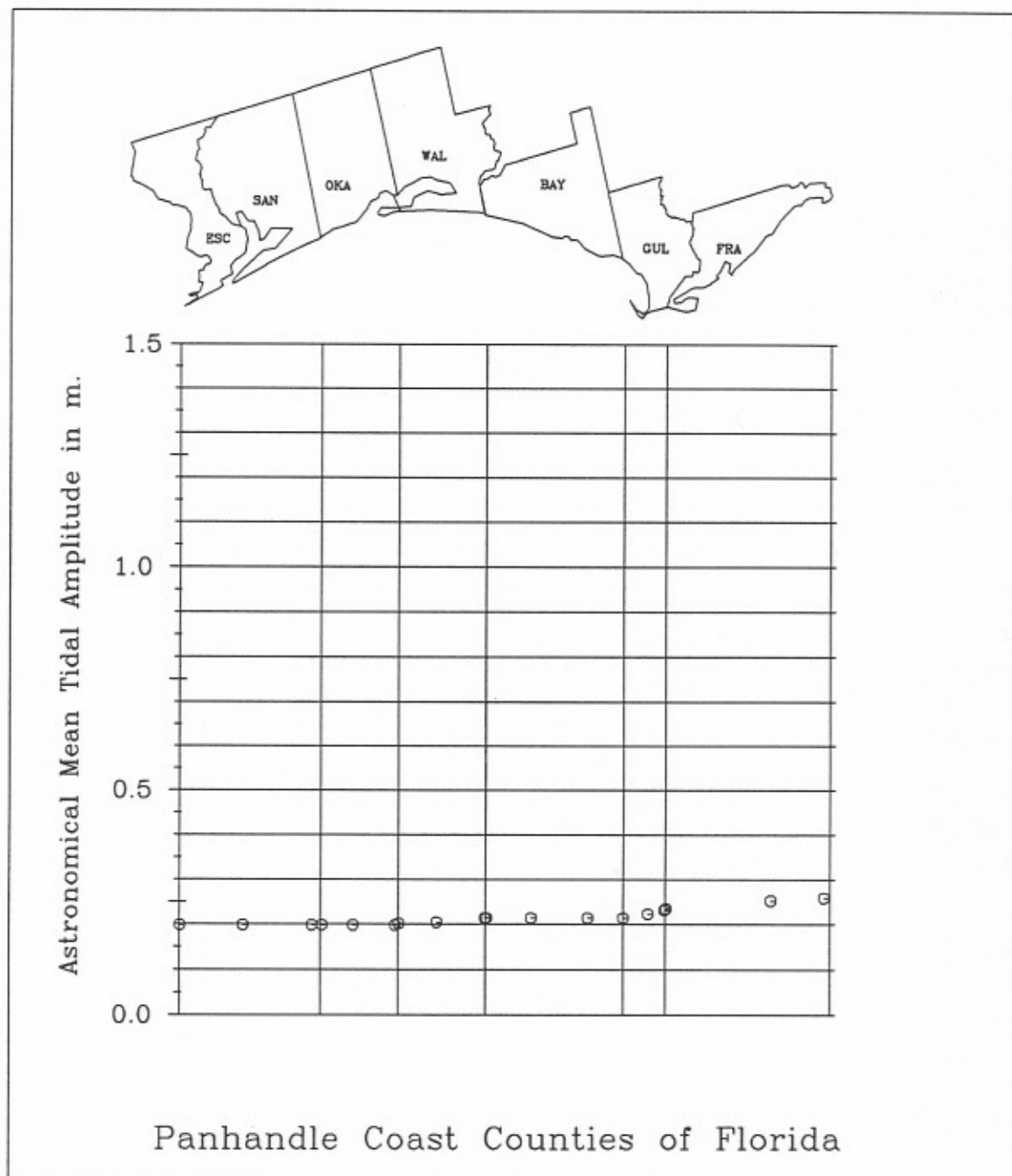


Figure 9. Variation of Mean Astronomical Tidal Amplitude Along the Panhandle Coast of Florida.

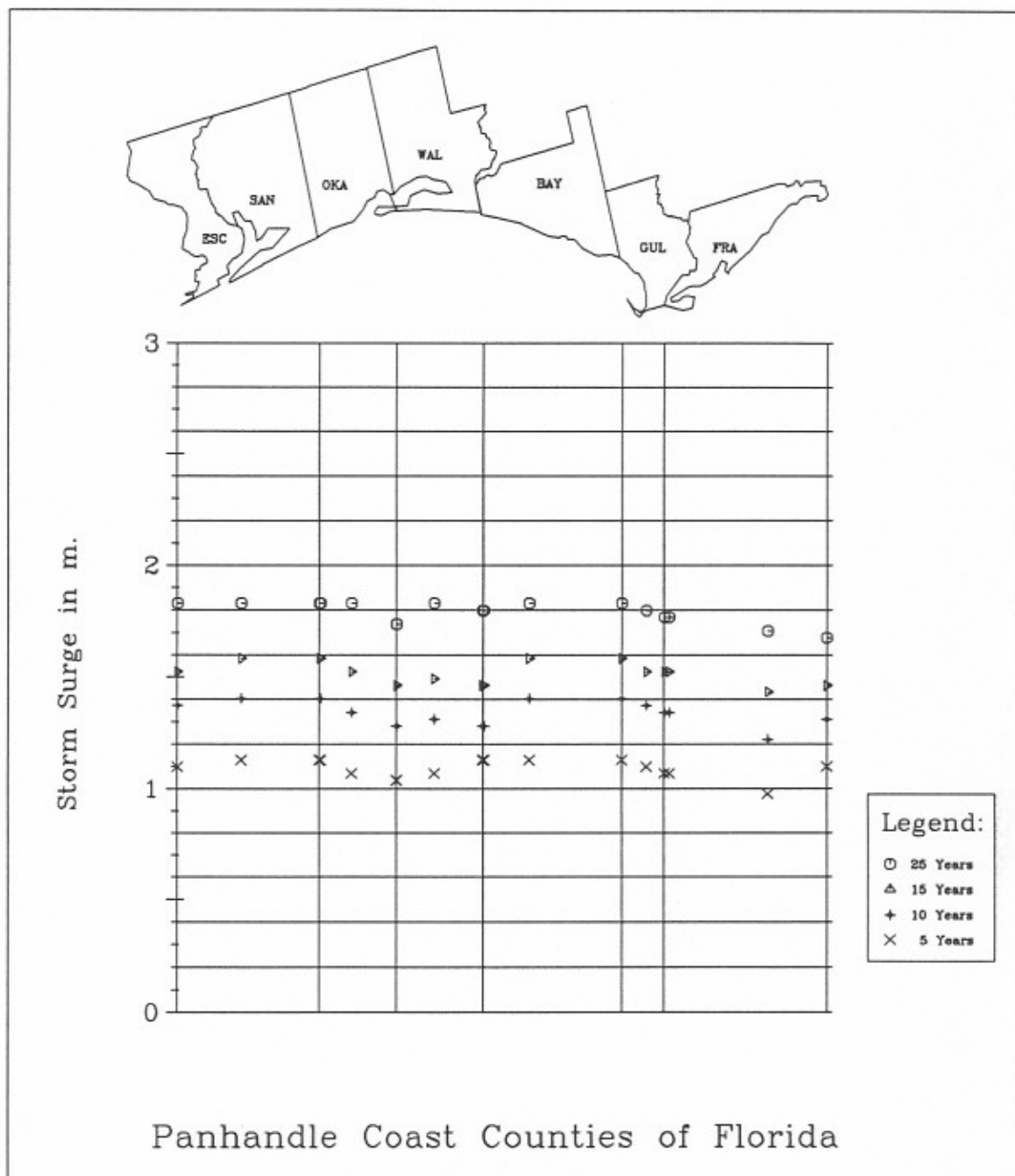


Figure 10. Variation of Storm Surge with Return Period Along the Panhandle Coast of Florida.

3 SAMPLE RUN

3.1 Input Survey Profile File: SURVEY.DAT -(The data are in feet.)

FLAGLER	JUL-AUG 72	CONTROL LINE	(R-35 # POINTS?)
R-1	MAY1972 00 1940066.310	432163.110	70.0014.69
	25JUL72 06AUG72 35 0 17 18		
-100.0	8.32	-50.0	10.30
50.0	14.53	69.0	10.97
250.0	-4.13	300.0	-4.44
500.0	-5.59	550.0	-8.29
989.0	-10.62	1133.0	-11.42
1727.0	-22.62	1868.0	-24.32
2432.0	-31.32	2627.0	-33.42
R-2	MAY1972 00 1939146.500	432510.050	70.0013.81
	25JUL72 11 0 11 0		
-50.0	14.15	0.0	13.79
100.0	7.23	150.0	0.80
350.0	-6.31		
R-3	MAY1972 00 1938183.290	432862.150	70.0014.80
	25JUL72 06AUG72 38 0 16 22		
-50.0	12.84	0.0	14.47
68.0	8.48	73.0	4.75
250.0	-4.93	300.0	-5.53
500.0	-7.52	509.0	-7.53
881.0	-11.73	1016.0	-10.63
1526.0	-20.53	1652.0	-22.83
2162.0	-30.53	2282.0	-32.03
2792.0	-35.03	2927.0	-37.23
R-4	MAY1972 00 1937263.450	433216.890	70.0012.81
	25JUL72 10 0 10 0		
-50.0	11.53	0.0	12.57
100.0	5.19	150.0	-0.56
R-5	MAY1972 00 1936385.690	433570.290	70.0013.69
	25JUL72 10 0 10 0		
-100.0	9.08	-50.0	10.84
50.0	8.25	100.0	3.37
R-6	MAY1972 00 1935440.460	433912.090	70.0012.66
	25JUL72 06AUG72 34 0 11 23		
-50.0	10.96	0.0	12.46
100.0	6.29	150.0	-1.08
350.0	-6.89	369.0	-4.02
810.0	-12.32	921.0	-11.82
1446.0	-20.02	1626.0	-23.12
2106.0	-29.42	2244.0	-31.12
2706.0	-36.42	2844.0	-37.62
R-7	MAY1972 00 1934533.460	434240.950	70.0013.57
	25JUL72 11 0 11 0		
-50.0	11.02	0.0	13.15
100.0	0.79	150.0	-2.23
350.0	-6.07		
R-8	MAY1972 00 1933644.980	434479.820	70.0011.42
	25JUL72 11 0 11 0		
-50.0	9.46	0.0	11.20
150.0	6.27	196.0	0.10
350.0	-4.82		
R-9	MAY1972 00 1932634.130	434882.330	70.0011.85
	25JUL72 06AUG72 33 0 12 21		
-100.0	9.14	-50.0	10.55
80.0	11.69	100.0	8.51
300.0	-5.42	350.0	-5.87
733.0	-11.58	862.0	-10.98

```

1408.0 -17.18 1522.0 -19.48 1714.0 -21.78 1855.0 -23.98 1990.0 -25.88
2104.0 -27.68 2254.0 -29.48 2389.0 -31.48 2509.0 -33.28 2698.0 -34.68
2809.0 -36.28 2959.0 -37.68 3094.0 -38.68
R-10 MAY1972 00 1931703.250 435223.450 70.0011.77
    25JUL72          9 0 9 0

-50.0 10.05 0.0 11.37 50.0 12.58 62.0 13.59 100.0 6.84
150.0 0.29 200.0 -3.06 250.0 -4.00 300.0 -6.38
R-11 MAY1972 00 1930829.530 435543.610 70.0011.37
    25JUL72          11 0 11 0

-100.0 8.27 -50.0 9.56 0.0 11.14 50.0 13.60 54.0 13.04
56.0 11.15 100.0 5.35 150.0 0.24 200.0 -4.05 250.0 -4.50
300.0 -5.15
R-12 MAY1972 00 1929911.990 435877.080 70.0011.50
    25JUL72 10AUG72 30 0 9 21

0.0 11.23 -29.0 10.25 47.0 13.51 50.0 12.63 61.0 11.27
100.0 5.53 150.0 -0.24 200.0 -3.67 250.0 -5.26 467.0 -5.50
566.0 -12.20 650.0 -14.20 734.0 -12.70 857.0 -12.00 998.0 -11.80
1124.0 -14.20 1277.0 -17.20 1403.0 -20.00 1541.0 -22.70 1712.0 -25.20
1844.0 -27.20 1967.0 -29.20 2141.0 -31.30 2282.0 -33.40 2432.0 -35.50
2567.0 -37.50 2702.0 -39.20 2777.0 -40.40 2912.0 -42.00 3062.0 -43.10
R-13 MAY1972 00 1929065.260 436176.250 70.0010.22
    25JUL72          10 0 10 0

-12.0 9.31 0.0 9.98 50.0 13.38 52.0 13.95 55.0 12.67
68.0 11.93 100.0 7.63 150.0 -0.92 200.0 -3.67 250.0 -5.67
R-14 MAY1972 00 1928184.800 436477.550 70.00 8.98
    25JUL72          8 0 8 0

-36.0 6.75 0.0 8.90 45.0 12.82 50.0 11.46 100.0 6.26
150.0 0.04 200.0 -3.37 250.0 -5.42
R-15 MAY1972 00 1927166.190 436829.820 70.00 9.87
    25JUL72 10AUG72 32 0 9 23

0.0 9.63 47.0 11.76 50.0 11.51 88.0 9.98 100.0 8.06
150.0 3.02 200.0 -1.82 250.0 -4.29 300.0 -4.34 456.0 -6.40
555.0 -10.60 677.4 -13.40 708.0 -13.40 771.0 -11.10 813.0 -11.10
948.0 -11.60 1116.0 -13.10 1230.0 -16.10 1389.0 -19.50 1524.0 -22.40
1674.0 -24.60 1851.0 -26.60 1974.0 -29.10 2109.0 -30.60 2304.0 -32.90
2424.0 -34.60 2559.0 -36.10 2709.0 -37.10 2874.0 -37.40 2970.0 -37.50
2994.0 -38.60 3129.0 -40.90

```

3.2 Output File: CCCLa.OUT

1	County	Range	Hb (m.)	Surge (m)	Amp (m)	"A" (m**1/3)	Erosion-@2.4m contour	Return Period
	FLA	R- 1	2.62	1.52	.70	.1080	-21.8m	25 years
	FLA	R- 1	2.02	1.40	.70	.1080	-16.7m	15 years
	FLA	R- 1	1.72	1.28	.70	.1080	-12.7m	10 years
	FLA	R- 1	1.39	1.16	.70	.1080	-8.2m	5 years
	FLA	R- 3	2.62	1.52	.70	.1076	-21.9m	25 years
	FLA	R- 3	2.02	1.40	.70	.1076	-16.4m	15 years
	FLA	R- 3	1.72	1.28	.70	.1076	-13.1m	10 years
	FLA	R- 3	1.40	1.16	.70	.1076	-9.4m	5 years
	FLA	R- 6	2.62	1.52	.70	.1070	-23.7m	25 years
	FLA	R- 6	2.02	1.40	.70	.1070	-16.2m	15 years
	FLA	R- 6	1.72	1.28	.70	.1070	-11.8m	10 years
	FLA	R- 6	1.40	1.16	.70	.1070	-7.3m	5 years
	FLA	R- 9	2.62	1.52	.70	.1064	-23.0m	25 years
	FLA	R- 9	2.02	1.40	.70	.1064	-16.7m	15 years
	FLA	R- 9	1.73	1.28	.70	.1064	-12.5m	10 years

FLA R-	9	1.41	1.16	.70	.1064	-8.0m	5 years
FLA R-	12	2.62	1.52	.70	.1059	-22.8m	25 years
FLA R-	12	2.02	1.40	.70	.1059	-16.0m	15 years
FLA R-	12	1.73	1.28	.70	.1059	-11.7m	10 years
FLA R-	12	1.41	1.16	.70	.1059	-7.2m	5 years
FLA R-	15	2.62	1.52	.69	.1053	-23.3m	25 years
FLA R-	15	2.03	1.40	.69	.1053	-14.5m	15 years
FLA R-	15	1.74	1.28	.69	.1053	-9.4m	10 years
FLA R-	15	1.42	1.16	.69	.1053	-4.8m	5 years

3.3 Output File: SURVEY.OUT

```

Init. Smoothed Prof: FLA R-1      #pts= 40 (Dist & Elev.(-ve) are in meters.)
-30.48      -2.44
-25.25      -2.74
-17.55      -3.05
-12.71      -3.35
-9.09       -3.66
-5.47       -3.96
-1.85       -4.27
 1.61       -4.57
 4.91       -4.88
 9.45       -5.18
12.34       -5.18
13.92       -4.88
14.82       -4.57
16.10       -4.27
17.73       -3.96
19.36       -3.66
20.98       -3.35
25.20       -3.05
29.49       -2.74
32.03       -2.44
34.05       -2.13
36.06       -1.83
38.08       -1.52
40.10       -1.22
42.11       -.91
44.13       -.61
46.40       -.30
49.64       .00
52.89       .30
56.13       .61
59.37       .91
73.11       1.22
105.97      1.52
147.46      1.83
160.36      2.13
166.01      2.44
177.08      2.74
190.38      3.05
234.52      3.35
309.00      3.66

Final Smoothed Prof (Dist & Elev(-ve) in m.): FLA R-1      #pts= 40  -25 year return
period
 8.86      -5.18
 9.01      -4.88
 9.16      -4.57
 9.31      -4.27
 9.47      -3.96
 9.62      -3.66
 9.77      -3.35
 9.92      -3.05
10.08      -2.74
10.23      -2.44
10.38      -2.13

```

10.57	-1.83
14.97	-1.52
20.64	-1.22
26.98	-.91
36.26	-.61
47.50	-.30
60.16	.00
74.04	.30
89.07	.61
105.09	.91
123.01	1.22
139.60	1.52
155.71	1.83
165.29	2.13
172.52	2.44
179.69	2.74
190.46	3.05
234.52	3.35
309.00	3.66
373.17	3.96
390.78	4.27
404.84	4.57
418.42	4.88
432.00	5.18
445.25	5.49
458.28	5.79
471.30	6.10
487.41	6.40
511.48	6.71

Final Smoothed Prof (Dist & Elev(-ve) in m.): FLA R-1
period

#pts= 40 -15 year return

12.34	-5.18
13.92	-4.88
14.25	-4.57
14.40	-4.27
14.56	-3.96
14.71	-3.66
14.86	-3.35
15.01	-3.05
15.17	-2.74
15.32	-2.44
15.47	-2.13
15.63	-1.83
19.33	-1.52
24.47	-1.22
30.31	-.91
38.32	-.61
49.04	-.30
61.19	.00
74.69	.30
87.96	.61
96.20	.91
103.65	1.22
115.73	1.52
147.19	1.83
160.36	2.13
166.01	2.44
177.08	2.74
190.38	3.05
234.52	3.35
309.00	3.66
373.17	3.96
390.78	4.27
404.84	4.57
418.42	4.88
432.00	5.18
445.25	5.49
458.28	5.79
471.30	6.10
487.41	6.40

511.48	6.71	
Final Smoothed Prof (Dist & Elev(-ve) in m.): FLA R-1		#pts= 40 -10 year return
period		
12.34	-5.18	
13.92	-4.88	
14.82	-4.57	
16.10	-4.27	
17.73	-3.96	
18.73	-3.66	
18.88	-3.35	
19.04	-3.05	
19.19	-2.74	
19.34	-2.44	
19.49	-2.13	
19.65	-1.83	
22.40	-1.52	
26.64	-1.22	
31.74	-.91	
38.63	-.61	
48.59	-.30	
60.18	.00	
72.79	.30	
80.22	.61	
86.24	.91	
91.88	1.22	
109.43	1.52	
147.46	1.83	
160.36	2.13	
166.01	2.44	
177.08	2.74	
190.38	3.05	
234.52	3.35	
309.00	3.66	
373.17	3.96	
390.78	4.27	
404.84	4.57	
418.42	4.88	
432.00	5.18	
445.25	5.49	
458.28	5.79	
471.30	6.10	
487.41	6.40	
511.48	6.71	
Final Smoothed Prof (Dist & Elev(-ve) in m.): FLA R-1		#pts= 40 - 5 year return
period		
12.34	-5.18	
13.92	-4.88	
14.82	-4.57	
16.10	-4.27	
17.73	-3.96	
19.36	-3.66	
20.98	-3.35	
23.54	-3.05	
23.72	-2.74	
23.87	-2.44	
24.02	-2.13	
24.17	-1.83	
24.36	-1.52	
28.62	-1.22	
34.09	-.91	
40.63	-.61	
49.66	-.30	
60.20	.00	
67.03	.30	
72.52	.61	
76.94	.91	
82.25	1.22	
105.97	1.52	
147.46	1.83	
160.36	2.13	

166.01	2.44
177.08	2.74
190.38	3.05
234.52	3.35
309.00	3.66
373.17	3.96
390.78	4.27
404.84	4.57
418.42	4.88
432.00	5.18
445.25	5.49
458.28	5.79
471.30	6.10
487.41	6.40
511.48	6.71

APPENDIX A

PROGRAM NOTES:

You may compile the Fortran source code of the CCCLa erosion model using any Fortran compiler that uses the Fortran 77 standard. It has been successfully compiled and run on the VAX running the VMS operating system; it has also been run on a 486 microcomputer with MS DOS version 6.0 and MS Fortran compiler version 5.0.

You can follow the steps below to install the program on a personal computer:

1. First create a subdirectory from the root directory in the hard disk, such as,

[c:>md c:\hfe <return>]
2. Next, copy all the files from the disk provided to the subdirectory in the hard disk, such as,

[c:>copy a:*. * c:\hfe <return>]
3. Although the executable file, CCCLa.exe is included in the disk, it is recommended that you compile and link the source code, CCCLa.for and produce the executable file on your computer system.
4. To run the program on the PC, first copy a file with the profile data in DNR format with the first 3 characters of the county name in capitals on the first line of the file to SURVEY.DAT, then type: CCCLa to start running the program. Output files generated are: CCCLa.OUT and SURVEY.OUT.
5. When CCCLa is running, it will look for these files:

For the East Coast:

enas0594.dat,	eduv0594.dat,	estj0594.dat,
efla0594.dat,	evol0594.dat,	ebre0594.dat,
eind0594.dat,	estl0594.dat,	emar0594.dat,
epal0594.dat,	ebro0594.dat,	edad0594.dat, and
anas0594.dat,	aduv0594.dat,	astj0594.dat,

afla0594.dat, avol0594.dat, abre0594.dat,
aind0594.dat, astl0594.dat, amar0594.dat,
apal0594.dat, abro0594.dat, adad0594.dat.

For the West Coast:

wpin1094.dat, wman1094.dat, wsar1094.dat,
wchal1094.dat, wlee1094.dat, wcol1094.dat, and
apin1094.dat, aman1094.dat, asar1094.dat,
achal1094.dat, alee1094.dat, acol1094.dat.

For the Panhandle Coast:

pesc1094.dat, poka1094.dat, pwal1094.dat,
pbay1094.dat, pgul1094.dat, pfra1094.dat, and
aesc1094.dat, aoka1094.dat, awal1094.dat,
abay1094.dat, agul1094.dat, afra1094.dat.

You should not alter these files.

B-1

APPENDIX B

CCCLa.FOR SOURCE LISTING

```

c-----mai 10
c      +-----+mai 20
c      | CCCLa | This is the application version of the modified CCCLmai 30
c      +-----+ model. This program will compute the high frequencymai 40
c      erosion distances for a given survey profile. It ismai 50
c      assumed that the survey profile will be: along themai 60
c      DEP( formerly DNR) monument,survey bearing, andmai 70
c      DEP data format.mai 80
c      mai 90
c      The dune erosion model was developed and modified formai 100
c      the "High Frequency Erosion Project" by R.G.Dean.mai 110
c      mai 120
c      Changes implemented by S.B.Malakar.mai 130
c      Updated: 11/15/1994mai 131
c      mai 140
c      |-----|mai 150
c      |-< Currently for the East & the West Coasts of Florida only >-|mai 160
c      | (Insufficient Data to Run the Model for the Panhandle Coast)|mai 170
c      |-----|mai 180
c      mai 190
c      Comments:mai 200
c      This model has been run for ALL the available [*.ccl files]mai 210
c      survey profile data for the East & the West coasts of Floridamai 220
c      High frequency erosion distances for a survey profile willmai 230
c      not be computed or printed: if the survey profile depth ismai 240
c      not to at least 12 feet in depth, if the dune elevation ismai 250
c      less than the contour elevation (the berm elevationmai 260
c      determined for the coastal region), and if accretion ismai 270
c      computed.mai 280
c      mai 290
c      parameter (nx=250)mai 300
c      COMMON /A/ H1(nx),XA(nx),HA(nx),NELM,X1(nx),NP,NELM1mai 310
c      dimension sxx(5,nx),syy(5,nx),nsxx(5)mai 320
c      DIMENSION DH(nx), DX(nx), DPTC(nx), XAO(nx),mai 330
c      X(nx,2),NCON(nx),KASAVE(nx),HASAVE(nx)mai 340
c      DIMENSION xbsave(nx), stides(7), hbs(7)mai 350
c      real xread(nx), yread(nx)mai 360
c      dimension jper(4)mai 370
c      mai 380
c      CHARACTER*8 RNGDAT,DOTDAT,BCHDAT,OFFDATmai 390
c      CHARACTER*8 CNTYmai 400
c      character*3 ectys(12)mai 410
c      CHARACTER*3 wctys(6)mai 420
c      character*3 pctys(6)mai 430
c      character*1 rng(8)mai 440
c      character*3 acntymai 450
c      data Ectys/'NAS','DUV','STJ','FLA','VOL','BRE','IND','STL',mai 460
c      'MAR','PAL','BRO','DAD'/mai 470
c      data wctys/'PIN','MAN','SAR','CHA','LEE','COL'/mai 480
c      data pctys/'ESC','OKA','WAL','BAY','GUL','FRA'/mai 490
c      data jper/25,15,10,5/mai 500
c      mai 510
c      Open USER input files: [ There are other files open in themai 520
c      program (data provided). ]mai 530
c      mai 540
c      open(UNIT=4,file='survey.dat', status='unknown')mai 550
c      open(UNIT=5,file='hfe.inp', status='old')mai 560
c      mai 570
c      mai 580
c      Open output file:mai 590

```

```

c
c          unit 6 for erosion computational data
c          open(UNIT=6,file='hfe.out',status='unknown')
c
c          [There is an optional file which may be opened for output of
c          the initial smoothed and final smoothed profiles, survey.out]
C      H(N) VALUES ARE DEPTH VALUES TO CENTER OF ELEMENT.
C      ELEVATIONS ABOVE MEAN SEA LEVEL ARE NEGATIVE.
c
c          pi=3.1416
1234  write(*,*)' '
c      write(*,*)' Enter 1 for British Unit, 2 for Metric Unit'
c      read(*,*)ibm
c      ibm=1
c      if(ibm.lt.1.or.ibm.gt.2)go to 1234
c      if(ibm.eq.2)ftm=3.2808
c      if(ibm.eq.1)ftm=1.0
c      ifct25=0 => Erosion Factor=1.0; ifct25=1 =>Erosion Factor=efact
1235  write(*,*)' Enter Value for Erosion Factor, efect'
c      write(*,*)' (1.0 .ge. efect .le. 10.0):'
c      read(*,*)efact
c      ifct25=0
c      if(efact.gt.1.0)ifct25=1
c      if(efact.lt.1.0.or.efact.gt.10.0)go to 1235
c      efect=1.0
c----- write to screen -----
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
1  '+-----+
c      write(*,*)' '
2  '| Note:          Output is written to file:  HFE.OUT
c      write(*,*)' '
3  '|          Init. & final Smoothed profile to file: SURVEY.OUT
c      write(*,*)' '
4  '|
c      write(*,*)' '
5  '|          Erosion Factor=',efact,'
c      write(*,*)' '
6  '|          British Units in Effect
c      write(*,*)' '
7  '|
c      write(*,*)' '
8  '|          ("ero"=-9999.9 indicates accretion for profile)
c      write(*,*)' '
9  '+-----+
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      write(*,*)' '
c      pause
c      dcut= a profile should be surveyed to at least "dcut" depth
c      for erosion to computed

```

```

mai 600
mai 610
mai 620
mai 630
mai 640
mai 650
mai 660
mai 670
mai 680
mai 690
mai 700
mai 710
mai 720
mai 730
mai 740
mai 750
mai 760
mai 770
mai 780
mai 790
mai 800
mai 810
mai 820
mai 830
mai 840
mai 850
mai 860
mai 870
mai 880
mai 890
mai 900
mai 910
mai 920
mai 930
mai 940
mai 950
mai 960
mai 970
mai 980
mai 990
mai1000
mai1010
mai1020
mai1030
mai1040
mai1050
mai1060
mai1070
mai1080
mai1090
mai1100
mai1110
mai1120
mai1130
mai1140
mai1150
mai1160
mai1170
mai1180
mai1190
mai1200
mai1210
mai1220
mai1230
mai1240
mai1250

```

	dcut=-12.0	mail1260
	iout=0	mail1270
C		mail1280
C	Read erosion model parameters:	mail1290
C	READ(5,33) CNTY	mail1300
11	FORMAT(10X,F10.4)	mail1310
C	READ(5,11) XK	mail1320
C	READ(5,11) XMD	mail1330
C	READ(5,22) IFCT25	mail1340
22	FORMAT(10X,I10)	mail1350
C		mail1360
C	dt=time increment in hr for erosion computation;	mail1370
C	clev=contour elev at which erosion was computed in ft;	mail1380
C	dero=duration of model run in hrs, used for tidal period,	mail1390
C	values used 18.6 hr for East & West coasts, 12.4 hr for	mail1400
C	Panhandle; for area specify: E or e for East coast,	mail1410
C	W or w for West coast, P or p for Panhandle	mail1420
C		mail1430
C	READ(5,11) dt	mail1440
C	read(5,11) clev	mail1450
C	read(5,11) dero	mail1460
C	read(5,31) area	mail1470
31	format(10x,a1)	mail1480
C	if iandf=1, then output initial & final smoothed profiles	mail1490
C	read(5,22) iandf	mail1500
C	close(unit=5)	mail1510
C	Initialize erosion parameters:	mail1520
	xk=0.075	mail1530
	xmd=2.00	mail1540
	dt=0.05	mail1550
	dero=18.6	mail1560
	iandf=1	mail1570
C		mail1580
C	read survey profile header data:	mail1590
	read(4,32,end=99999) acnty	mail1600
32	format(a3)	mail1610
	backspace(unit=4)	mail1620
	do 18 i=1,12	mail1630
	if(ectys(i).eq.acnty) area='e'	mail1640
	if(ectys(i).eq.acnty) go to 19	mail1650
	if(i.lt.7.and.wctys(i).eq.acnty) area='w'	mail1660
	if(i.lt.7.and.wctys(i).eq.acnty) go to 19	mail1670
	if(i.lt.7.and.pctys(i).eq.acnty) area='p'	mail1680
	if(i.lt.7.and.pctys(i).eq.acnty) go to 19	mail1690
18	continue	mail1700
	write(*,188)	mail1710
188	format(' *** Error: First 3 characters in line 1 of survey',	mail1720
1	lx,'data not county ID or not in CAPs')	mail1730
	stop	mail1740
		mail1750
		mail1760
C	area='e'	mail1770
19	if(iandf.eq.1) open(unit=9,file='survey.out',status='unknown')	mail1780
C		mail1790
	timesn=dero/dt	mail1800
	ntimes=int(timesn)	mail1810
C		mail1820
	if(area.eq.'E'.or.area.eq.'e'.or.area.eq.'W'.or.area.eq.'w')	mail1830
1	tinc=3.0*pi/timesn	mail1840
C		mail1850
	if(area.eq.'P'.or.area.eq.'p') tinc=pi/timesn	mail1860
C	write(*,333) dt,dero,ntimes,tinc,area	mail1870
333	format(' dt dero ntimes tinc area ... HFE.inp'	mail1880
1	/' ',f7.3,f7.1,i5,f8.3,2x,a1)	mail1890
C		mail1900
		mail1910

	READ(4,33,end=99999) cnty	mai1920
33	format(a8)	mai1930
c	write(*,*) ' acnty,cnty=',acnty,' ',cnty	mai1940
c	##### This is the profile loop, the outermost loop #####	mai1950
1	READ(4,44,END=99999) RNG,RNGDAT,ICODE,YNORTH,XEAST,AZMUTH	mai1960
	READ(4,55)DOTDAT,BCHDAT,OFFDAT,NP,NPDOT,NPBCH,NPOFF	mai1970
44	FORMAT(8a1,A8,I2,2F12.3,F7.2)	mai1980
55	FORMAT(A8,A8,A8,4I3,/))	mai1990
c		mai2000
c		mai2010
c	extract range number irang from rng	mai2020
c	call range(rng,irang)	mai2030
c		mai2040
c	write(*,*) ' range#=',irang	mai2050
c	get high frequency erosion data	mai2060
c	call hfe(acnty,irang,aval,amp,kounty,stides,hbs)	mai2070
	if(kounty.le.4) clev=-8.0	mai2080
	if(kounty.ge.5.and.kounty.le.12) clev=-7.0	mai2090
	if(kounty.ge.13.and.kounty.le.20) clev=-5.0	mai2100
	if(kounty.ge.21.and.kounty.le.24) clev=-6.0	mai2110
	write(*,63)	mai2120
63	format(' ')	mai2130
c	write(*,64) acnty,irang,aval,amp,(stides(i),i=1,4),	mai2140
c	(hbs(j),j=1,4)	mai2150
c64	format(' hfe: ',a3,' R-',i3,' a=',f5.3,' amp=',f3.1,' stide:',	mai2160
c	1 4f5.1,' Hb:',4f5.1)	mai2170
	a=aval	mai2180
c		mai2190
c		mai2200
c	read survey profile data:	mai2210
	READ(4,66,END=99999) (Xread(I),yread(I),I=1,NP)	mai2220
66	format(5(f7.1,1x,f7.2))	mai2230
c	screen survey profile:	mai2240
	idcut=0	mai2250
	idunee=0	mai2260
	do 170 i=1,np	mai2270
	if(yread(i).ge.-clev) idunee=1	mai2280
	if(yread(i).le.dcut) idcut=1	mai2290
170	continue	mai2300
	if(idunee.eq.0.or.idcut.eq.0) go to 1	mai2310
c		mai2320
c		mai2330
c	----- start loop irank -----	mai2340
	do 450 irank=1,4	mai2350
c		mai2360
	stval=stides(irank)	mai2370
	hbi=hbs(irank)	mai2380
c	write(*,*) ' irank,hbi(irank)=' ,irank,hbi	mai2390
c		mai2400
	DO 20 I=1,nx	mai2410
	X(I,1)=0.0	mai2420
	X(I,2)=0.0	mai2430
	X1(I)=0.0	mai2440
	H1(I)=0.0	mai2450
	NCON(I)=0	mai2460
	dh(i)=0.0	mai2470
	dx(i)=0.0	mai2480
	dptc(i)=0.0	mai2490
	xao(i)=0.0	mai2500
	xasave(i)=0.0	mai2510
	hasave(i)=0.0	mai2520
	xbsave(i)=0.0	mai2530
	XA(I)=0.0	mai2540
20	HA(I)=0.0	mai2550
c		mai2560
c	transfer original survey profile data to x1 & h1:	mai2570

	i20=1	mai2580
	DO 40 I=1,NP	mai2590
	x1(i)=xread(i)	mai2600
	h1(i)=yread(i)	mai2610
	IF (i20.eq.1.and.yread(I).LT.dcut) nstop=i	mai2620
	if(yread(i).lt.dcut)i20=0	mai2630
40	CONTINUE	mai2640
	if(i20.eq.1)NSTOP=I-1	mai2650
C		mai2660
	IDUNEM = 1	mai2670
	DO 50 I = 1,NP	mai2680
	IF (H1(I) .GT. H1(IDUNEM)) IDUNEM = I	mai2690
50	CONTINUE	mai2700
	MIN1=1	mai2710
	DO 60 I = 1,IDUNEM	mai2720
	IF (H1(I) .LT. H1(MIN1)) MIN1 = I	mai2730
60	CONTINUE	mai2740
	MIN2=1	mai2750
	DO 70 I = IDUNEM,NP	mai2760
	IF (H1(I) .LT. H1(MIN2)) MIN2 = I	mai2770
70	CONTINUE	mai2780
	IMAX=IFIX (H1 (IDUNEM)+0.2)	mai2790
	IMIN1=IFIX (H1 (MIN1))	mai2800
	IMIN2=IFIX (H1 (MIN2))	mai2810
	NELM1=IMAX-IMIN1+1	mai2820
	NELM2=IMAX-IMIN2+1	mai2830
	NELM=NELM1+NELM2	mai2840
	NL=NELM	mai2850
	II=1	mai2860
	IMAXM1=IMAX-1	mai2870
	DO 80 I=IMIN1,IMAX	mai2880
	HA(II)=I	mai2890
	II = II + 1	mai2900
80	CONTINUE	mai2910
	DO 90 I=IMAX,IMIN2,-1	mai2920
	HA(II)=I	mai2930
	II = II + 1	mai2940
90	CONTINUE	mai2950
	DO 100 I=1,NELM	mai2960
	NCON(I)=1	mai2970
100	XA(I)=0.0	mai2980
	CALL SMOOTH(IDUNEM,1,2)	mai2990
	CALL SMOOTH(IDUNEM,NP,1)	mai3000
	NELM1M = NELM1 - 1	mai3010
	N2=NELM1+1	mai3020
	N3=NELM-1	mai3030
	DO 110 M=1,NELM1M	mai3040
	XASAVE(M) = -XA(M) + X1(IDUNEM)	mai3050
	HASAVE(M)=-HA(M)	mai3060
110	CONTINUE	mai3070
	XASAVE(NELM1) = X1(IDUNEM)	mai3080
	HASAVE(NELM1) = -HA(NELM1)	mai3090
	DO 120 M=N2,NELM	mai3100
	XASAVE(M) = XA(M) + X1(IDUNEM)	mai3110
	HASAVE(M)=-HA(M)	mai3120
120	CONTINUE	mai3130
		mai3140
		mai3150
	DO 130 I=1,NL	mai3160
	IF (HASAVE(I).GT.-dcut) GO TO 135	mai3170
130	CONTINUE	mai3180
135	NSTOP=I-1	mai3190
		mai3200
C	-----mai3210	
C	(xasave(i),hasave(i),i=1,nstop) contain initial smoothed profile	mai3220
C	data	mai3230

c	write to unit 9:*****	mai3240
c	if(irank.eq.1.and.iandf.eq.1.and.ibm.eq.2)	mai3250
c	1 write(9,122)acnty, rng, nstop	mai3260
c122	format(' Init.Smoothed Prof: ',a3,1x,8a1,' #pts=',i3,	mai3270
c	1 '(Dist & Elev.(+ve) are in meters.)')	mai3280
c	if(irank.eq.1.and.iandf.eq.1.and.ibm.eq.1)	mai3290
c	1 write(9,123)acnty, rng, nstop	mai3300
c123	format(' Init. Smoothed Prof: ',a3,1x,8a1,' #pts=',i3,	mai3310
c	1 '(Dist & Elev.(+ve) are in ft.)')	mai3320
	if(irank.ne.1)go to 126	mai3330
	if(iandf.ne.1)go to 126	mai3340
	nsxx(1)=nstop	mai3350
	do 125 i=1,nstop	mai3360
c	*** write array:	mai3370
	xasm=xasave(i)/ftm	mai3380
	yasm=-hasave(i)/ftm	mai3390
	sxx(1,i)=xasm	mai3400
	syy(1,i)=yasm	mai3410
c	write(9,124)xasm,yasm	mai3420
124	format(' ',2f10.2)	mai3430
125	continue	mai3440
126	hastop=hasave(nstop)	mai3450
c	-----	mai3460
	DO 140 M=1,NELM1	mai3470
	I=NELM1-M+1	mai3480
	NCON(I)=2	mai3490
	XA(I) = -XA(I) + X1(IDUNEM)	mai3500
	X(M,1)=XA(I)	mai3510
140	CONTINUE	mai3520
	DO 150 M=N2,NELM	mai3530
	I=M-NELM1	mai3540
	XA(M) = XA(M) + X1(IDUNEM)	mai3550
	X(I,2)=XA(M)	mai3560
	HA(I)=-HA(M)	mai3570
	NLS=I	mai3580
150	CONTINUE	mai3590
C-----	INITIALIZE 'ACTIVE' PROFILE	mai3600
	NL=NLS	mai3610
	X(NELM1,1)=-1000.0	mai3620
	DO 160 I=1,NL	mai3630
160	XA(I)=X(I,2)	mai3640
		mai3650
		mai3660
		mai3670
		mai3680
C		mai3690
C-----	CALCULATE HSTAR TO NEAREST FOOT	mai3700
	DO 190 I = 1,NL	mai3710
	XAO(I) = XA(I)	mai3720
190	CONTINUE	mai3730
	HS = (0.667 * A**1.5 / XMD)**2	mai3740
	DXS = (HS / A)**1.5	mai3750
	HSTAR = HS	mai3760
C		mai3770
	XR = 0.0	mai3780
	DH(1) = 1.011	mai3790
C		mai3800
C	ESTABLISH INSTANTANEOUS WATER LEVEL, ST	mai3810
C		mai3820
	PIO2 = 1.5708	mai3830
		mai3840
		mai3850
	DO 200 I = 2,NL	mai3860
	DH(I) = (HA(I) - HA(I-1) - DH(I-1)/2.0)*2.0	mai3870
200	CONTINUE	mai3880
C		mai3890

	ZZ=1.0-EXP(-XK*DT)	mai3900
	IBP=1	mai3910
	NPA=1	mai3920
C	dtime=0.02534 ! 3 min	mai3930
	dtimen=-tinc	mai3940
	DO 320 NTIME = 1,NTIMES	mai3950
	dtimen=dtimen+tinc	mai3960
	tidev=amp*sin(dtimen)	mai3970
	TIME = (NTIME-1)*DT	mai3980
C		mai3990
C		mai4000
	ST = stval+tidev	mai4010
	DO 230 I = 1,NL	mai4020
	DPTC(I) = HA(I) + ST	mai4030
230	CONTINUE	mai4040
C		mai4050
C		mai4060
	hb=hbi	mai4070
	DPTB = hb/0.8	mai4080
	XB = abs(DPTB / A) ** 1.5	mai4090
C		mai4100
C	ESTABLISH THE INDICES OF STILL WATER LEVEL AND HSTAR	mai4110
C		mai4120
	IE = 1	mai4130
	AA = 100.0	mai4140
	DO 240 I = 1,NL	mai4150
	BB = ABS(DPTC(I) - DPTB)	mai4160
	IF (BB .GT. AA) GO TO 235	mai4170
	AA = BB	mai4180
	IE = I	mai4190
235	IF (ABS(DPTC(I)) .LE. DH(I)/2) IWL = I	mai4200
	IF (ABS(HSTAR - DPTC(I)) .LE. DH(I)/2) ISTAR = I	mai4210
240	CONTINUE	mai4220
	IB=NPA	mai4230
	xr=0.0	mai4240
	DO 290 IIT = 1,10	mai4250
	SUM1 = 0.0	mai4260
	SUM2 = 0.0	mai4270
	SUM3 = 0.0	mai4280
	SUM4 = 0.0	mai4290
	DO 250 I = IB,ISTAR	mai4300
	SUM1 = SUM1 + (HA(I) + ST - HSTAR) * DH(I) / XMD +	mai4310
	1 (DXS+xr) * DH(I)	mai4320
	SUM4 = SUM4 + DH(I)	mai4330
	SUM3 = SUM3 + XA(I) * DH(I)	mai4340
250	CONTINUE	mai4350
	ISP = ISTAR + 1	mai4360
	DO 260 I = ISP, IE	mai4370
	SUM4 = SUM4 + DH(I)	mai4380
	if(dptc(i).gt.0.0)	mai4390
	1 SUM2 = SUM2 + (DPTC(I)/A)**1.5 * DH(I) +dh(i)+xr*dh(i)	mai4400
	SUM3 = SUM3 + XA(I) * DH(I)	mai4410
260	CONTINUE	mai4420
	IEP = IE + 1	mai4430
	DO 270 I = IEP,NL	mai4440
	IFILL = I-1	mai4450
	IF (XR + XB .LT. XA(I)) GO TO 275	mai4460
	SUM3 = SUM3 + XA(I) * DH(I)	mai4470
	SUM4 = SUM4 + DH(I)	mai4480
	SUM2 = SUM2 + XB * DH(I)+xr*dh(i)	mai4490
	IFILL = I	mai4500
270	CONTINUE	mai4510
275	CONTINUE	mai4520
	XRO = XR	mai4530
	dxr=0.5/sum4*(sum3-sum2-sum1)	mai4540
	xr=xr+dxr	mai4550

```

C      XR = 1.0/SUM4 * (SUM3 - SUM2 - SUM1)
C+++++
C      ESTABLISH NEW VALUE OF IB
C      IBO = IB
C      BSTAR=XR+DXS-XA(ISTAR)+(DPTC(ISTAR)-HSTAR)/XMD
C      BSTAR=BSTAR*ZZ
C      XSTAR=XA(ISTAR)+BSTAR
C      IB=NPA
C      DO 280 I = NPA,IE
C      BB=XSTAR+(DPTC(I)-HSTAR)/XMD-XA(I)
C      IF (BB.LT.0.0) GO TO 285
C      IB=I+1
280  CONTINUE
285  CONTINUE
C      IF (IB.LT.1) IB=1
C      IF (IB.GT.IWL) IB = IWL
C      IF ((IB.EQ. IBO) .AND. (ABS(XRO - XR) .LE. .01)) GO TO 295
C+++++
290  CONTINUE
295  CONTINUE
C
C      CALCULATE DX VALUES
C
C      DO 300 I = IB,IFILL
C      IF (I .GT. ISTAR .AND. I .LE. IE) GO TO 292
C      IF (I .GT. IE) GO TO 294
C      DX(I) = XA(I) - (XR + DXS + (HA(I) + ST-HSTAR) / XMD)
C      GO TO 300
292  DX(I) = XA(I) - (XR + ((HA(I) + ST)/A)**1.5)
C      GO TO 300
294  DX(I) = XA(I) - (XR + XB)
300  CONTINUE
C      BA = (1.0 - EXP(-XK*DT))
C      DO 310 I = IB,IFILL
C      BB = -DX(I) * BA
C      XA(I) = XA(I) + BB
C      IF (NCON(NPA) .EQ. 2 .AND. XA(NPA) .LT. X(NPA,1)) NPA=NPA+1
310  CONTINUE
C      xmoft=0.08
C      call avala(xmd,xmoft,nl,iwl)
320  CONTINUE
C
C *** END OF TIME LOOP ***
C
C -----
C      (xa(i),ha(i),i=npa,nstop) final smoothed profile data
C      nstop1=nstop-npa+1
C      if(iandf.eq.1.and.ibm.eq.2)
C      1 write(9,322)acnty,rng,nstop1,jper(irank)
C322  format(' Final Smoothed Prof (Dist & Elev(+ve) in m.): ',
C      1 a3,1x,8a1,' #pts=',i3,' -',i2,' year return period')
C      if(iandf.eq.1.and.ibm.eq.1)
C      1 write(9,323)acnty,rng,nstop1,jper(irank)
C323  format(' Final Smoothed Prof (Dist & Elev(+ve) in ft): ',
C      1 a3,1x,8a1,' #pts=',i3,' -',i2,' year return period')
C      if(iandf.ne.1)go to 329
C      iss=0
C      do 328 i=npa,nstop
C      iss=iss+1

```

xams=xa(i)/ftm	mai5220
hams=-ha(i)/ftm	mai5230
sxx(irank+1,iss)=xams	mai5240
syy(irank+1,iss)=hams	mai5250
c if(ha(i).gt.hastop)go to 329	mai5260
c if((i+1.le.nstop).and.(xa(i+1).lt.xa(i)))go to 329	mai5270
c write(9,124)xams,hams	mai5280
c -----	mai5290
C	mai5300
328 continue	mai5310
nsxx(irank+1)=iss	mai5320
329 continue	mai5330
DO 330 I = 1,NL	mai5340
DX(I) = XA(I) - XAO(I)	mai5350
330 CONTINUE	mai5360
IF (IFCT25.EQ.0) GO TO 374	mai5370
c =====added 11/3/1994 ===== do loop to comments #1	mai5380
DO 340 I=NPA,NL	mai5390
IF (DX(I).GT.0.0) GO TO 374	mai5400
XA(I)=XA(I)+(efact-1.0)*DX(I)	mai5410
DXT=DX(I)*efact	mai5420
340 continue	mai5430
c ===== start of "commenting" #1 =====	mai5440
c NPA2=NPA	mai5450
c IE=0	mai5460
c DO 340 I=NPA,NL	mai5470
c IE=I	mai5480
c IF (DX(I).GT.0.0) GO TO 342	mai5490
c XA(I)=XA(I)+(efact-1.0)*DX(I)	mai5500
c DXT=DX(I)*efact	mai5510
c write(1,1111)i,dx(i),dxt,efact	mai5520
c1111 format(' main1: i,dx(i),dxt,efact=',i3,3f10.2)	mai5530
c NB=I	mai5540
c IF (I.GT.NELM1) GO TO 340	mai5550
c IF (XA(I).GT.X(I,1)) GO TO 340	mai5560
c XA(I)=X(I,1)	mai5570
c DXT=X(I,1)-X(I,2)	mai5580
c NPA2=I	mai5590
c 340 DX(I)=DXT	mai5600
c 342 CONTINUE	mai5610
cc write(1,2222)i,dx(i),dxt,efact	mai5620
c2222 format(' main2: i,dx(i),dxt,efact=',i3,3f10.2)	mai5630
c	mai5640
c NPA=NPA2	mai5650
c IEM1=IE-1	mai5660
c IO=0	mai5670
c XMIN=10000.0	mai5680
c DO 350 I=NPA,NB	mai5690
c IF (XA(I).GT.XMIN) GO TO 350	mai5700
c IF (DX(I).EQ.0.0) GO TO 350	mai5710
c XMIN=XA(I)	mai5720
c IO=I	mai5730
c350 CONTINUE	mai5740
c SUM=0.0	mai5750
c IF (IO.EQ.0) GO TO 372	mai5760
c IBEG=0	mai5770
c DO 360 I=1,IO	mai5780
c XT=XA(IO) - (HA(IO)-HA(I))/XMD	mai5790
c IF (XT.GT.XA(I)) GO TO 360	mai5800
c IF (IBEG.EQ.0) IBEG=I	mai5810
c SUM=SUM+XA(I)-XT	mai5820
c XA(I)=XT	mai5830
c 360 CONTINUE	mai5840
c II=0	mai5850
c IF (II.EQ.0) GO TO 374	mai5860
c IF (IBEG.EQ.0) GO TO 372	mai5870

c	DZ=HA(IE)-HA(IEG)	mai5880
c	YMAX=1.5*SUM/DZ	mai5890
c	DO 370 I=IEG,IE	mai5900
c	AA=(HA(IE)-HA(I))/DZ	mai5910
c	XT=XA(I)	mai5920
c	IF (I.LT.IO) XT=XA(IO) - (HA(IO)-HA(I))/XMD	mai5930
c 370	XA(I)=XT + 4.0*YMAX*(AA-AA**2)	mai5940
c 372	CONTINUE	mai5950
c	===== end of commenting #1 =====	mai5960
374	CONTINUE	mai5970
	DO 380 I = 1,NL	mai5980
	DX(I) = XA(I) - XAO(I)	mai5990
c	write(1,3333)i,dx(i)	mai6000
3333	format(' main3: i,dx(i)=' ,i3,f10.2)	mai6010
380	CONTINUE	mai6020
	NPA2=NPA	mai6030
	DO 390 I=NPA,NL	mai6040
	IF (DX(I).GT.0.0) GO TO 395	mai6050
	IF (I.GT.NELM1.OR.XA(I).GT.X(I,1)) GO TO 390	mai6060
	NPA2=I	mai6070
390	CONTINUE	mai6080
395	CHK=XA(NPA2)-X(NPA2,1)	mai6090
	IF (CHK.LT.-10.0) NPA2=NPA2+1	mai6100
	NPA=NPA2	mai6110
397	S1=0.0	mai6120
	NPAM=NPA-1	mai6130
	IF (NPA.EQ.1) GO TO 402	mai6140
	DO 400 I=1,NPAM	mai6150
	IF (HA(I).GT.0.0) GO TO 412	mai6160
400	S1=S1+(X(I,2)-X(I,1))	mai6170
402	DO 410 I=NPA,NL	mai6180
	AA=X(I,2)-XA(I)	mai6190
	IF (HA(I).GT.0.0) GO TO 412	mai6200
	IF (AA.LT.0.0) GO TO 412	mai6210
410	S1=S1+AA	mai6220
412	CONTINUE	mai6230
	DO 420 I=1,NL	mai6240
	IF (HA(I).EQ.Clev) GO TO 425	mai6250
420	CONTINUE	mai6260
425	NRNG=I	mai6270
C	-----	mai6280
	DO 430 IJK = NPA,NL	mai6290
	dxijk=dx(ijk)	mai6300
c	write(1,4444)i,dx(ijk)	mai6310
4444	format(' main4: ijk,dx(ijk)=' ,i3,f10.2)	mai6320
		mai6330
c	write(*,*)' dxijk,ha(ijk)=' ,dxijk,ha(ijk),ijk	mai6340
	if(ha(ijk).eq.clev) go to 432	mai6350
430	CONTINUE	mai6360
	write(*,431)acnty,rng(1),rng(2),irang,clev	mai6370
431	format(' ',a3,1x,2a1,i3,	mai6380
1	' no erosion computed for contour elev=' ,f7.2)	mai6390
	iout=iout+1	mai6400
c	if(mod(iout,4).eq.1)write(6,882)	mai6410
	hbim=hbi/ftm	mai6420
	stvalm=stval/ftm	mai6430
	ampm=amp/ftm	mai6440
	avalm=aval**3	mai6450
	avalm=(avalm/ftm)**0.3333	mai6460
	dxijkm=dxijk/ftm	mai6470
	if(dxijkm.gt.0.0)dxijkm=-9999.9	mai6480
c	write(6,818)acnty,rng(1),rng(2),irang,stvalm,ampm,avalm,	mai6490
c	1 jper(irank)	mai6500
818	format(' ',10x,a3,1x,2a1,i3,6x,F6.2,f5.2,f7.4,	mai6510
1	18x,i3,' years')	mai6520
		mai6530

```

go to 616
c
432 if(hbi.eq.0.0.and.ibm.eq.1)
1 write(*,433)acnty,rng(1),rng(2),irang,clev
433 format(' ',a3,1x,2a1,i3,
1 ' no erosion computed for contour elev=',f7.2,'ft')

hbm=hbi/ftm
stvalm=stval/ftm
ampm=amp/ftm
avalm=aval**3
avalm=(avalm/ftm)**0.3333
dxijkm=dxijk/ftm
if(dxijkm.gt.0.0)dxijkm=-9999.9
if(hbi.eq.0.0.and.ibm.eq.2)
434 1 write(*,434)acnty,rng(1),rng(2),irang,clev
format(' ',a3,1x,2a1,i3,
1 ' no erosion computed for contour elev=',f7.2,'m')
if(hbi.ne.0.0.and.ibm.eq.1)
1 write(*,87)acnty,rng(1),rng(2),irang,hbi,stval,amp,aval,
1 dxijkm,jper(irank)
87 format(' ',a3,1x,2a1,i3,' hb=',F4.1,'ft s.surge=',F3.1,
1 'ft ampm=',f4.2,'ft a=',f4.3,'ft**1/3 ero=',F7.1,'ft ',
2 i2,' yrs')
if(hbi.ne.0.0.and.ibm.eq.2)
1 write(*,871)acnty,rng(1),rng(2),irang,hbm,stvalm,ampm,avalm,
1 dxijkm,jper(irank)
871 format(' ',a3,1x,2a1,i3,' hb=',F4.1,'m s.surge=',F3.1,
1 'm ampm=',f4.2,'m a=',f4.3,'m**1/3 ero=',F7.1,'m ',
2 i2,' yrs')

C -----
clevl=-clev+0.1
levc=int(clevl)
iout=iout+1
alevc=-clev/ftm
if(mod(iout,999).eq.1.and.ibm.eq.2)write(6,881)alevc
881 format('1',7x,
1 'County Range Hb Surge Amp "A" Erosion-@',f3.1,
2 'm Dist Return Period',
2 '/' ,21x,'(m.) (m) (m) (m**1/3)',4x,'contour From Mon')

if(mod(iout,999).eq.1.and.ibm.eq.1)write(6,884)alevc
884 format('1',7x,
1 'County Range Hb Surge Amp "A" Erosion-@',f3.1,
2 'ft Dist Return Period',
2 '/' ,21x,'(ft) (ft) (ft) (ft**1/3)',3x,'contour From Mon')
c if(mod(iout,4).eq.1)write(6,882)
882 format(' ')
hbm=hbi/ftm
stvalm=stval/ftm
ampm=amp/ftm
avalm=aval**3
avalm=(avalm/ftm)**0.3333
dxijkm=dxijk/ftm
if(dxijkm.gt.0.0)dxijkm=-9999.9
np0=np
call xmon(np0,clev,dxijk,xero,xread,yread)
if(ibm.eq.2)xero=xero/ftm
c write(6,88)acnty,rng(1),rng(2),irang,hbi,stval,amp,aval,
c 1 dxijk,jper(irank)
if(hbi.gt.0.0.and.ibm.eq.2)
1 write(6,88)acnty,rng(1),rng(2),irang,hbm,stvalm,ampm,avalm,
1 dxijkm,xero,jper(irank)
88 format(' ',10x,a3,1x,2a1,i3,F6.2,F6.2,f5.2,f7.4,4x,f7.1,
1 'm',f7.1,'m',2x,i3,' years')

```

```

mai6540
mai6550
mai6560
mai6570
mai6580
mai6590
mai6600
mai6610
mai6620
mai6630
mai6640
mai6650
mai6660
mai6670
mai6680
mai6690
mai6700
mai6710
mai6720
mai6730
mai6740
mai6750
mai6760
mai6770
mai6780
mai6790
mai6800
mai6810
mai6820
mai6830
mai6840
mai6850
mai6860
mai6870
mai6880
mai6890
mai6900
mai6910
mai6920
mai6930
mai6940
mai6950
mai6960
mai6970
mai6980
mai6990
mai7000
mai7010
mai7020
mai7030
mai7040
mai7050
mai7060
mai7070
mai7080
mai7090
mai7100
mai7110
mai7120
mai7130
mai7140
mai7150
mai7160
mai7170
mai7180
mai7190

```

```

      if(hbi.gt.0.0.and.ibm.eq.1)
1 write(6,89)acnty,rng(1),rng(2),irang,hbim,stvalm,ampm,avalm,
1 dxijkm,xero,jper(irank)
89 format(' ',10x,a3,1x,2a1,i3,F6.2,F6.2,f5.2,f7.4,4x,f7.1,
1 'ft',f7.1,'ft',2x,i3,' years')
      if(hbi.le.0.0)
1 write(6,828)acnty,rng(1),rng(2),irang,stvalm,ampm,avalm,
1 jper(irank)

828 format(' ',10x,a3,1x,2a1,i3,6x,F6.2,f5.2,f7.4,24x,
1 i3,' years')

c      if(iout.ne.1.and.mod(iout,44).eq.0)write(6,883)
883 format(' ',//10x,
1 'Note: A value of 0.0m indicates no erosion',
1 1x,'or erosion not computed.'//)
450 continue
c      write smoothed profile data: *****
      if(ibm.eq.1)write(9,455)acnty,(rng(irn),irn=1,6)
455 format(' ',a3,1x,6a1,
1 ' Smoothed Orig & 25,15,10,5 Yr Final Profs',
2 ' -(dist,elev) in ft:')
      if(ibm.eq.2)write(9,456)acnty,(rng(irn),irn=1,6)
456 format(' ',a3,1x,6a1,
1 ' Smoothed Orig & 25,15,10,5 Yr Final Profs, ',
2 ' -(dist,elev) in m:')
      nxx1=nsxx(1)
      nxx2=nsxx(2)
      nxx3=nsxx(3)
      nxx4=nsxx(4)
      nxx5=nsxx(5)
      nxxs=max(nxx1,nxx2,nxx3,nxx4,nxx5)
      ntxt=min(nxx1,nxx2,nxx3,nxx4,nxx5)
c      write(*,*)'max of nsxx=',nxx1,nxx2,nxx3,nxx4,nxx5,nxxs
      do 600 i=1,ntxt
        write(9,606)(sxx(ii,i),syy(ii,i),ii=1,5)
606 format(' ',
1 f7.1,f7.2,f7.1,f7.2,f7.1,f7.2,f7.1,f7.2,f7.1,f7.2)
c      if(i.le.nxx1)write(9,601)sxx(1,i),syy(1,i)
601 format(' ',f7.1,f7.2)
c      if(i.le.nxx2)write(9,602)sxx(2,i),syy(2,i)
602 format('+',14x,f7.1,f7.2)
c      if(i.le.nxx3)write(9,603)sxx(3,i),syy(3,i)
603 format('+',28x,f7.1,f7.2)
c      if(i.le.nxx4)write(9,604)sxx(4,i),syy(4,i)
604 format('+',42x,f7.1,f7.2)
c      if(i.le.nxx5)write(9,605)sxx(5,i),syy(5,i)
605 format('+',56x,f7.1,f7.2)
600 continue
616 nsxx(1)=0
      nsxx(2)=0
      nsxx(3)=0
      nsxx(4)=0
      nsxx(5)=0
      GO TO 1
c99999 if(mod(iout,44).ne.0)write(6,883)
99999 STOP
      end
      subroutine smooth(NBEG,NMAX,ICODE)
C
C      Smooth survey data:
C      if icode = 2, smooth rear of dune
C      if icode = 1, smooth front of dune
C

```

```

mai7200
mai7210
mai7220
mai7230
mai7240
mai7250
mai7260
mai7270
mai7280
mai7290
mai7300
mai7310
mai7320
mai7330
mai7340
mai7350
mai7360
mai7370
mai7380
mai7390
mai7400
mai7410
mai7420
mai7430
mai7440
mai7450
mai7460
mai7470
mai7480
mai7490
mai7500
mai7510
mai7520
mai7530
mai7540
mai7550
mai7560
mai7570
mai7580
mai7590
mai7600
mai7610
mai7620
mai7630
mai7640
mai7650
mai7660
mai7670
mai7680
mai7690
mai7700
mai7710
mai7720
mai7730
mai7740
mai7750
mai7760
mai7770
mai7780
mai7790
smo 10
smo 20
smo 30
smo 40
smo 50
smo 60

```

c	smoothing is done from highest elev	smo 70
C	to 1st or last survey point	smo 80
c	uses H,X to set HA, XA	smo 90
C		smo 100
	parameter (nx=250)	smo 110
		smo 120
	COMMON /A/ H1(nx),XA(nx),HA(nx),NELM,X1(nx),NP,NELM1	smo 130
	NSTART=NBEG	smo 140
	NFINI=NMAX-1	smo 150
	IF (ICODE .EQ. 2) NFINI = NMAX	smo 160
	IF (ICODE .EQ. 2) NSTART = NBEG - 1	smo 170
C	WRITE(6,10)NSTART,NFINI	smo 180
10	FORMAT(' NSTART,NFINI ',2I5)	smo 190
	INC = 1	smo 200
	IF (ICODE .EQ. 2) INC = -1	smo 210
	DO 800 I = NSTART,NFINI,INC	smo 220
	H11=H1(I)	smo 230
	H22=H1(I+1)	smo 240
	DH=ABS(H11-H22)	smo 250
	DX=X1(I+1)-X1(I)	smo 260
	HU=H11	smo 270
	HL=H22	smo 280
C	WRITE(6,20)H11,H22,DH,DX,HU,HL	smo 290
20	FORMAT(' H1,H2,DH,DX,HU,HL ',6F7.2)	smo 300
	IF (H22.LT.H11) GO TO 100	smo 310
	HU=H22	smo 320
	HL=H11	smo 330
100	NEL1 = NELM1	smo 340
	NEL2 = NELM	smo 350
	IF (ICODE .NE. 2) GO TO 200	smo 360
	NEL1 = NELM1	smo 370
	NEL2 = 1	smo 380
200	DO 300 NEL=NEL1,NEL2,INC	smo 390
	IF (HA(NEL).LT.HU) GO TO 400	smo 400
300	CONTINUE	smo 410
C400	WRITE(6,30)NEL,NEL1,NEL2,HA(NEL),HU	smo 420
30	FORMAT(' NEL,NEL1,NEL2,HA(NEL),HU',3I5,2F7.2)	smo 430
400	NB=NEL	smo 440
	NEL2 = NELM	smo 450
	IF (ICODE .EQ. 2) NEL2=1	smo 460
	DO 700 N = NB,NEL2,INC	smo 470
	IF (HA(N).GT.HL) GO TO 600	smo 480
	XA(N)=XA(N)+DX	smo 490
	GO TO 700	smo 500
600	DEL1 = (HU - HA(N))/DH	smo 510
	XA(N) = XA(N) + DX * DEL1	smo 520
C	WRITE(6,40)N,HA(N),HL,X(N),XA(N),DEL1	smo 530
40	FORMAT(' N,HA(N),HL,X(N),XA(N),DEL1 ',I5,5F7.2)	smo 540
700	CONTINUE	smo 550
800	CONTINUE	smo 560
	RETURN	smo 570
	end	smo 580
	subroutine calca(A,NL)	cal 10
	parameter (nx=250)	cal 20
		cal 30
	COMMON /A/ H1(nx),XA(nx),HA(nx),NELM,X1(nx),NP,NELM1	cal 40
	XM=2.0/3.0	cal 50
	SUM1=0.0	cal 60
	SUM2=0.0	cal 70
	N=0	cal 80
	XS=0.0	cal 90
	DO 10 I=1,NL	cal 100
	IF (HA(I).Lt.0.0) GO TO 10	cal 110
	IF (XS.NE.0.0) GO TO 5	cal 120
	IX=I	cal 130
	XS=XA(I)	cal 140

	GO TO 10	cal 150
5	if(xa(i).eq.xs)go to 10	cal 160
c	SUM1=SUM1 + ALOG(XA(I)-XS)	cal 170
	if((xa(i)-xs).gt.0.0)SUM1=SUM1 + ALOG(XA(I)-XS) ! mod 1/7/1994	cal 180
c	SUM2=SUM2 + ALOG(HA(I))	cal 190
	if(ha(i).gt.0.0)SUM2=SUM2 + ALOG(HA(I)) ! mod. 1/7/1994	cal 200
	N = N + 1	cal 210
10	CONTINUE	cal 220
	XN = FLOAT(N)	cal 230
	if(xn.gt.0.0)A = EXP((SUM2 - XM*SUM1)/XN)	cal 240
C	WRITE(6,12)	cal 250
12	FORMAT(' I XA(I) HA(I)',/)	cal 260
C	WRITE(6,13)(I,XA(I),HA(I),I=1,NL)	cal 270
13	FORMAT(I5,F10.2,F10.2)	cal 280
C	WRITE(6,15)SUM1,SUM2,XN,A,XS,IX	cal 290
15	FORMAT(' SUM1= ',F8.2,' SUM2= ',F8.2,' XN= ',F8.2,' A= ',F8.2,	cal 300
-	' XS= ',F8.2,' IX= ',I5,/)	cal 310
	RETURN	cal 320
	end	cal 330
	subroutine avala(xmd,xmoff,nl,iwl)	ava 10
	parameter (nx=250)	ava 20
	COMMON /A/ H1(nx),XA(nx),HA(nx),NELM,X1(nx),NP,NELM1	ava 30
c	real ha(300),xa(300),xasave(300)	ava 40
c	do 40 i=1,nl	ava 50
c40	xasave(i)=xa(i)	ava 60
	nlml=nl-1	ava 70
	do 60 ll=1,10	ava 80
	do 50 i=1,nlml	ava 90
	scur=xmd	ava 100
		ava 110
	if(i.gt.iwl)scur=xmoff	ava 120
	dx=xa(i+1)-xa(i)	ava 130
	if(dx.le.0.0)dx=0.001 ! ** added 12/7/1993	ava 140
	dh=ha(i+1)-ha(i)	ava 150
	sl=dh/dx	ava 160
	if(sl.le.scur.and.dx.gt.0.0)go to 50	ava 170
	dxa=dh/scur	ava 180
c	determine excess slope	ava 190
	dx1=1.1*(dxa-dx)	ava 200
	xa(i)=xa(i)-dx1/2.0	ava 210
	xa(i+1)=xa(i+1)+dx1/2.0	ava 220
50	continue	ava 230
60	continue	ava 240
	return	ava 250
	end	ava 260
	subroutine range(rng,irang)	ran 10
	integer x(3)	ran 20
c	integer value of the monument range ID is extracted, any letters	ran 30
c	associated with the range ID are ignored.	ran 40
	character*1 val,rng(8)	ran 50
	character*1 z(10)	ran 60
	data z/'1','2','3','4','5','6','7','8','9','0'/	ran 70
	x(1)=0	ran 80
	x(2)=0	ran 90
	x(3)=0	ran 100
	kk=0	ran 110
	do 30 k=3,5	ran 120
	kk=k-2	ran 130
	val=rng(k)	ran 140
	do 20 i=1,10	ran 150
	if(val.eq.z(i))go to 25	ran 160
20	continue	ran 170
	kk=kk-1	ran 180
	go to 35	ran 190
25	x(kk)=mod(i,10)	ran 200
c	write(*,*)' i=',i	ran 210

30	continue	ran 220
35	if(kk.eq.1)irang=x(1)	ran 230
	if(kk.eq.2)irang=x(1)*10+x(2)	ran 240
	if(kk.eq.3)irang=x(1)*100+x(2)*10+x(3)	ran 250
	return	ran 260
	end	ran 270
	subroutine hfe (acnty,r,a,amp,kounty,stides,hbs)	hfe 10
c	get high frequency erosion data from hfe.dat file	hfe 20
c		hfe 30
c	r= monument number; a=parameter "a"; amp=mean tidal	hfe 40
c	amplitude; stides=max storm tides; hbs=wave heights	hfe 50
c	acnty,cnty= 3 character county id; rid=2 character mon id;	hfe 60
c		hfe 70
	dimension ztide(10),zstide(4,10),zhbs(4,10)	hfe 80
	dimension stides(4),hbs(4)	hfe 90
c	1 ,trange(12)	hfe 100
	integer r,zr2(100)	hfe 110
	character*3 acnty,cnty,county(24),dum	hfe 120
	character*2 rid	hfe 130
	external yint	hfe 140
c		hfe 150
	data county	hfe 160
1	/'NAS','DUV','STJ','FLA','VOL','BRE','IND','STL','MAR','PAL',	hfe 170
2	'BRO','DAD','GUL','FRA','PIN','MAN','SAR','CHA','LEE','COL',	hfe 180
3	'ESC','OKA','WAL','BAY'/	hfe 190
c		hfe 200
c	data trange	hfe 210
c	1 /82.,80.,209.,100.,234.,219.,119.,115.,127.,227.,128.,113./	hfe 220
c		hfe 230
c	get index for county	hfe 240
	do 10 k=1,24	hfe 250
	if(acnty.eq.county(k))go to 15	hfe 260
10	continue	hfe 270
	write(*,*)' *** Error: county id did not match,acnty=',acnty	hfe 280
	stop	hfe 290
15	kounty=k	hfe 300
		hfe 310
	if(kounty.eq. 1)open(unit=3,file='enas0594.dat',status='old')	hfe 320
	if(kounty.eq. 2)open(unit=3,file='eduv0594.dat',status='old')	hfe 330
	if(kounty.eq. 3)open(unit=3,file='estj0594.dat',status='old')	hfe 340
	if(kounty.eq. 4)open(unit=3,file='efla0594.dat',status='old')	hfe 350
	if(kounty.eq. 5)open(unit=3,file='evol0594.dat',status='old')	hfe 360
	if(kounty.eq. 6)open(unit=3,file='ebre0594.dat',status='old')	hfe 370
	if(kounty.eq. 7)open(unit=3,file='eind0594.dat',status='old')	hfe 380
	if(kounty.eq. 8)open(unit=3,file='estl0594.dat',status='old')	hfe 390
	if(kounty.eq. 9)open(unit=3,file='emar0594.dat',status='old')	hfe 400
	if(kounty.eq.10)open(unit=3,file='epal0594.dat',status='old')	hfe 410
	if(kounty.eq.11)open(unit=3,file='ebro0594.dat',status='old')	hfe 420
	if(kounty.eq.12)open(unit=3,file='edad0594.dat',status='old')	hfe 430
	if(kounty.eq.13)open(unit=3,file='pgull1094.dat',status='old')	hfe 440
	if(kounty.eq.14)open(unit=3,file='pfra1094.dat',status='old')	hfe 450
	if(kounty.eq.15)open(unit=3,file='wpin1094.dat',status='old')	hfe 460
	if(kounty.eq.16)open(unit=3,file='wman1094.dat',status='old')	hfe 470
	if(kounty.eq.17)open(unit=3,file='wsar1094.dat',status='old')	hfe 480
	if(kounty.eq.18)open(unit=3,file='wcha1094.dat',status='old')	hfe 490
	if(kounty.eq.19)open(unit=3,file='wlee1094.dat',status='old')	hfe 500
	if(kounty.eq.20)open(unit=3,file='wcoll1094.dat',status='old')	hfe 510
	if(kounty.eq.21)open(unit=3,file='pesc1094.dat',status='old')	hfe 520
	if(kounty.eq.22)open(unit=3,file='poka1094.dat',status='old')	hfe 530
	if(kounty.eq.23)open(unit=3,file='pwall1094.dat',status='old')	hfe 540
	if(kounty.eq.24)open(unit=3,file='pbay1094.dat',status='old')	hfe 550
c		hfe 560
	read(3,111)dum	hfe 570
	read(3,111)dum	hfe 580
111	format(1x,a3)	hfe 590
		hfe 600

```

do 20 n=1,10
read(3,11,end=35) cnty, rid, zr2(n), (zstide(i,n), i=1,4), ztide(n),
1      (zhbs(j,n), j=1,4)
11      format(1x,a3,1x,a2,i3,1x,5f7.2/11x,4f7.2)
      if(cnty.ne.acnty) go to 25
c11      format(1x,a3,1x,a2,i3,f7.4) ... for para.dat
20      continue
      go to 35
25      write(*,22) acnty, cnty
22      format(' *** Error check:',a3,' .ne. ',a3)
      stop
35      close(unit=3)
      n=n-1
c      check if range id match, in which case get amp, stides, hbs direch
do 50 j=1,n
c      In some cases a monument ID is designated R-0 !
      if(r.eq.0) go to 51
      if(zr2(j).ne.r) go to 50
51      amp=ztide(j)
      do 40 i=1,4
      stides(i)=zstide(i,j)
      hbs(i)=zhbs(i,j)
40      continue
      go to 75
50      continue
      do 60 i=2,n
      if(r.gt.zr2(i-1).and.r.lt.zr2(i)) go to 65
60      continue
      write(*,*) ' ***Error: Monument Range # out of range=',r
      stop
65      izr2=zr2(i-1)
      x1=float(izr2)
      izr2=zr2(i)
      x2=float(izr2)
      x3=float(r)
c      write(*,*) ' hfe: x3,r=',x3,r
      y1=ztide(i-1)
      y2=ztide(i)
      amp=yint(x1,x2,x3,y1,y2)
      do 70 j=1,4
      y1=zstide(j,i-1)
      y2=zstide(j,i)
      stides(j)=yint(x1,x2,x3,y1,y2)
      y1=zhbs(j,i-1)
      y2=zhbs(j,i)
      hbs(j)=yint(x1,x2,x3,y1,y2)
70      continue
c      get parameter "A"
75      continue
      if(kounty.eq. 1) open(unit=3,file='anas0594.dat',status='old')
      if(kounty.eq. 2) open(unit=3,file='aduv0594.dat',status='old')
      if(kounty.eq. 3) open(unit=3,file='astj0594.dat',status='old')
      if(kounty.eq. 4) open(unit=3,file='afla0594.dat',status='old')
      if(kounty.eq. 5) open(unit=3,file='avol0594.dat',status='old')
      if(kounty.eq. 6) open(unit=3,file='abre0594.dat',status='old')
      if(kounty.eq. 7) open(unit=3,file='aind0594.dat',status='old')
      if(kounty.eq. 8) open(unit=3,file='astl0594.dat',status='old')
      if(kounty.eq. 9) open(unit=3,file='amar0594.dat',status='old')
      if(kounty.eq.10) open(unit=3,file='apal0594.dat',status='old')
      if(kounty.eq.11) open(unit=3,file='abro0594.dat',status='old')
      if(kounty.eq.12) open(unit=3,file='adad0594.dat',status='old')

```

```

if(kounty.eq.13) open(unit=3,file='agull1094.dat',status='old') hfe1270
if(kounty.eq.14) open(unit=3,file='afra1094.dat',status='old') hfe1280
if(kounty.eq.15) open(unit=3,file='apin1094.dat',status='old') hfe1290
if(kounty.eq.16) open(unit=3,file='aman1094.dat',status='old') hfe1300
if(kounty.eq.17) open(unit=3,file='asar1094.dat',status='old') hfe1310
if(kounty.eq.18) open(unit=3,file='acha1094.dat',status='old') hfe1320
if(kounty.eq.19) open(unit=3,file='alee1094.dat',status='old') hfe1330
if(kounty.eq.20) open(unit=3,file='acoll1094.dat',status='old') hfe1340
if(kounty.eq.21) open(unit=3,file='aesc1094.dat',status='old') hfe1350
if(kounty.eq.22) open(unit=3,file='aoka1094.dat',status='old') hfe1360
if(kounty.eq.23) open(unit=3,file='awal1094.dat',status='old') hfe1370
if(kounty.eq.24) open(unit=3,file='abay1094.dat',status='old') hfe1380
hfe1390
read(3,111)dum hfe1400
read(3,44,end=99) cnty,rid,nr21,za1 hfe1410
if(acnty.ne.cnty)write(*,*)' ***Error County ID do not match' hfe1420
if(acnty.ne.cnty)stop hfe1430
if(r.eq.nr21)a=za1 hfe1440
if(r.eq.nr21)return hfe1450
do 80 j=1,500 hfe1460
read(3,44,end=99) cnty,rid,nr22,za2 hfe1470
44 format(1x,a3,1x,a2,i3,f7.4) hfe1480
c write(*,*)' cnty,rid,nr22,za2=',cnty,rid,nr22,za2,r,x3 hfe1490
if(acnty.ne.cnty)write(*,*)' *Error County ID not matched',cnty hfe1500
if(acnty.ne.cnty)stop hfe1510
if(r.eq.nr22)a=za2 hfe1520
if(r.eq.nr22)return hfe1530
if(r.gt.nr21.and.r.lt.nr22)go to 85 hfe1540
nr21=nr22 hfe1550
za1=za2 hfe1560
80 continue hfe1570
99 write(*,*)' *** Error Range # out of bound',x3 hfe1580
stop hfe1590
85 a=yint(x1,x2,x3,za1,za2) hfe1600
close(unit=3) hfe1610
hfe1620
return hfe1630
end hfe1640
function yint(x1,x2,x3,y1,y2) yin 10
c interpolate between 2 points for y-value: yin 20
deno=x2-x1 yin 30
if(abs(deno).le.0.01)deno=0.01 yin 40
sl=(y2-y1)/deno yin 50
yint=y2-sl*(x2-x3) yin 60
return yin 70
end yin 80
subroutine xmon(np0,clev,dxijk,xero,xread,yread) xmo 10
real xread(250),yread(250) xmo 20
do 10 np1=1,np0 xmo 30
if(yread(np1).lt.0.0)go to 15 xmo 40
10 continue xmo 50
write(*,*)' *** Error: survey data needs to extend below msl' xmo 60
15 do 20 np2=np1,1,-1 xmo 70
if(yread(np2).gt.-clev.and.yread(np2+1).le.-clev)go to 25 xmo 80
20 continue xmo 90
write(*,*)' *** Error: dune elev. lower than clev' xmo 100
25 xdif=xread(np2+1)-xread(np2) xmo 110
xero=xread(np2) xmo 120
if(xdif.ne.0.0)slope=(yread(np2+1)-yread(np2))/xdif xmo 130
if(xdif.ne.0.0)xero=xread(np2+1)-(yread(np2+1)+clev)/slope xmo 140
xero=xero+dxijk xmo 150
return xmo 160
end xmo 170

```