

NUREG/CR-2424
PNL-5088-1
Vol. 2

Mathematical Simulation of Sediment and Radionuclide Transport in Coastal Waters

Vol. 2: User's Manual and Computer Program Listing for FETRA

Prepared by Y. Onishi, F. L. Thompson

Pacific Northwest Laboratory

Prepared for
U.S. Nuclear Regulatory
Commission

8407110172 840531
PDR NUREG
CF-2424 R PDR

NOTICE

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, or any of their employees, makes any warranty, expressed or implied, or assumes any legal liability of responsibility for any third party's use, or the results of such use, of any information, apparatus, product or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

NOTICE

Availability of Reference Materials Cited in NRC Publications

Most documents cited in NRC publications will be available from one of the following sources:

1. The NRC Public Document Room, 1717 H Street, N.W.
Washington, DC 20555
2. The NRC/GPO Sales Program, U.S. Nuclear Regulatory Commission,
Washington, DC 20555
3. The National Technical Information Service, Springfield, VA 22161

Although the listing that follows represents the majority of documents cited in NRC publications, it is not intended to be exhaustive.

Referenced documents available for inspection and copying for a fee from the NRC Public Document Room include NRC correspondence and internal NRC memoranda; NRC Office of Inspection and Enforcement bulletins, circulars, information notices, inspection and investigation notices; Licensee Event Reports; vendor reports and correspondence; Commission papers; and applicant and licensee documents and correspondence.

The following documents in the NUREG series are available for purchase from the NRC/GPO Sales Program: formal NRC staff and contractor reports, NRC-sponsored conference proceedings, and NRC booklets and brochures. Also available are Regulatory Guides, NRC regulations in the *Code of Federal Regulations*, and *Nuclear Regulatory Commission Issuances*.

Documents available from the National Technical Information Service include NUREG series reports and technical reports prepared by other federal agencies and reports prepared by the Atomic Energy Commission, forerunner agency to the Nuclear Regulatory Commission.

Documents available from public and special technical libraries include all open literature items, such as books, journal and periodical articles, and transactions. *Federal Register* notices, federal and state legislation, and congressional reports can usually be obtained from these libraries.

Documents such as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings are available for purchase from the organization sponsoring the publication cited.

Single copies of NRC draft reports are available free, to the extent of supply, upon written request to the Division of Technical Information and Document Control, U.S. Nuclear Regulatory Commission, Washington, DC 20555.

Copies of industry codes and standards used in a substantive manner in the NRC regulatory process are maintained at the NRC Library, 7920 Norfolk Avenue, Bethesda, Maryland, and are available there for reference use by the public. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from the American National Standards Institute, 1430 Broadway, New York, NY 10018.

USNRC
JO ANN RESNER
W 501

ORDER NUMB: N11389-00003
ORDER TYPE: 2

DOCUMENT NO.	QTY @	COST	TOTAL COST	COMMENTS
NUREGCR2424V02	1	.00		
TOTAL DOCUMENTS ORDERED :	1			
TOTAL DOCUMENTS ENCLOSED:	1			

NUREG/CR-2424
PNL-5088-1
Vol. 2
RE

Mathematical Simulation of Sediment and Radionuclide Transport in Coastal Waters

Vol. 2: User's Manual and Computer Program Listing for FETRA

Manuscript Completed: April 1984
Date Published: May 1984

Prepared by
Y. Onishi, F. L. Thompson

Pacific Northwest Laboratory
Richland, WA 99352

Prepared for
Division of Health, Siting and Waste Management
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555
NRC FIN B2271

CONTENTS

1.0 INTRODUCTION.....	1.1
2.0 FETRA MODEL USER' MANUAL.....	2.1
APPENDIX - FETRA COMPUTER PROGRAM LISTINGSW.....	A.1

1.0 INTRODUCTION

FETRA is a finite element model for simulating the sediment and contaminant transport to surface water (Onishi 1981). The model was applied to a test site in the Irish Sea and modified to account for wave mechanisms that effect sediment suspension (Onishi, Arnold and Mayer 1979). The model consists of three submodels that are coupled to simulate sediment/contaminant interactions. The submodels simulate 1) sediment transport, 2) dissolved contaminant transport, and 3) particulate contaminant (those contaminants absorbed by sediment) transport.

Volume 1 of this report discusses the application of FETRA to the Irish Sea study area to simulate the effect of wind-generated surface waves on the transport of sediment, dissolved ^{137}Cs , and particulate ^{137}Cs . Volume 2 presents a very brief users guide for FETRA in Chapter 2 and a computer program listing of FETRA in Appendix A.

2.0 FETRA MODEL USER'S MANUAL

Most of the data are read by subroutine SPECS. The exceptions include the wind data, read by subroutine RWIND, and time-varying hydrodynamic data, read by subroutine RVEL. FETRA includes an option that allows the use of the coefficient subroutines for the transport code only. This option is considered separately following the main input instructions.

DIMENSIONS

A) Variables used to determine the correct dimensions are as follows:

Bandwidth	= $2*NCOL-1$ = largest difference of node numbers within any element (See calculations after Card Group 8 in SPECS)
IDN	See Card Group 3
JMAX	See Card Group 20
NBED	See Card Group 18-19
NCNDS	No. of corner nodes - calculated internally when Card Group 9 is read in
NDS	See Card Group 4-5
NE	See Card Group 4-5

B) The dimensions should be assigned as follows:

```
COMMON)BLKI/CT(NDS), C(IDN+2,NDS)
      BLK2/P(NDS, Bandwidth), F(NDS)
      BK3/S(NDS, Bandwidth)
      BLK4/R(NDS), RPAST(NDS, IDN)
      BLK6/VX(NDS), VY(NDS), H(NDS), STRESS(3)
      BLK7/NOD(NE,6), X(NDS), Y(NDS)
      BLK8/MBC(IDN, JMAX), NBC(IDN, JMAX), DB(IDN, JMAX), BC(IDN, JMAX)
      KODE(10)
      BLK9/DX(NE), UY(NE), ALFA(NE), BETA(NE), HS(NE)
      BLK10/PEL(6,6), SEL(6,6), REL(6)
      BLK11/D50(3,NE), BD50(NE), SR(3,NE), SD(3,NE)
      BLK12/ACOF(6), U(6), V(6), D(2), AKJ(9,NE), AKP(3)
```

BLK13/WS(3,NE), CRSTRS(3,NE), CDSTRS(3,NL), ERODA(3,NE)
 BLK14/GBA(NE,NBED), GBB(NE,NBED), GBC(NE,NBD), GBD(NE,NBED)
 GBE(NE,NBED), GBF(NE,NBEU), GB3(NE,NBED), POR
 BLK15/ILAYR(NE,3), XYSO(NE), BDIV(NE), NBED(NE), BED(NE),
 RHOSD(3), XNT(NE,3), RSAVI(NE), RSAV2(NE), RSAV(NE)
 BLK17/QLATE(NDS), QSARA(IDN,NE), QPNT(IDN,NDS), CRATE(3,NE)
 ARAD(3), ALEFT(3), B2(3), B3(3), GB](3), CS(3), DD(NDS,3)
 COMMON/WAVE(NDS), NC(NCND), (NCND,10), K(NCND,10), W(NCND,10),
 HB(NCND), ALPHAC(NCND), WAVE,N,D,VIS,DPTH(NCND)
 COMMON/WIND/WVEL(), WANG() - dependent on number of wind entries
 to be used
 COMMON/ZTYPE/ZTYP(NCND)

Subroutine RVEL DIMENSION NOD(NCND)

Note: DEFINE FILE statements must be consistent with
 DEFINE FILE in CAFE-I

Subroutine RWAWE

Note: Binary files called internally

Subroutine SAND NCHECK(NDS)

QCHECK(NDS)

UNITS

Meter, Kilogram (Force), Day, Picocurie
 Sediment Concentration: kg/cubic meter of water
 Dissolved Chemical: kg or pCi/cubic meter of water
 Chemical Attached to Sediment:
 Input: kg or pCi/cubic meter of water
 Output: kg or pCi/kg of sediment

REQUIRED INPUT DATA

CARD GROUP 1-2: Title. Two cards (20A4)

CARD GROUP 3: Identification of Simulation Substances. One card (215)

ID Simulation substance
 = 1,2,3 sediment (sand, silt, clay)
 = 4 dissolved chemical
 = 5,6,7 chemical adsorbed by sediment
 = 8 total amount of sediment
 = 9 total amount of chemical
 (Only one substance is identified. The card groups are repeated
 for each substance to be simulated.)

IN Flag for coefficient subroutines for transport code
 = 0 Only transport subroutines are to be called
 (See separate input instructions in following
 section)
 / 0 Used to include sediment/chemical interactions.

CARD GROUP 4-5: Integration Parameters. Two cards. (I) Card I (5I5)

(1) Card 1 (5I5)

IDN Total number of simulation substances
 NE total number of triangular elements
 NDS Total number of nodes
 NTP Maximum number of time steps to be simulated (0 for
 steady-state solution)
 NPRNT Print frequency (hard copy output for every NPRNT time
 steps)

(2) Card 2 (2E10.3)

T Time increment (in days)
 STYP Numerical solution type
 = 0 implies explicit solution
 = 0.5 implies Crank-Nicholson solution (default)
 = 1 implies implicit solution

CARD GROUP 6-7: Source Information. Two card sets.

(1) Area Source

(a) Card 1 (15)

NUQSA Number of area sources

(b) Only if NUQSA \neq 0: Card 2 (215, E10.3)
I Constituent number
J Source location (element number)
QSARA (I,J) Source strength (pCi/cubic meter of discharge/day)

(2) Point Source

(a) Card 1 (15)

NUQSP Number of point sources

(b) Only if NUQSP \neq 0 : Card 2 (215, E10.3)

I Constituent number

J Source location (node number)

QPNT(I,J) Source strength (pCi/cubic meter of discharge/day)

CARD GROUP 8: Element-Node Connectivity. NE cards (715).

I Element number

NOD (I,K) Six node numbers corresponding to each element
(counterclockwise direction)

CARD GROUP 9: Node Coordinates. NCNDS cards (15, 2E10.2) where NCNDS is the number of corner nodes.

J Corner node number (FETRA interpolates to find the
the coordinates of midpoints)

X(J) X-coordinate of j-th node

Y(J) Y-coordinate of j-th node

(1 Blank Card must follow this card group.)

CARD GROUP 10: Specification of Options. Two cards.

By specifying any of the values given below (in any order on a given card), the associated options are activated. Separate the options with a comma. If no options are desired, input a blank card.

(1) Card 1 [9(A5,1X)]

NUCOF,	Nonuniform coefficients (for card group 15-16-17)
NUVEL,	For nonuniform velocity and/or depth
TVB-S,	For time varying boundary conditions
TVHYD,	For time varying hydraulics
TVINP,	For time varying input data
USSDI,	Uniform Suspended sediment diameter
NSSDI,	Nonuniform
UBSDI,	Uniform Bed sediment diameter
NBSDI,	Uniform
UBGEO,	Uniform Bed configuration
NBGEO,	Nonuniform
UBCON,	Uniform Bed concentration
NBCON,	Nonuniform
STORE,	Store computed results at the last time step for future restart
RSTRT,	Restart from the previously stored results

(2) Card 2 (4A5)

CLAY,	Required if INO (see Card 3) is greater than zero and clay is the limiting bed layer.
SAND,	Required if INO (see Card 3) is greater than zero and sand is the limiting bed layer.
WAVE,	Required if the contaminant and/or sediment transport at any of the nodes is influenced by a wave or surf environment. This will be specified only for sand computation input data stream.
WAVR,	Required if 'WAVE,' is specified and the WAVE and SURF characteristics are to be read from files created by other programs (e.g., L03D output files).
VELR,	Required if the velocity field is to be read from files created by other programs (e.g., CAFE-I output files).

CARD GROUP 11-12: Sediment Properties. Two card sets.

(1) Only for ID = 1,2,3: 1 Card or NE Cards (5E10.3)

D50(ID,M) Suspended sediment diameter size
WS(ID,M) Fall velocity of sediment element M
CRSTRS(ID,M) Critical shear stress for scouring
CDSTRS(ID,M) Critical shear stress for deposition
ERODA(ID,M) Erodeability coefficient ID = 1,2,3 at element M
 If 'USSDI,' M = 1 (1 card read)
 If 'NSSDI,' M = 1, NE (NE cards read)

(2) Only for ID = 1: 1 card or NE cards (E 10.3)

BD50(M) Bed sediment size
 If 'UBSDI,' M = 1 (1 card read)
 if 'NBSDI,' M = 1, NE (NE cards read)

CARD GROUP 13-14: Initial Conditions. One of the following card sets
(A or B):

[A] For uniform initial conditions

(1) Card 1 (A5,1X)

UNICS, Flag for uniform initial conditions

(2) Card 2 (E10.3)

C(ID,1) Initial concentration of substance ID

[B] For nonuniform initial conditions

(1) Card 1 (A5,1X)

NUICS,

(2) NDS Cards (6E10.3)

C(ID,J), J = 1, NDS

[Card Group 13-14 is repeated here for the other (IDN-1) substances].

CARD GROUP 15-16-17: System Properties. Four card sets.

(1) Card 1 (4E10.3)

ALMBDA Decay rate
AKP(I), I=1,3 Flag for adsorption/desorption
 = 1 if adsorption/desorption occurs
 = 0 if no adsorption/desorption

Note: AKP(I) values are no longer read by FETRA

(2) One of the following card sets (a or b):

(a) For nonuniform coefficients (NUCOF option selected): (6E10.3)

[(AKJ(I,J), Adsorption/desorption constants
J=1,NE), I = 1,2,3 Kd value for simulation substance ID = 1,2,3;
 (m³/kg)

I=1,9] I = 4,5,6 Adsorption/desorption rate for simulation
 substance ID = 1,2,3 in suspended sediment;
 (1/day)

I = 7,8,9 Adsorption/desorption rate for simulation
 substance ID = 1,2,3 in bed sediment; (1/day)

(b) For uniform coefficients (NUCOF not selected): 9 cards (E 10.3)

AKJ(I,I) for I = 1,9 (Defined above)

(3) One Card (7E10.3)

RHOSED(I), I = 1,3 Specific weight of I-th sediment
RHOWAT Specific weight of water [1000kg (force)/cubic
 meters]
POR Porosity
VIS Kinematic viscosity of water, square meters/day

(4) One Card or NE Cards (5E10.3)

DX(J) Dispersion coefficient - X component
DY(J) Dispersion coefficient - Y component
ALFA(J) Decay term
BETA(J) Source/sink term (used only for INO ≠ 0; see
 following section)
H5(J) Element thickness

For uniform coefficients (NUCOF option selected): J = 1, NE

For uniform coefficients: J = 1

CARD GROUP 18-19: Bed Layer Inputs. Two card sets.

(1) One Card or NE cards (15, 3E10.3)

NBED(M) Initial number of bed layers
BDIV(M) Thickness of bed layers (used for all layers except top layer).

XYSO(M) Thickness of top bed layer

BED(M) Initial total bed thickness

If 'UBGEO,' M = 1 (1 card read)

If 'NBGEO,' M = 1, NE (NE cards read)

(2) NBED(M) sets of One Card or NE cards (6E10.3)

For element M, bed layer J:

GBA(M,J) = Weight fraction of clay if clay is the limiting bed layer

= Weight fraction of sand if sand is the limiting bed layer

GBB(M,J) = Weight fraction of silt

GBC(M,J) = Weight fraction of sand if clay is the limiting bed layer

= Weight fraction of silt if sand is the limiting bed layer

GBD(M,J) = Chemical concentration of contaminant per unit weight of sediment (clay or sand depending on the sediment type that GBA(M,J) applies to).

GBE(M,J) = Chemical concentration of contaminant per unit weight of silt

GBF(M,J) = Chemical concentration of contaminant per unit weight of sediment (clay or silt depending on the sediment type that GBC(M,J) applies to).

Note: Bed layers are numbered from deepest to shallowest

Note: Appropriate units for GBD, GBE, and GBF are pCi/kg or kg/kg.

If 'UBCON,' M = 1 (1 card read for each bed layer)

If 'NBCON,' M = 1, NE (NE cards read for each bed layer)

CARD GROUP 20: Boundary Condition Specification. Two card sets.

(1) Card 1 (2I5)

LBC Number of specified boundary condition nodes
KBC Number of derivative boundary condition nodes

(2) JMAX Cards [2(I5, E10.2)], where JMAX = larger of LBC, KB,

NBC(ID,J) Boundary node number
B(ID,J) Boundary node value (specified)
MBC(ID,J) Boundary node number
DBC(ID,J) Boundary node value (derivative)

Note: nodes must be input in a numerically increasing sequence.

CARD GROUP 21: Flow Field Information. One of the following card sets
(A, B, or C):

[A] For 'NUVEL,' (nonuniform velocity and/or depth) and 'VELR,' (Velocity field read from files created by other programs)

(1) Card 1 (E10.3)

VFREQ Frequency of data entries for velocity and/or depth (day)

(2) Card 2 (F10.0) - read by subroutine RVEL

AMAX Time in days for which velocity and depth files will be repeated

(3) NCNDS Cards (16I5), where NCNDS = no. of corner codes. (Read by Subroutine RVEL)

NOD(I), FETRA node numbers corresponding to CAFE node numbers
I=I,NCNDS (In ascending CAFE node number order)

[B] For 'NUVEL,' where 'VELR,' is not also specified

(1) NDS cards (3E10.3)

H(J) Flow depth of j-th node
VX(J) X-component of velocity at j-th node
VY(J) Y-component of velocity at j-th node

[C] For a uniform velocity field ('NUVEL,' is not specified)

(1) 1 Card (3E10.3)

H(1)

VX(1) As defined above

VY(1)

CARD'GROUP 22: Wave Sediment Transport Input Data. One of the following card sets.

[A] For 'WAVE,' and 'WAVR,'- subroutine RWAVE reads wave characteristics from files created by another program (e.g., L03D)

(1) Card 1 (I5, F10.0) - Read by subroutine RWIND

LW Number of wind data points to be input. The first wind data point will be used at time = 0. days

WFREQ Time interval in days between each wind data point

(2) (LW/8.) Cards (8E10.2)

WVEL(I) Wind velocity, m/day

WANG(I) Direction from which wind is blowing, degrees from true north measured CW

(I=1,LW)

[B] For 'WAVE,' where 'WAVR,' is not also specified - Wave characteristics for each node may be read in or calculated in subroutine WAVSIM

(1) For temporally constant wind: Two cards.

(a) Card 1 (A5,IX)

CWIND Flag for constant wind

(b) Card 2 (F10.0)

WNDVEL Wind velocity (m/day)

(2) For temporally variable wind: Card 1 (A5,IX)

VWIND,

(3) NCNDS card sets, where NCNDS = number of corner nodes

(a) Card 1 (I5,A5,I5)

NODNO Node number

ZTYP (NP) Wave zone type

= WAVE, if node is beyond surf zone and wave

characteristics for the node are to be read in

= SURF, if node is in surf zone and surf characteristics
for the node are to be read in

= WAVC, if wave characteristics are to be calculated (in
subroutine WAVSIM)

NC(NP) Number of wave characteristics to be read for this node
(default = 1)

(b) If ZTYP = 'WAVE,': NO cards (3E10.4), where NO = NC(NP)

A(NP,I) Wave amplitude, m

K(NP,I) Wave number = $2\pi/\text{wave length}$, 1./m

W(NP,I) Wave frequency = $2\pi/\text{wave period}$, 1./sec

(I=I,NO)

(c) If ZTYP = 'SURF, , (2*NO) cards [2F10.0/(5F10.0)], where NO = NC(NP)

HB(NP) Wave height at breaking, m

ALPHAC(NP) Angle between wave ray and the gradient of the bottom
bathymetry, degrees

K(NP,I) Wave number = $2\pi/\text{wave length}$, 1./m

(I=I, NO)

(d) If ZTYP = 'WAVC,': One card (3E10.4)

(i) For temporally constant wind ('CWIND,'):

DM Mean fetch depth, m

F Effective fetch length, m

(ii) For temporally variable wind ('VWIND,'):

WV Wind velocity, m/day

DM Mean fetch depth, m

F Effective fetch length, m

(4) Blank card

Repeat Card Groups for (ID = 2, IDN):

For (ID = 2,3), the following card groups are required:

- 1-2. Title (The substance being simulated is usually identified on line 2). Two cards (20A4)
- 3. Identification of Simulation Substances. One card (2I5)
- 10. Specification of Options. Two cards
- 11-12 Sediment Properties. First card set only.
- 20. Boundary Condition Specification. Two card sets

For (ID = 4,7), the card groups listed above are required with the exception of card group 11-12.

VFREQ (card group 2I) must be repeated at the end of the input data if AMAX (Card Group 2I) is less than the amount of time to be simulated. The number of cards (EIO.3) to be inserted is equal to $(NTP * T / AMAX)$ (see card group 4-5).

TRANSPORT CODE ONLY

FETRA has an option to run the coefficient subroutines for the transport code only. Sediment/chemical interactions are not accounted for when this option is selected. Card groups required as input for this option are listed below.

Required Input Data

See main input instructions for definitions of parameters. Parameter values specific to the transport code option are noted below.

CARD GROUP 1-2: Title. Two cards (20A4)

CARD GROUP 3: Identification of Simulation Substances. One card (2I5)

ID = i

INO = 0

CARD GROUP 4-5: Integration Parameters. Two cards (5I5/2E10.3)

IDN = 1

NTP = 0 For steady-state solution only

CARD GROUP 6-7: Source Information. Two blank cards. (Source/sink terms are handled through card group 15-16-17)

CARD GROUP 8: Element-Node Connectivity. NE cards (7I5)

CARD GROUP 9: Node Coordinates. NCNDS cards (I5,2E10.2), where NCNDS is the number of corner nodes.

(1 Blank Card = flag for end of node coordinate data.)

CARD GROUP 10: Specification of Options. Two blank cards

CARD GROUP 13-14: Initial Conditions. (See main input instructions.)

CARD GROUP 15-16-17: System Properties. One card (5E10.3)

DX(I) Dispersion coefficient - X component

DY(I) Dispersion coefficient - Y component

ALFA(I) Decay term

BETA(I) Source/sink specification used for transport code option

HS(I) Element thickness

CARD GROUP 20: Boundary Condition Specification. (See main input instructions.)

CARD GROUP 21: Flow Field Information. One card (3E10.3)

H(1) Flow depth

VX(1) X-component of velocity

VY(1) Y-component of velocity

REFERENCES

- Onishi Y. 1981. "Sediment-Contaminant Transport Model," Journal of the Hydraulic Division 107(HY9):1089-1107.
- Onishi, Y., E. M. Arnold and D. W. Mayer. 1979. Modified Finite Element Transport Model, FETRA, for Sediment and Radionuclides Migration in Open Coastal Waters. NUREG/CR-1026 (PNL-3114). Nuclear Regulatory Commission, Washington, D.C.

APPENDIX

FETRA COMPUTER PROGRAM LISTINGG

```

PROGRAM FETRA(INPUT,OUTPUT,LUQ,LUH,WAVZON,SURF,TAPES=INPUT,      FETRA      2
1 TAPE6=OUTPUT,TAPE7=LUQ,TAPE8=LUH,TAPE3=WAVZON,TAPE4=SURF,    FETFIX1    1
2 INTAPE,OUTAPE,REPOST,TAPE1=INTAPE,TAPE2=OUTAPE,TAPE9=REPOST) FETFIX1    2
C PRUGRAM FETRA (INPUT,OUTPUT,TAPES=INPUT,TAPE6=OUTPUT,TAPE1,TAPE2) FETRA      4
C FINITE ELEMENT TRANSPORT MODEL                                FETRA      5
C THIS PROGRAM SOLVES THE "TIME DEPENDENT" CONVECTION-DIFFUSION FETRA      6
C EQUATION IN TWO DIMENSIONS BY THE FINITE ELEMENT METHOD. THE FETRA      7
C GALERKIN PRINCIPLE WAS USED TO FORMULATE THE SYSTEM OF EQUATIONS FETRA      8
C FOR A TRIANGULAR ELEMENT. A QUADRATIC SHAPE FUNCTION BASED ON FETRA      9
C A LAGRANGIAN INTERPOLATING POLYNOMIAL IS USED. DOMAINS OF COMPLEX FETRA     10
C GEOMETRY ARE REPRESENTED BY A MESH OF 6-NODE TRIANGULAR ELEMENTS. FETRA     11
C TIME DEPENDENT MASS OR HEAT TRANSPORT MAY SIMULATED BY DEFINITION FETRA     12
C OF THE APPROPRIATE SINK AND/OR SOURCE TERMS.                FETRA     13
C                                                                FETRA     14
C                                                                FETRA     15
C 1. ONISHI,Y.,JOHANSON,P.A.,BACA,R.G.,AND HILTY,E.L.,*STUDIES OF FETRA     16
C COLUMBIA RIVER WATER QUALITY---DEVELOPMENT OF MATHEMATICAL FETRA     17
C MODELS FOR SEDIMENT AND RADIONUCLIDE TRANSPORT ANALYSIS*, FETRA     18
C BNWL-B-452, BATTELLE-PACIFIC NORTHWEST LABORATORIES TO THE FETRA     19
C U.S.ENERGY RESEARCH AND DEVELOPMENT ADMINISTRATION, JAN. 1976 FETRA     20
C 2. ONISHI,Y.,*FINITE ELEMENT MODELS FOR SEDIMENT AND CONTAMINANT FETRA     21
C TRANSPORT IN SURFACE WATERS TRANSPORT OF SEDIMENT AND FETRA     22
C RADIONUCLIDES IN THE CLINCH RIVER,* BNWL-2227, BATTELLE- FETRA     23
C NORTHWEST LABORATORIES, RICHLAND, WASHINGTON,JULY 1977 FETRA     24
C 3. ONISHI,Y.,*MATHEMATICAL SIMULATION OF SEDIMENT AND FETRA     25
C RADIONUCLIDE TRANSPORT IN THE COLUMBIA RIVER,* BNWL-2228, FETRA     26
C BATTELLE-PACIFIC NORTHWEST LABORATORIES, RICHLAND, WASHINGTON FETRA     27
C AUGUST 1977 FETRA     28
C 4. SCHECTER,R.S.,*VARIATIONAL METHODS IN ENGINEERING*, MCGRAW- FETRA     29
C HILL BOOK COMPANY, NEW YORK, 1967. FETRA     30
C 5. DESAI,C.S. AND J.F. ABEL,*INTRODUCTION TO THE FINITE ELEMENT FETRA     31
C METHOD*, VAN NOSTRAND REINHOLD COMPANY, 1972 FETRA     32
C 6. GUYMON,G.L., *APPLICATION OF THE FINITE-ELEMENT METHOD FOR FETRA     33
C SIMULATION OF SURFACE WATER TRANSPORT PROBLEMS,* INSTITUTE OF FETRA     34
C WATER RESOURCES, UNIVERSITY OF ALASKA, REPORT NO. IWR-21, 1972 FETRA     35
C 7. LAPIDUS, L. AND J. SEINFELD, *NUMERICAL SOLUTION OF ORDINARY FETRA     36
C DIFFERENTIAL EQUATIONS*, ACADEMIC PRESS, INC. 1971. FETRA     37
C                                                                FETRA     38
C**** RHOSED OR RHOWAT ARE ACTUALLY SPECIFIC WEIGHTS IN KG(FORCE)/M**3 FETRA     39
C**** RATHER THAN MASS DENSITY. THESE SHOULD BE CHANGED TO SPWSED AND FETRA     40
C**** SPW*AT. FETRA     41
C                                                                FETRA     42
C                                                                FETRA     43
COMMON NDS FETRA     44
COMMON /BLK1/CT(240),C(9,240) FETRA     45
COMMON /BLK2/P(240,86),F(240) FETRA     46
LEVEL 2,P,F FETFIX4    1
COMMON /BLK3/S(240,86) FETRA     47
LEVEL 2,S FETFIX4    2
COMMON /BLK4/ R(240), RPAST(240,7), NODBET, BETA1, AREA1 FETRA     48
LEVEL 2,R,RPAST,NODBET,BETA1,AREA1 FETFIX4    3
COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3) FETRA     49
COMMON /BLK7/NOD(240,6),X(240),Y(240) FETRA     50
COMMON /BLK8/K8C,L8C,M8C(7,120),N8C(7,120),DBC(7,120),BC(7,120), FETRA     51
1 KODE(10) FETRA     52
COMMON /BLK11/D50(3,100),BD50(100),SR(3,100),SD(3,100) FETRA     53
COMMON /BLK12/ACOF(b),U(b),V(b),D(2),AKJ(9,100),ALMBDA,RHOWAT, FETRA     54
1 AKP(3) FETRA     55
COMMON /BLK13/WS(3,100),CRSTRS(3,100),CDSTRS(3,100),ERODA(3,100) FETRA     56
COMMON /BLK14/G8A(100,10),G8B(100,10),G8C(100,10),G8D(100,10), FETRA     57
1 G8E(100,10),G8F(100,10),G8G(100,10),POR FETRA     58
COMMON /BLK15/ILAYR(100,3),XYSO(100),BDIV(100),NBED(100),BEU(100), FETRA     59
1 RHOSED(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100) FETRA     59

```

	COMMON /BLK17/QLATE(240),QSARA(7,100),QPNT(7,240),CRATE(3,100),	FETRA	60
	1 CD(4,240)	FETRA	61
	LEVEL 2,QLATE,QSARA,QPNT,CRATE,CD	FETFIX4	4
	LOGICAL RSTRT,STORE	FETRA	62
	DIMENSION ARAD(3),ALEFT(3),B2(3),B3(3),GBJ(3),CS(3)	FETRA	63
C		FETRA	64
C		FETRA	65
	DIMENSION DD(240,3)	FETRA	66
	DATA EPSI/1.0E-12/	FETRA	67
C		FETRA	68
C		FETRA	69
C		FETRA	70
	K=1	FETRA	71
	KKK=1	FETRA	72
	TSPEC = 0.0	FETRA	73
	TTRANS = 0.0	FETRA	74
	TBNDRY = 0.0	FETRA	75
	TAMSOL = 0.0	FETRA	76
	AT=0.0	FETRA	77
	RAT=0.0	FETRA	78
C		FETRA	79
C	INPUT THE PROBLEM SPECIFICATIONS	FETRA	80
C		FETRA	81
C	COMPUTATIONAL LOOP	FETRA	82
C		FETRA	83
C		FETRA	84
100	CONTINUE	FETFIX1	3
	T1 = SECOND(0.0)	FETRA	86
	CALL SPECS(NE,NDS,NCOL,NTP,T, ID, IDN, INO,K,NPRNT,RSTRT,STORE, KKK,	FETRA	87
	1 RAT)	FETFIX1	4
	TT1 = SECOND(T1)	FETRA	89
	TSPEC = TSPEC + TT1	FETRA	90
	IF(K.NE.1) GO TO 120	FETRA	91
	DO 115 I=1,NDS	FETRA	91
115	RPAST(I, ID) = 0.0	FETRA	92
120	NROW=NDS	FETRA	93
	IF(ID.GT.4) GO TO 126	FETRA	94
	DO 125 I=1,NDS	FETRA	95
	CD(ID, I) = C(ID, I)	FETRA	96
	IF(ID.LE.3.AND.CD(ID, I).LT.0.0) CD(ID, I) = 0.0	FETHA	97
125	CONTINUE	FETRA	98
126	CONTINUE	FETRA	99
C		FETRA	100
C	FORM THE COEFFICIENT MATRICES +	FETRA	101
C	(P), (S), (R)	FETRA	102
C		FETRA	103
	T2 = SECOND(0.0)	FETFIX1	5
C	H3 SPECIAL	FETRA	105
C	IF(ID.NE.4) GO TO 997	FETRA	106
	CALL TRANSP(NE,NROW,NCOL,T, ID, IDN, INO,K,NPRNT,NTP)	FETRA	107
	TT2 = SECOND(T2)	FETFIX1	6
	TTRANS = TTRANS + TT2	FETRA	109
C		FETRA	110
C	SET THE SPECIFIED BOUNDARY CONDITIONS IN THE LOAD VECTOR (R)	FETRA	111
C		FETRA	112
	T3 = SECOND(0.0)	FETFIX1	7
	CALL BNDRYS (NROW,NCOL, ID)	FETRA	114
	TT3 = SECOND(T3)	FETFIX1	8
	TBNDRY = TBNDRY + TT3	FETRA	116
C		FETRA	117
	IF (NTP.GT.0) GO TO 130	FETRA	118
C	COMPUTE STEADY STATE SOLUTION FOR TWO-DIMENSIONAL CONVECTION-	FETHA	119
C	DIFFUSION EQUATION	FETRA	120

WRITE (6,270)	FETRA	121
CALL BANSOL (NROW,NCOL,S,R)	FETRA	122
DO 140 I=1,NROW	FETRA	123
CT(I)=R(I)	FETRA	124
140 CONTINUE	FETRA	125
GO TO 160	FETRA	126
130 CONTINUE	FETRA	127
C	FETRA	128
C	FETRA	129
C	FETRA	130
C	FETRA	131
H3 SPECIAL CHANGE 1 TO 4	FETRA	132
IF(KKK.EQ.1.AND.ID.EQ.1) AT=AT+RAT	FETRA	133
IF (ID.EQ.1) AT=AT+T	FETRA	134
IF(MOD(K,NPRT) .NE. 0 .AND. K .NE. NTP) GO TO 150	FETRA	135
WRITE (6,280)	FETRA	136
*WRITE (6,225) K,T,AT,ID	FETRA	137
150 CONTINUE	FETFIX1	9
T4 = SECOND(0,0)	FETRA	139
CALL AMSOL(NROW,NCOL,ID,T)	FETFIX1	10
TT4 = SECOND(T4)	FETRA	141
TAMSOL = TAMSOL + TT4	FETRA	142
DO 155 I=1,NROW	FETRA	143
155 RPAST(I,ID) = R(I)	FETRA	144
C	FETRA	145
C	FETRA	146
C	FETRA	147
160 CONTINUE	FETRA	148
N=0	FETRA	149
J=0	FETRA	150
DO 170 I=1,LBC	FETRA	151
II=NBC(ID,I)	FETRA	152
170 C(ID,II)=BC(ID,I)	FETRA	153
180 N=N+1	FETRA	154
J=J+1	FETRA	155
DO 190 I=1,LBC	FETRA	156
IF (NBC(ID,I).EQ.J) J=J+1	FETRA	157
190 CONTINUE	FETRA	158
IF (J.GT.NDS) GO TO 200	FETRA	159
C(ID,J)=CT(N)	FETRA	160
IF (N.LT.NROW) GO TO 180	FETRA	161
200 CONTINUE	FETRA	162
IF(MOD(K,NPRT) .NE. 0 .AND. K .NE. NTP) GO TO 212	FETRA	163
C	FETRA	164
IF (ID.LT.5) GO TO 202	FETRA	165
DO 201 J=1,NDS	FETRA	166
IF (C(ID=4,J).GT.1.E-6)	FETRA	167
* C(ID,J) = C(ID,J)/C(ID=4,J)	FETRA	168
IF (C(ID=4,J).LE.1.E-6)	FETRA	169
* DO(J,ID=4)=C(ID,J)	FETRA	170
IF (C(ID=4,J).LE.1.E-6)	FETRA	171
* C(ID,J) = 0.0	FETRA	172
201 CONTINUE	FETRA	173
202 CONTINUE	FETRA	174
WRITE (6,230)	FETRA	175
WRITE(6,240) ID	FETRA	176
NPRT=NDS/6+1	FETRA	177
DO 210 I=1,NPRT	FETRA	178
NST=(I-1)*6+1	FETRA	179
IF (NST.GT.NDS) GO TO 210	FETRA	180
II=NST+5	FETRA	181
IF (II.GT.NDS) II=NDS	FETRA	182
WRITE (6,310) (J,C(ID,J),J=NST,II)		

210	CONTINUE	FETRA	183
	WRITE (9) (J,C(ID,J),J=1,NDS)	FETRA	184
C		FETRA	185
	IF (ID.LT.5) GO TO 212	FETRA	186
	DO 203 J = 1,NDS	FETRA	187
	IF (C(ID=4,J).GT.1.E=6)	FETRA	188
	* C(ID,J) = C(ID,J)*C(ID=4,J)	FETRA	189
	IF (C(ID=4,J).LE.1.E=6)	FETRA	190
	* C(ID,J) = DD(J,ID=4)	FETRA	191
203	CONTINUE	FETRA	192
212	CONTINUE	FETRA	193
	IF (INO.LE.0) GO TO 380	FETRA	194
C	IF (ID.EQ.IDN) CALL BEDHIS (NE,T)	FETRA	195
C	H3 SPECIAL	FETRA	196
997	CONTINUE	FETRA	197
	ID=ID+1	FETRA	198
	IF ((ID-IDN).LE.0) GO TO 100	FETRA	199
	ID=1	FETRA	200
C	H3 SPECIAL	FETRA	201
C	IF (ID.GT.0) GO TO 998	FETRA	202
	IF (MOD(K,NPRNT) .NE. 0 .AND. K .NE. NTP) GO TO 380	FETRA	203
C		FETRA	204
C**	CALCULATE AND OUTPUT TOTAL SUSPENDED SEDIMENT	FETRA	205
C**	CONCENTRATION, KG/M**3 OF WATER.	FETRA	206
C		FETRA	207
	DO 104 J=1,NDS	FETRA	208
	C(8,J)=0.	FETRA	209
	DO 104 I=1,3	FETRA	210
104	C(8,J)=C(8,J)+C(I,J)	FETRA	211
	WRITE(6,290)	FETRA	212
	WRITE(6,320)	FETRA	213
	WRITE(6,310) (J,C(8,J),J=1,NDS)	FETRA	214
	WRITE (9) (J,C(8,J),J=1,NDS)	FETRA	215
C		FETRA	216
C**	CALCULATE AND OUTPUT THE TOTAL CONCENTRATION OF CONTAMINANT	FETRA	217
C**	ATTACHED TO SUSPENDED SEDIMENT, PCI/M**3 OF WATER.	FETRA	218
C		FETRA	219
	DO 105 J=1,NDS	FETRA	220
	C(9,J)=0.	FETRA	221
	DO 105 I=1,3	FETRA	222
105	C(9,J)=C(9,J)+C(I+4,J)	FETRA	223
	WRITE(6,235)	FETRA	224
	WRITE(6,325)	FETRA	225
	WRITE(6,310) (J,C(9,J),J=1,NDS)	FETRA	226
	WRITE (9) (J,C(9,J),J=1,NDS)	FETRA	227
C		FETRA	228
C**	CALCULATE AND OUTPUT THE TOTAL SUSPENDED AND	FETRA	229
C**	DISSOLVED CONTAMINANT CONCENTRATION, PCI/M**3 OF WATER.	FETRA	230
C		FETRA	231
	DO 106 J=1,NDS	FETRA	232
106	C(9,J)=C(9,J)+C(4,J)	FETRA	233
	WRITE(6,235)	FETRA	234
	WRITE(6,326)	FETRA	235
	WRITE(6,310) (J,C(9,J),J=1,NDS)	FETRA	236
	WRITE (9) (J,C(9,J),J=1,NDS)	FETRA	237
C	H3 SPECIAL	FETRA	238
C 998	CONTINUE	FETRA	239
C	THAS=(0.25*C(4,160)+0.25*C(4,186)+0.5*C(4,188)+	FETRA	240
C	1 2.0*C(4,173)+	FETRA	241
C	2 3.0*C(4,187)+3.0*C(4,174)+0.5*C(4,160)+0.26*C(4,188)+	FETRA	242
C	3 0.24*C(4,162)+4.0*C(4,174)+5.0*C(4,175)+3.96*C(4,161)+	FETRA	243
C	4 0.186*C(4,162)+0.163*C(4,188)+0.348*C(4,164)+6.66*C(4,175)+	FETRA	244

C	5	7.37*C(4,176)+7.34*C(4,163)+0.209*C(4,164)-0.302*C(4,188)+	FETRA	245
C	6	0.093*C(4,190)+7.93*C(4,176)+7.77*C(4,186)+8.46*C(4,177))	FETRA	246
C		TMASS=TMASS+(-0.25*C(4,186)-0.252*C(4,212)+	FETRA	247
C	1	0.504*C(4,188)+2.01*C(4,199)+	FETRA	248
C	2	3.02*C(4,200)+3.02*C(4,167)+0.240*C(4,188)-0.516*C(4,212)+	FETRA	249
C	3	0.259*C(4,214)+4.07*C(4,200)+4.12*C(4,213)+5.13*C(4,201)-	FETRA	250
C	4	0.142*C(4,188)-0.106*C(4,214)+0.248*C(4,216)+6.65*C(4,201)+	FETRA	251
C	5	7.17*C(4,215)+7.13*C(4,202)-0.261*C(4,188)+0.128*C(4,216)+	FETRA	252
C	6	0.134*C(4,190)+7.60*C(4,202)+8.13*C(4,203)+7.61*C(4,189))	FETRA	253
C		TMASS = 2.0E6 * TMASS	FETRA	254
C		WRITE (6,780) K, TMASS	FETRA	255
C780		FORMAT (" K, TMASS = ", I5, 1PE15.4)	FETRA	256
C			FETRA	257
		TMASS = 0.0	FETRA	258
		PAMASS = 0.0	FETFIX6	1
		TOTBED = 0.0	FETFIX6	2
		DO 800 M=1, NE	FETRA	259
		N1 = NOD(M,1)	FETRA	260
		N2 = NOD(M,2)	FETRA	261
		N3 = NOD(M,3)	FETRA	262
		N4 = NOD(M,4)	FETRA	263
		N5 = NOD(M,5)	FETRA	264
		N6 = NOD(M,6)	FETRA	265
		A2 = X(N1)-X(N3)	FETRA	266
		A3 = X(N2)-X(N1)	FETRA	267
		A5 = Y(N3)-Y(N1)	FETRA	268
		A6 = Y(N1)-Y(N2)	FETRA	269
		ARE = (A3*A5-A2*A6)/2.0	FETRA	270
		TMASS = TMASS + ((H(N1)/30.-H(N2)/60.-H(N3)/60.)*C(4,N1) +	FETRA	271
	1	(-H(N1)/60.+H(N2)/30.-H(N3)/60.)*C(4,N2) + (-H(N1)/60.-	FETRA	272
	2	H(N2)/60.+H(N3)/30.)*C(4,N3) + (H(N1)/7.5+H(N2)/7.5+	FETRA	273
	3	H(N3)/15.)*C(4,N4) + (H(N1)/15.+H(N2)/7.5+H(N3)/7.5)*	FETRA	274
	4	C(4,N5) + (H(N1)/7.5+H(N2)/15.+H(N3)/7.5)*C(4,N6))*ARE	FETRA	275
C			FETFIX6	3
		PAMASS = PAMASS + ((H(N1)/30.-H(N2)/60.-H(N3)/60.)*C(9,N1) +	FETFIX6	4
	1	(-H(N1)/60.+H(N2)/30.-H(N3)/60.)*C(9,N2) + (-H(N1)/60.-	FETFIX6	5
	2	H(N2)/60.+H(N3)/30.)*C(9,N3) + (H(N1)/7.5+H(N2)/7.5+	FETFIX6	6
	3	H(N3)/15.)*C(9,N4) + (H(N1)/15.+H(N2)/7.5+H(N3)/7.5)*	FETFIX6	7
	4	C(9,N5) + (H(N1)/7.5+H(N2)/15.+H(N3)/7.5)*C(9,N6))*ARE	FETFIX6	8
C			FETFIX6	9
		NBE1 = NBED(M)	FETFIX6	10
		TOTAL = GBA(M,NBE1)/RHOSED(1)+GBB(M,NBE1)/RHOSED(2)	FETFIX6	11
		1 +GBC(M,NBE1)/RHOSED(3)	FETFIX6	12
		XND = (1.0-POR)/TOTAL	FETFIX6	13
		TOTBED = TOTBED+GBG(M,NBE1)*XND*XYSO(M)*ARE	FETFIX6	14
		NBED1 = NBED(M)-1	FETFIX6	15
		DO 800 J=1,NBED1	FETFIX6	16
		TOTAL = GBA(M,J)/RHOSED(1)+GBB(M,J)/RHOSED(2)	FETFIX6	17
		1 +GBC(M,J)/RHOSED(3)	FETFIX6	18
		XND = (1.0-POR)/TOTAL	FETFIX6	19
		TOTBED = TOTBED + GBG(M,J)*XND*BDIV(M)*ARE	FETFIX6	20
800		CONTINUE	FETRA	276
		PAMASS = PAMASS+TMASS	FETFIX6	21
		TOTMAS = TMASS+PAMASS+TOTBED	FETFIX6	22
		WRITE(6,702) TOTMAS, TMASS, PAMASS, TOTBED	FETFIX6	23
702		FORMAT (/ "TOTAL MASS = ", 1PE15.4, 2X, "DISSOLVED MASS = ", 1PE15.4,	FETFIX6	24
	1	2X, "PARTICULATE MASS = ", 1PE15.4, 2X, "BED MASS = ", 1PE15.4/)	FETFIX6	25
C			FETRA	279
C		H3 SPECIAL	FETRA	280
C		IF(ID.GT.0) GO TO 999	FETRA	281
C**		CALCULATE AND OUTPUT THE WEIGHTED AVERAGE OF	FETRA	282
C**		CONTAMINANT ATTACHED TO SUSPENDED SEDIMENT,	FETRA	283

C**	PCI/KG OF SEDIMENT.	FETRA	284
C		FETRA	285
	DO 107 J=1,NDS	FETRA	286
	IF(C(8,J) .EQ. 0.) GO TO 107	FETRA	287
	C(9,J)=(C(9,J)-C(4,J))/C(8,J)	FETRA	288
107	CONTINUE	FETRA	289
	WRITE(6,235)	FETRA	290
	WRITE(6,330)	FETRA	291
	WRITE(6,310) (J,C(9,J),J=1,NDS)	FETRA	292
	WRITE(9) (J,C(9,J),J=1,NDS)	FETRA	293
	WRITE(6,290)	FETRA	294
	WRITE(6,600)	FETRA	295
	DO 370 I=1,NE	FETRA	296
	NBE1=NBED(I)	FETRA	297
	WRITE(6,601) I,BED(I),GBA(I,NBE1),GBB(I,NBE1),GBC(I,NBE1),GBD(I,	FETRA	298
	1 NBE1),GBE(I,NBE1),GBF(I,NBE1),GBG(I,NBE1)	FETRA	299
	XX=BED(I)-XYSO(I)	FETRA	300
	NBED1=NBED(I)-1	FETRA	301
	DO 370 LM=1,NBED1	FETRA	302
	XY=XX=BDIV(I)*FLOAT(LM=1)	FETRA	303
	LN=NBED(I)-LM	FETRA	304
	WRITE(6,602)XY,GBA(I,LN),GBB(I,LN),GBC(I,LN),GBD(I,LN),GBE(I,LN),	FETRA	305
	1 GBF(I,LN),GBG(I,LN)	FETRA	306
370	CONTINUE	FETRA	307
	WRITE(9) (I,BED(I),GBA(I,4),GBB(I,4),GBC(I,4),GBD(I,4),	FETRA	308
	1 GBE(I,4),GBF(I,4),GBG(I,4),I=1,NE)	FETRA	309
380	CONTINUE	FETRA	310
C	WRITE(6,700) ID,K,TSPEC,TTRANS,TBNDRY,TAMSOL	FETRA	311
C700	FORMAT(" TOTAL TIME = ID,K,SPEC,TRANS,BNDRYS,AMSOL ="	FETRA	312
C	1 2I5,4F12.4)	FETRA	313
C	M3 SPECIL	FETRA	314
C 999	CONTINUE	FETRA	315
	IF (K.GE.NTP) GO TO 350	FETRA	316
	K=K+1	FETRA	317
	KKK=KKK+1	FETRA	318
	GO TO 100	FETRA	319
350	CONTINUE	FETRA	320
	K = K+1	FETRA	321
	IF(STORE.AND.K.GE.NTP) WRITE(2) AT,K,C,NBED,XYSO,BED,GBA,GBB,GBC,	FETFIX1	11
	1 GBD,GBE,GBF,GBG	FETRA	323
	STOP	FETRA	324
225	FORMAT(9X,"TIME SEGMENT NO.",15,/,9X,"TIME STEP SIZE",4X,1PE11	FETRA	325
	1.4,/,9X,"COMPUTED TIME PLANE",1X,1PE11.4,/,9X,"SIMU. SUBSTANCE ID="	FETRA	326
	2",3X,15,//)	FETRA	327
230	FORMAT(//,10X,"LIST THE RESULTS FOR EACH NODE",//)	FETRA	328
235	FORMAT(//)	FETRA	329
240	FORMAT(10X,"NODE",5X,"CONCENTRATION OF SUBSTANCE ID=",I3,/))	FETRA	330
270	FORMAT(1M1,//,10X,"STEADY STATE SOLUTION",//)	FETRA	331
280	FORMAT(1M1,//,10X,"TIME DEPENDENT SOLUTION",//)	FETRA	332
290	FORMAT(1M1)	FETRA	333
	310 FORMAT((10X,6(15,1PE15.7)))	FETRA	334
	320 FORMAT(10X,"NODE",5X,"TOTAL SEDIMENT CONCENTRATION, KGF/M**3",/)	FETRA	335
	325 FORMAT(10X,"NODE",5X,"TOTAL PARTICULATE CONTAMINANT CONCENTRATION	FETRA	336
	\$ ATTACHED TO SEDIMENT",/)	FETRA	337
	326 FORMAT(10X,"NODE",5X,"TOTAL PARTICULATE AND DISSOLVED CONTAMINANT	FETRA	338
	\$ CONCENTRATION",/)	FETRA	339
	330 FORMAT(10X,"NODE",5X,"WEIGHTED AVERAGE PARTICULATE CONTAMINANT CON	FETRA	340
	\$ CENTRATION ATTACHED TO SEDIMENT",/)	FETRA	341
600	FORMAT(9X,"I",7X,"BED",12X,"GBA",12X,"GBB",12X,"GBC",12X,"GBD",12	FETRA	342
	1X,"GBE",12X,"GBF",12X,"GBG",/)	FETRA	343
601	FORMAT(5X,15,/,10X,8(1PE15.7))	FETRA	344
602	FORMAT(10X,8(1PE15.7))	FETRA	345

	END	FETRA	346
	SUBROUTINE ACOEFS(M,AREA)	ACOEFS	2
C	THIS SUBROUTINE COMPUTES THE ELEMENT COEFFICIENTS FORMING THE	ACOEFS	3
C	MATRICES OF THE M-TH ELEMENT.	ACOEFS	4
	COMMON /BLK7/NOD(240,6),X(240),Y(240)	ACOEFS	5
	COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHO*AT,	ACOEFS	6
1	AKP(3)	ACOEFS	7
C		ACOEFS	8
	N1=NOD(M,1)	ACOEFS	9
	N2=NOD(M,2)	ACOEFS	10
	N3=NOD(M,3)	ACOEFS	11
C		ACOEFS	12
	ACOF(1)=X(N3)-X(N2)	ACOEFS	13
	ACOF(2)=X(N1)-X(N3)	ACOEFS	14
	ACOF(3)=X(N2)-X(N1)	ACOEFS	15
	ACOF(4)=Y(N2)-Y(N3)	ACOEFS	16
	ACOF(5)=Y(N3)-Y(N1)	ACOEFS	17
	ACOF(6)=Y(N1)-Y(N2)	ACOEFS	18
C		ACOEFS	19
C	COMPUTE THE AREA OF M-TH TRIANGLE FROM THE LOCAL COORDINATES.	ACOEFS	20
C		ACOEFS	21
	AREA=(ACOF(3)*ACOF(5)-ACOF(2)*ACOF(6))/2.	ACOEFS	22
C		ACOEFS	23
	IF (AREA.GT.0.) GO TO 100	ACOEFS	24
	WRITE (6,110) AREA,M	ACOEFS	25
	WRITE (6,115) N1,N2,N3	ACOEFS	26
115	FORMAT (" N1,N2,N3 ",3I5)	ACOEFS	27
	WRITE (6,120) X(N3),X(N2)	ACOEFS	28
	WRITE (6,120) X(N1),X(N3)	ACOEFS	29
	WRITE (6,120) X(N2),X(N1)	ACOEFS	30
	WRITE (6,125) Y(N2),Y(N3)	ACOEFS	31
	WRITE (6,125) Y(N3),Y(N1)	ACOEFS	32
	WRITE (6,125) Y(N1),Y(N2)	ACOEFS	33
120	FORMAT (" X ",1P2E14,4)	ACOEFS	34
125	FORMAT (" Y ",1P2E14,4)	ACOEFS	35
	WRITE (6,130) ACOF(3),ACOF(5),ACOF(2),ACOF(6)	ACOEFS	36
130	FORMAT (" ACOF 3,5,2,6 ",1P4E14,4)	ACOEFS	37
	STOP	ACOEFS	38
C		ACOEFS	39
100	CONTINUE	ACOEFS	40
	RETURN	ACOEFS	41
110	FORMAT (//,10X,"NEGATIVE OR ZERO AREA ",1PE12,4," ELEMENT",15)	ACOEFS	42
	END	ACOEFS	43
	SUBROUTINE AMSOL(NROW,NCOL,ID,T)	ACOEFS	44
C		AMSOL	2
C**	THIS SUBROUTINE USES A NUMERICAL APPROXIMATION TO	AMSOL	3
C**	SOLVE THE SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS.	AMSOL	4
C**	STYP=0. IMPLIES AN EXPLICIT SOLUTION	AMSOL	5
C**	STYP=.5 IMPLIES A CRANK-NICHOLSON SOLUTION	AMSOL	6
C**	STYP=1. IMPLIES AN IMPLICIT SOLUTION	AMSOL	7
C		AMSOL	8
	COMMON /BLK1/CT(240),C(9,240)	AMSOL	9
	COMMON /BLK2/P(240,86),F(240)	AMSOL	10
	LEVEL 2,P,F	AMSOL	11
	COMMON /BLK3/S(240,86)	FETFIX4	5
	LEVEL 2,S	AMSOL	12
	COMMON /BLK4/R(240),RPAST(240,7),NOOBET,BETA1,AREA1	FETFIX4	6
	LEVEL 2,R,RPAST,NOOBET,BETA1,AREA1	AMSOL	13
	COMMON /BLK16/ STYP	FETFIX4	7
	COMMON /BLK17/QLATE(240),QSARA(7,100),QPNT(7,240),CRATE(3,100),	AMSOL	14
1	CD(4,240)	AMSOL	15
		AMSOL	16

	LEVEL 2, GLATE, QSARA, QPNT, CRATE, CO	FETFIX4	8
C	NBND=2*NCOL-1	AMSOL	17
	DO 120 J=1, NBND	AMSOL	18
	DO 120 I=1, NROW	AMSOL	19
	PBAR=P(I, J)+T*S(I, J)*STYP	AMSOL	20
	SBAR=P(I, J)-T*S(I, J)*(1.-STYP)	AMSOL	21
	P(I, J)=PBAR	AMSOL	22
	S(I, J)=SBAR	AMSOL	23
120	CONTINUE	AMSOL	24
C		AMSOL	25
C	IF (ID.LT.5) GO TO 302	AMSOL	26
C	WRITE(6, 399)	AMSOL	27
C	WRITE(6, 400) (R(J), J=1, NROW)	AMSOL	28
C	WRITE(6, 401)	AMSOL	29
C	DO 300 I=1, NROW	AMSOL	30
C	WRITE(6, 400) (S(I, J), J=1, NBND)	AMSOL	31
C 300	CONTINUE	AMSOL	32
C	WRITE(6, 402)	AMSOL	33
C	DO 301 I=1, NROW	AMSOL	34
C	WRITE(6, 400) (P(I, J), J=1, NBND)	AMSOL	35
C 301	CONTINUE	AMSOL	36
C 302	CONTINUE	AMSOL	37
C		AMSOL	38
C	CALL COMB(NROW, NCOL, ID)	AMSOL	39
	DO 130 I=1, NROW	AMSOL	40
130	F(I)=F(I)+T*R(I)*STYP+T*RPAST(I, ID)*(1.-STYP)	AMSOL	41
C		AMSOL	42
C		AMSOL	43
C	WRITE(6, 406)	AMSOL	44
C 406	FORMAT(//, 10X, "R MATRIX", /)	AMSOL	45
C	WRITE(6, 405) (R(I), I=1, NROW)	AMSOL	46
C	WRITE(6, 407)	AMSOL	47
C 407	FORMAT(//, 10X, "RPAST MATRIX", /)	AMSOL	48
C	WRITE(6, 405) (RPAST(I, ID), I=1, NROW)	AMSOL	49
C		AMSOL	50
C		AMSOL	51
C	IF (ID.LT.5) GO TO 307	AMSOL	52
C	WRITE(6, 403)	AMSOL	53
C	DO 999 I=1, NROW	AMSOL	54
C	WRITE(6, 405) (P(I, J), J=1, NBND)	AMSOL	55
C 999	CONTINUE	AMSOL	56
C	WRITE(6, 404)	AMSOL	57
C	WRITE(6, 405) (F(J), J=1, NROW)	AMSOL	58
C 307	CONTINUE	AMSOL	59
C		AMSOL	60
C		AMSOL	61
C	CALL BANSOL(NROW, NCOL, P, F)	AMSOL	62
	DO 140 I=1, NROW	AMSOL	63
	CT(I)=F(I)	AMSOL	64
140	CONTINUE	AMSOL	65
	RETURN	AMSOL	66
399	FORMAT(//, 10X, "R MATRIX", /)	AMSOL	67
400	FORMAT(12(1PE10.3))	AMSOL	68
401	FORMAT(//, 10X, "S MATRIX", /)	AMSOL	69
402	FORMAT(//, 10X, "P MATRIX", /)	AMSOL	70
403	FORMAT(//, 10X, "P MATRIX FOR PX = F", /)	AMSOL	71
404	FORMAT(//, 10X, "F MATRIX FOR PX = F", /)	AMSOL	72
405	FORMAT(12(1PE10.3))	AMSOL	73
	END	AMSOL	74
	SUBROUTINE BANSOL(NROW, NCOL, P, F)	AMSOL	75
		AMSOL	76
		BANSOL	2

C		BANSOL	3
C	THIS ROUTINE SOLVES THE LINEAR SYSTEM OF EQUATIONS (P) X = (F)	BANSOL	4
C	WHERE (P) IS BANDED AND UNSYMMETRIC. THE ALGORITHM EMPLOYED IS	BANSOL	5
C	GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING.	BANSOL	6
C		BANSOL	7
	LEVEL 2,P,F	FETFIX4	9
	DIMENSION P(240,86),F(240)	BANSOL	8
	DATA EPSI/1.E-36/	BANSOL	9
C		BANSOL	10
	N=NCOL	BANSOL	11
	L=NCOL	BANSOL	12
	M=2*NCOL-1	BANSOL	13
	N1=N-1	BANSOL	14
C		BANSOL	15
C	FORWARD ELIMINATION WITH PARTIAL PIVOTING.	BANSOL	16
C		BANSOL	17
	DO 160 I=1,N1	BANSOL	18
	IPIV=I	BANSOL	19
	IRE=I+1	BANSOL	20
	DO 100 IR=IRE,L	BANSOL	21
	IF (ABS(P(IR,1)) .LE. ABS(P(I,1))) GO TO 100	BANSOL	22
	IPIV=IR	BANSOL	23
100	CONTINUE	BANSOL	24
C		BANSOL	25
	IF (IPIV .EQ. I) GO TO 120	BANSOL	26
	T=F(I)	BANSOL	27
	F(I)=F(IPIV)	BANSOL	28
	F(IPIV)=T	BANSOL	29
	DO 110 J=1,M	BANSOL	30
	T=P(I,J)	BANSOL	31
	P(I,J)=P(IPIV,J)	BANSOL	32
110	P(IPIV,J)=T	BANSOL	33
120	CONTINUE	BANSOL	34
	IF (ABS(P(I,1)) .LE. EPSI) GO TO 135	BANSOL	35
	F(I)=F(I)/P(I,1)	BANSOL	36
	DO 130 J=2,M	BANSOL	37
	P(I,J)=P(I,J)/P(I,1)	BANSOL	38
130	CONTINUE	BANSOL	39
135	CONTINUE	BANSOL	40
	DO 150 IR=IRE,L	BANSOL	41
	T=P(IR,1)	BANSOL	42
	F(IR)=F(IR)-T*F(I)	BANSOL	43
	DO 140 J=2,M	BANSOL	44
140	P(IR,J-1)=P(IR,J)-T*P(I,J)	BANSOL	45
150	P(IR,M)=0.0	BANSOL	46
	IF (L .EQ. N) GO TO 160	BANSOL	47
	L=L+1	BANSOL	48
160	CONTINUE	BANSOL	49
C		BANSOL	50
C	BACK SUBSTITUTION	BANSOL	51
C		BANSOL	52
	IF (ABS(P(N,1)) .GE. EPSI) GO TO 400	BANSOL	53
	WRITE (6,920)N,F(N),P(N,1)	BANSOL	54
400	CONTINUE	BANSOL	55
	F(N)=F(N)/P(N,1)	BANSOL	56
	JM=2	BANSOL	57
	DO 180 ICE=1,N1	BANSOL	58
	IR=N-ICE	BANSOL	59
	DO 170 J=2,JM	BANSOL	60
	IRM1=IR-1+J	BANSOL	61
	F(IR)=F(IR)-P(IR,J)*F(IRM1)	BANSOL	62
170	CONTINUE	BANSOL	63

	IF (JM .EQ. M) GO TO 180	BANSOL	54
	JM=JM+1	BANSOL	55
180	CONTINUE	BANSOL	56
C	RETURN	BANSOL	57
920	FORMAT(10X,"N=",I4,5X,"F(N)=",E10,3,5X,"P(N,1)=",E10,3)	BANSOL	58
	END	BANSOL	59
	SUBROUTINE BEDHIS(M,T)	BEDHIS	70
C		BEDHIS	2
C	THIS SUBROUTINE KEEPS A RECORD OF BED HISTORY, INCLUDING BED	BEDHIS	3
C	SURFACE ELEVATION, RATIO OF BED SEDIMENT WEIGHT FRACTIONS,	BEDHIS	4
C	AND ASSOCIATED CHEMICAL'S CONCENTRATIONS IN THE BED	BEDHIS	5
C		BEDHIS	6
C	ILAYR(J).....NO. OF LAYERS COMPLETELY SCoured BY EACH RESPECTIVE	BEDHIS	7
C	SEDIMENT. ILAYR(J)=-1 FOR DEPOSITION	BEDHIS	8
C	ARAD.....AMOUNT OF CHEMICALS LEFT IN THE TOP BED LAYER	BEDHIS	9
C	ALEFT.....AMOUNT OF SEDIMENT LEFT IN THE TOP BED LAYER	BEDHIS	10
C	XDTOP.....TOTAL WEIGHT OF THE SEDIMENT IN THE TOP BED LAYER	BEDHIS	11
C	DRAD.....AMOUNT OF CHEMICALS DEPOSITED PER TIME STEP	BEDHIS	12
C	XND.....TOTAL WEIGHT OF THE SEDIMENT IN A BED LAYER	BEDHIS	13
C		BEDHIS	14
	COMMON /BLK1/CT(240),C(9,240)	BEDHIS	15
	COMMON /BLK7/NOO(240,6),X(240),Y(240)	BEDHIS	16
	COMMON /BLK11/D50(3,100),BD50(100),SR(3,100),SD(3,100)	BEDHIS	17
	COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHOMAT,	BEDHIS	18
1	AKP(3)	BEDHIS	19
1	COMMON /BLK14/GBA(100,10),GBB(100,10),GBC(100,10),GBD(100,10),	BEDHIS	20
1	GBE(100,10),GBF(100,10),GBG(100,10),POR	BEDHIS	21
	COMMON /BLK15/ILAYR(100,3),XYS0(100),BDIV(100),NBED(100),BED(100),	BEDHIS	22
1	RHOSED(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)	BEDHIS	23
1	COMMON /BLK17/GLATE(240),QSARA(7,100),QPNT(7,240),CRATE(3,100),	BEDHIS	24
1	CD(4,240)	BEDHIS	25
	LEVEL 2,QLATE,QSARA,QPNT,CRATE,CD	FETFIX4	10
	DIMENSION ARAD(3),ALEFT(3),B2(3),B3(3),GBJ(3),CS(3)	BEDHIS	27
	DATA EPSI/1.0E-10/	BEDHIS	28
C		BEDHIS	29
C		BEDHIS	30
	IN=ILAYR(M,1)	BEDHIS	31
	INN=MINO(IN+1,NBED(M))	BEDHIS	32
	IP=ILAYR(M,2)	BEDHIS	33
	IQ=ILAYR(M,3)	BEDHIS	34
C		BEDHIS	35
	GBJ(1) = GBD(M,NBED(M))	BEDHIS	36
	GBJ(2) = GBE(M,NBED(M))	BEDHIS	37
	GBJ(3) = GBF(M,NBED(M))	BEDHIS	38
	CS(1) = RSAV1(M)	BEDHIS	39
	CS(2) = RSAV2(M)	BEDHIS	40
	CS(3) = RSAV3(M)	BEDHIS	41
C		BEDHIS	42
	DO 100 II=1,3	BEDHIS	43
	ARAD(II) = 0.0	BEDHIS	44
	ALEFT(II)=0.0	BEDHIS	45
100	CONTINUE	BEDHIS	46
C		BEDHIS	47
	TOTAL = GBA(M,NBED(M))/RHOSED(1)+GBB(M,NBED(M))/RHOSED(2)+	BEDHIS	48
	\$ GBC(M,NBED(M))/RHOSED(3)	BEDHIS	49
	XND=(1.-POR)/TOTL	BEDHIS	50
	XNTTMP=(XNT(M,1)+XNT(M,2)+XNT(M,3))/XND	BEDHIS	51
	IF(SR(1,M)+SR(2,M)+SR(3,M) .GT. 0.0) GO TO 110	BEDHIS	52
	IF(ABS(XNTTMP-BDIV(M)) .GE. EPSI) GO TO 110	BEDHIS	53
	XNT(M,1)=0.	BEDHIS	54
	XNT(M,2)=0.	BEDHIS	55

	XNT(M,3)=0.	BEDHIS	56
	NBED(M)=NBED(M)+1	BEDHIS	57
	DO 105 I=1,3	BEDHIS	58
	ALEFT(I)=SD(I,4)*T	BEDHIS	59
	ARAD(I)=-((CS(I)+CRATE(I,M))*T	BEDHIS	60
105	CONTINUE	BEDHIS	61
	GO TO 270	BEDHIS	62
110	IF (ILAYR(M,1).LT.0) IN=0	BEDHIS	63
	ALEFT(1)=XNT(M,1)+SD(1,M)*T	BEDHIS	64
	ARAD(1)=XNT(M,1)*GBD(M,NBED(M)-IN)-(CS(1)+CRATE(1,M))*T	BEDHIS	65
	IF (ILAYR(M,2).LT.0) IP=0	BEDHIS	66
	ALEFT(2)=XNT(M,2)+SD(2,M)*T	BEDHIS	67
	ARAD(2)=XNT(M,2)*GBE(M,NBED(M)-IP)-(CS(2)+CRATE(2,M))*T	BEDHIS	68
	IF (ILAYR(M,3).LT.0) IQ=0	BEDHIS	69
	ALEFT(3)=XNT(M,3)+SD(3,M)*T	BEDHIS	70
	ARAD(3)=XNT(M,3)*GBF(M,NBED(M)-IQ)-(CS(3)+CRATE(3,M))*T	BEDHIS	71
	IF (ILAYR(M,1).LT.1 .OR. ILAYR(M,2).GE. ILAYR(M,1)) GO TO 150	BEDHIS	72
	IP2=ILAYR(M,2)+2	BEDHIS	73
	DO 140 IT=IP2,INN	BEDHIS	74
	IF (ILAYR(M,2).LT.0.AND.IT.EQ.1) GO TO 140	BEDHIS	75
	IU=NBED(M)-IT+1	BEDHIS	76
	XND=(GBA(M,IU)/RHOSED(1)+GBB(M,IU)/RHOSED(2)+GBC(M,IU)/RHOSED(3))	BEDHIS	77
	XND=(1.0-POR)/XND	BEDHIS	78
	XNT(M,2)=XND*GBB(M,IU)*BDIV(M)	BEDHIS	79
	ALEFT(2)=ALEFT(2)+XNT(M,2)	BEDHIS	80
	ARAD(2)=ARAD(2)+XNT(M,2)*GBE(M,IU)	BEDHIS	81
140	CONTINUE	BEDHIS	82
150	CONTINUE	BEDHIS	83
	IF (ILAYR(M,1).LT.1 .OR. ILAYR(M,3).GE. ILAYR(M,1)) GO TO 270	BEDHIS	84
	IQ2=ILAYR(M,3)+2	BEDHIS	85
	DO 220 IR=IQ2,INN	BEDHIS	86
	IF (ILAYR(M,3).LT.0.AND.IR.EQ.1) GO TO 220	BEDHIS	87
	IS=NBED(M)-IR+1	BEDHIS	88
	XND=(GBA(M,IS)/RHOSED(1)+GBB(M,IS)/RHOSED(2)+GBC(M,IS)/RHOSED(3))	BEDHIS	89
	XND=(1.0-POR)/XND	BEDHIS	90
	XNT(M,3)=XND*GBC(M,IS)*BDIV(M)	BEDHIS	91
	ALEFT(3)=ALEFT(3)+XNT(M,3)	BEDHIS	92
	ARAD(3)=ARAD(3)+XNT(M,3)*GBF(M,IS)	BEDHIS	93
220	CONTINUE	BEDHIS	94
C		BEDHIS	95
C		BEDHIS	96
C	XM...THICKNESS OF BED TOP LAYER WHICH WILL BE SET TO XYSO(M)	BEDHIS	97
270	CONTINUE	BEDHIS	98
	B1=ALEFT(1)+ALEFT(2)+ALEFT(3)	BEDHIS	99
	IF (B1.EQ.0.0.AND.ILAYR(M,1).LT.0) RETURN	BEDHIS	100
	XM = ALEFT(1)/RHOSED(1)+ALEFT(2)/RHOSED(2)+ALEFT(3)/RHOSED(3)	BEDHIS	101
	XM = XM/(1.0-POR)	BEDHIS	102
	Iw=0	BEDHIS	103
	REMAIN = XM	BEDHIS	104
280	CONTINUE	BEDHIS	105
	IF (REMAIN.LE.BDIV(M)) GO TO 300	BEDHIS	106
	Iw=Iw+1	BEDHIS	107
	REMAIN = REMAIN - BDIV(M)	BEDHIS	108
	GO TO 280	BEDHIS	109
300	CONTINUE	BEDHIS	110
	Iw=Iw+1	BEDHIS	111
	NBED(M) = NBED(M) - INN	BEDHIS	112
	IF (ILAYR(M,1).LT.0) NBED(M) = NBED(M) - 1	BEDHIS	113
	NBED1 = NBED(M) + 1	BEDHIS	114
	NBED2 = NBED(M) + Iw	BEDHIS	115
	DO 360 IY=NBED1,NBED2	BEDHIS	116
	DO 340 Ix=1,3	BEDHIS	117

	B2(IX)=0.0	BEDHIS	116
	IF (B1.NE.0.0) B2(IX)=ALEFT(IX)/B1	BEDHIS	119
	B3(IX)=0.0	BEDHIS	120
	IF (ALEFT(IX).GT.0.0.AND.ARAO(IX).GT.0.0) B3(IX)=	BEDHIS	121
	1 ARAO(IX)/ALEFT(IX)	BEDHIS	122
340	CONTINUE	BEDHIS	123
	T0N#ET#B2(1)*B3(1) + B2(2)*B3(2) + B2(3)*B3(3)	BEDHIS	124
	GBA(M,IY)#B2(1)	BEDHIS	125
	GBB(M,IY)#B2(2)	BEDHIS	126
	GBB(M,IY)#B2(3)	BEDHIS	127
	GBD(M,IY)#B3(1)* EXP(-ALMBDA*T)	BEDHIS	128
	GBE(M,IY)#B3(2)* EXP(-ALMBDA*T)	BEDHIS	129
	GBF(M,IY)#B3(3)* EXP(-ALMBDA*T)	BEDHIS	130
	GBG(M,IY)#T0N#ET* EXP(-ALMBDA*T)	BEDHIS	131
360	CONTINUE	BEDHIS	132
C		BEDHIS	133
	NBED(M)#NBED(M)+I#	BEDHIS	134
	XYSO(M)#REMAIN	BEDHIS	135
	BED(M)=(NBED(M)-1)*BDIV(M)+XYSO(M)	BEDHIS	136
	IF (NBED(M).LE.50) GO TO 400	BEDHIS	137
	#RITE(6,200) M,NBED(M)	BEDHIS	138
	STOP	BEDHIS	139
400	CONTINUE	BEDHIS	140
	RETURN	BEDHIS	141
200	FORMAT(2X,"DEPOSITION EXCEEDS PERMISSIBLE BED DEPTH IN BEDHIS",/,	BEDHIS	142
	15X,"M,NBED=",215)	BEDHIS	143
	END	BEDHIS	144
	SUBROUTINE BNDRYS (NROW,NCOL,ID)	BNDRYS	2
C		BNDRYS	3
C	THIS ROUTINE SETS THE BOUNDARY CONDITIONS IN THE GLOBAL MATRICES,	BNDRYS	4
C	ELIMINATES THE EQUATIONS FOR BOUNDARY NODES AND THEN REFORMS THE	BNDRYS	5
C	REDUCED SYSTEM MATRICES.	BNDRYS	6
C		BNDRYS	7
	COMMON /BLK1/CT(240),C(9,240)	BNDRYS	8
	COMMON /BLK2/P(240,86),F(240)	BNDRYS	9
	LEVEL 2,P,F	FETFIX4	11
	COMMON /BLK3/S(240,86)	BNDRYS	10
	LEVEL 2,S	FETFIX4	12
	COMMON /BLK4/R(240), RPAST(240,7), NODBET, BETA1, AREA1	BNDRYS	11
	LEVEL 2,R,RPAST,NODBET,BETA1,AREA1	FETFIX4	13
	COMMON /BLK8/LBC,LBC,MBC(7,120),NBC(7,120),OBC(7,120),BC(7,120),	BNDRYS	12
	1 KODE(10)	BNDRYS	13
C		BNDRYS	14
C	SET THE SPECIFIED BOUNDARY CONDITONS IN THE MATRICES	BNDRYS	15
C		BNDRYS	16
	NBND#2*NCOL-1	BNDRYS	17
	IF (LBC.LE.0) GO TO 190	BNDRYS	18
	DO 110 I#1,LBC	BNDRYS	19
	NN#NBC(ID,I)	BNDRYS	20
	R(NN)#0.0	BNDRYS	21
	DO 110 NC#1,NBND	BNDRYS	22
	P(NN,NC)#0.0	BNDRYS	23
	S(NN,NC)#0.0	BNDRYS	24
	NR#NN-NC+NCOL	BNDRYS	25
	IF (NR.LE.0.OR.NR.GT.NROW) GO TO 110	BNDRYS	26
	R(NR)#R(NR)-S(NR,NC)*BC(ID,I)	BNDRYS	27
	P(NR,NC)#0.0	BNDRYS	28
	S(NR,NC)#0.0	BNDRYS	29
110	CONTINUE	BNDRYS	30
C		BNDRYS	31
C	ELIMINATE THE EQUATIONS FOR THE BOUNDARY CONDITION NODES AND	BNDRYS	32
C	REFORM THE GLOBAL MATRICES.	BNDRYS	33

C	DO 180 L=1,LBC	BNDRYS	34
	NN=NBC(ID,L)=L+1	BNDRYS	35
	NRU=NR0W=1	BNDRYS	36
	IF (NN.GT.NR0W) GO TO 180	BNDRYS	37
	DO 120 NR=NN,NR0W	BNDRYS	38
	C(ID,NR)=C(ID,NR+1)	BNDRYS	39
	R(NR)=R(NR+1)	BNDRYS	40
	DO 120 NC=1,NCOL	BNDRYS	41
	MC=NCOL+NC-1	BNDRYS	42
	P(NR,MC)=P(NR+1,MC)	BNDRYS	43
	S(NR,MC)=S(NR+1,MC)	BNDRYS	44
	IF (NR.LT.NR0W) GO TO 120	BNDRYS	45
	P(NR+1,MC)=0.0	BNDRYS	46
	S(NR+1,MC)=0.0	BNDRYS	47
120	CONTINUE	BNDRYS	48
C		BNDRYS	49
	DO 140 NC=2,NCOL	BNDRYS	50
	MC=NCOL-NC+1	BNDRYS	51
	NL=NN+NC-1	BNDRYS	52
	IF (NL.GT.NR0W) GO TO 140	BNDRYS	53
	DO 130 NR=NL,NR0W	BNDRYS	54
	P(NR,MC)=P(NR+1,MC)	BNDRYS	55
	S(NR,MC)=S(NR+1,MC)	BNDRYS	56
	IF (NR.LT.NR0W) GO TO 130	BNDRYS	57
	P(NR+1,MC)=0.0	BNDRYS	58
	S(NR+1,MC)=0.0	BNDRYS	59
130	CONTINUE	BNDRYS	60
140	CONTINUE	BNDRYS	61
C		BNDRYS	62
	IF (NN.EQ.1) GO TO 180	BNDRYS	63
C		BNDRYS	64
	NCL=NCOL-1	BNDRYS	65
	DO 160 LC=2,NCL	BNDRYS	66
	NR=NN-LC+1	BNDRYS	67
	IF (NR.LE.0) GO TO 160	BNDRYS	68
	DO 150 NC=LC,NCL	BNDRYS	69
	MC=NCOL+NC-1	BNDRYS	70
	P(NR,MC)=P(NR,MC+1)	BNDRYS	71
	S(NR,MC)=S(NR,MC+1)	BNDRYS	72
	IF (NC.LT.NCL) GO TO 150	BNDRYS	73
	P(NR,NBND)=0.0	BNDRYS	74
	S(NR,NBND)=0.0	BNDRYS	75
C	P(NR,MC+1)=0.0	BNDRYS	76
C	S(NR,MC+1)=0.0	BNDRYS	77
150	CONTINUE	BNDRYS	78
160	CONTINUE	BNDRYS	79
C		BNDRYS	80
	NRL=NN+NCOL-3	BNDRYS	81
	DO 170 NR=NN,NRL	BNDRYS	82
	NCL=NN-NR+NCOL-2	BNDRYS	83
	DO 170 NC=1,NCL	BNDRYS	84
	P(NR,NC+1)=P(NR+1,NC)	BNDRYS	85
	S(NR,NC+1)=S(NR+1,NC)	BNDRYS	86
	IF (NC.NE.1) GO TO 170	BNDRYS	87
	P(NR+1,1)=0.0	BNDRYS	88
	S(NR+1,1)=0.0	BNDRYS	89
170	CONTINUE	BNDRYS	90
180	CONTINUE	BNDRYS	91
C		BNDRYS	92
C	PERFORM A LEFT SHIFT OF THE (P)AND (S) MATRIXFOR COMB INNER	BNDRYS	93
C	PRODUCT	BNDRYS	94
		BNDRYS	95

C		BNDRYS	96
190	L1=NCOL=1	BNDRYS	97
	DO 210 IR=1,L1	BNDRYS	98
	LR=NCOL=IR	BNDRYS	99
	DO 210 I=1,LR	BNDRYS	100
	DO 200 J=2,NBND	BNDRYS	101
	P(IR,J=1)=P(IR,J)	BNDRYS	102
	S(IR,J=1)=S(IR,J)	BNDRYS	103
200	CONTINUE	BNDRYS	104
	NP1=NROW+1-IR	BNDRYS	105
	MP1=NBND+1-I	BNDRYS	106
	P(IR,NBND)=0.0	BNDRYS	107
	S(IR,NBND)=0.0	BNDRYS	108
	P(NP1,MP1)=0.0	BNDRYS	109
	S(NP1,MP1)=0.0	BNDRYS	110
210	CONTINUE	BNDRYS	111
C		BNDRYS	112
	RETURN	BNDRYS	113
	END	BNDRYS	114
	SUBROUTINE CLAY (M, ID, T)	CLAY	2
C		CLAY	3
C	THIS SUBROUTINE COMPUTES THE AMOUNT OF RESUSPENSION, SR, OR	CLAY	4
C	DEPOSITION, SO, OF CLAY	CLAY	5
C	THIS ROUTINE DECIDES NO. OF BED LAYERS TO BE SCOURED	CLAY	6
C	CRSTRS(ID, M), CRITICAL SHEAR STRESS FOR SCOURING, ID=1 FOR CLAY	CLAY	7
C	COSTRS(ID, M), CRITICAL SHEAR STRESS FOR DEPOSITION	CLAY	8
C	WS(ID, M)..... FALL VELOCITY OF SEDIMENT	CLAY	9
C	ERODA(ID, M) .. ERODABILITY COEFFICIENT	CLAY	10
C	RSV1..... CHEMICAL AMOUNT SCOURED FROM BED DURING ONE TIME STEP	CLAY	11
C	RS..... SEDIMENT AMOUNT SCOURED FROM BED DURING ONE TIME STEP	CLAY	12
C		CLAY	13
	COMMON /BLK1/CT(240), C(9, 240)	CLAY	14
	COMMON /BLK6/VX(240), VY(240), H(240), STRESS(3)	CLAY	15
	COMMON /BLK7/NOD(240, 6), X(240), Y(240)	CLAY	16
	COMMON /BLK11/OS0(3, 100), BDS0(100), SR(3, 100), SO(3, 100)	CLAY	17
	COMMON /BLK13/WS(3, 100), CRSTRS(3, 100), COSTRS(3, 100), ERODA(3, 100)	CLAY	18
	COMMON /BLK14/GBA(100, 10), GBB(100, 10), GBC(100, 10), GBD(100, 10),	CLAY	19
1	GBE(100, 10), GBF(100, 10), GBG(100, 10), POR	CLAY	20
	COMMON /BLK15/ILAYR(100, 3), XYS0(100), BDIV(100), NBED(100), BED(100),	CLAY	21
1	RHOSED(3), XNT(100, 3), RSV1(100), RSV2(100), RSV3(100)	CLAY	22
	DIMENSION S1(3)	CLAY	23
	DATA EPSI / 1.0E-10 /	CLAY	24
	SD(ID, M)=0.0	CLAY	25
	SR(ID, M)=0.0	CLAY	26
	RSV1(M)=0.	CLAY	27
	RS=0.0	CLAY	28
	ILAYR(M, ID)=0	CLAY	29
	TOTAL = GBA(M, NBED(M))/RHOSED(1)+GBB(M, NBED(M))/RHOSED(2)+	CLAY	30
	\$ GBC(M, NBED(M))/RHOSED(3)	CLAY	31
	XOTOP=(1.-POR)/TOTAL	CLAY	32
	XNT(M, ID)=XYS0(M)*GBA(M, NBED(M))*XOTOP	CLAY	33
	DO 999 II=1, 3	CLAY	34
	S1(II)=0.0	CLAY	35
	IF (STRESS(II), LE, CRSTRS(ID, M), AND, STRESS(II), GE, COSTRS(ID, M)) GO	CLAY	36
	1 TO 999	CLAY	37
	IF (STRESS(II), GT, CRSTRS(ID, M)) GO TO 100	CLAY	38
C	DEPOSITION	CLAY	39
	S1(II)=WS(ID, M)*C(ID, NOD(M, II))*(1.0-(STRESS(II)/COSTRS(ID, M))	CLAY	40
	\$)*(-1)	CLAY	41
	GO TO 999	CLAY	42
C		CLAY	43
C		CLAY	44

C	RESUSPENSION	CLAY	45
C		CLAY	46
100	S1(II)=ERODA(ID,M)*(STRESS(II)/CRSTRS(ID,M)=1.0)	CLAY	47
999	CONTINUE	CLAY	48
	S2=(S1(1)+S1(2)+S1(3))/3.0	CLAY	49
	IF (S2) 300,305,302	CLAY	50
300	CONTINUE	CLAY	51
C		CLAY	52
C	DEPOSITION	CLAY	53
C		CLAY	54
	SD(ID,M)=(-1)*S2	CLAY	55
	ILAYR(M, ID)=-1	CLAY	56
	N1 = NOD(M,1)	CLAY	57
	N2 = NOD(M,2)	CLAY	58
	N3 = NOD(M,3)	CLAY	59
	N4 = NOD(M,4)	CLAY	60
	N5 = NOD(M,5)	CLAY	61
	N6 = NOD(M,6)	CLAY	62
	H1 = H(N1)	CLAY	63
	H2 = H(N2)	CLAY	64
	H3 = H(N3)	CLAY	65
	HAVG = (H1+H2+H3)/3.	CLAY	66
	CAVG1 = (C(ID,N1)*H1+C(ID,N2)*H2+C(ID,N3)*H3+C(ID,N4)*(H1+H2)/2.	CLAY	67
1	+C(ID,N5)*(H2+H3)/2.+C(ID,N6)*(H1+H3)/2.)/(6.*HAVG)	CLAY	68
	IF (CAVG1.LE.EPSI) GO TO 305	CLAY	69
	CAVGS = (C(ID+4,N1)*H1+C(ID+4,N2)*H2+C(ID+4,N3)*H3+C(ID+4,N4)*	CLAY	70
1	(H1+H2)/2.+C(ID+4,N5)*(H2+H3)/2.+C(ID+4,N6)*(H1+H3)/2.)/	CLAY	71
2	(6.*HAVG)	CLAY	72
	RSV1(M) = -SD(ID,M)*CAVGS/CAVG1	CLAY	73
	GO TO 305	CLAY	74
302	CONTINUE	CLAY	75
C		CLAY	76
C**	RESUSPENSION	CLAY	77
C		CLAY	78
	SR(ID,M)=S2	CLAY	79
C	TO COMPUTE A NUMBER OF LAYERS SCOURED IN ORDER TO RESUSPEND THE	CLAY	80
C	APPROPRIATE AMOUNT OF COHESIVE SEDIMENT	CLAY	81
C	ASSUME THAT CLAY IS MOST DIFFICULT TO SCOUR	CLAY	82
C	XNT(M)...WEIGHT OF THE SEDIMENT M IN THE TOP BED LAYER IN KG.	CLAY	83
C	NBED(M)...INITIAL NUMBER OF BED LAYERS IN ELEMENT M	CLAY	84
C	BED(M)...INITIAL BED THICKNESS IN ELEMENT M	CLAY	85
C	XYSO(M)...THICKNESS OF THE TOP BED LAYER IN ELEMENT M	CLAY	86
C	GBA(M,J)...WEIGHT FRACTION OF CLAY OR SAND2 OF BED LAYER J IN	CLAY	87
C	ELEMENT M	CLAY	88
C	GBB(M,J) WEIGHT FRACTION OF SILT OF BED LAYER J IN ELEMENT M	CLAY	89
C	GBC(M,J) WEIGHT FRACTION OF SAND OF BED LAYER J IN ELEMENT M	CLAY	90
C	GBD(M,J) CHEMICAL CONCENTRATION IN CLAY OF BED LAYER J IN ELEM. M	CLAY	91
C	PER UNIT WEIGHT OF SEDIMENT	CLAY	92
C	GBE(M,J) CHEMICAL CONCENTRATION IN SILT OF BED LAYER J IN ELEM. M	CLAY	93
C	PER UNIT WEIGHT OF SEDIMENT	CLAY	94
C	GBF(M,J) CHEMICAL CONCENTRATION IN SAND OF BED LAYER J IN ELEM. M	CLAY	95
C	PER UNIT WEIGHT OF SEDIMENT	CLAY	96
C	GBG(M,J) TOTAL CHEMICAL CONCENTRATION IN BED LAYER J IN ELEMENT M	CLAY	97
C	PER UNIT WEIGHT OF SEDIMENT	CLAY	98
C	GBG(M,J)=GBA(M,J)*GBD(M,J)+GBB(M,J)*GBE(M,J)+GBC(M,J)*GBF(M,J)	CLAY	99
C		CLAY	100
	NB = NBED(M)	CLAY	101
	SR(ID,M)=SR(ID,M)*T	CLAY	102
	IF (SR(ID,M).GT.XNT(M, ID)) GO TO 200	CLAY	103
	RSV1(M)=SR(ID,M)*GBD(M,NBED(M))	CLAY	104
	XNT(M, ID)=XNT(M, ID)-SR(ID,M)	CLAY	105
	GO TO 290	CLAY	106

C	ILAYR(J),,NO,OF TOTAL BED LAYERS SCOURED DURING TIME DURATION,T	CLAY	107
C	BDIV(M),,THICKNESS OF BED LAYER IN ELEMENT M	CLAY	108
200	ILAYR(M, ID)=ILAYR(M, ID)+1	CLAY	109
	IF (ILAYR(M, ID),EQ,NBED(M)) GO TO 280	CLAY	110
	RSAV1(M)=RSAV1(M)+XNT(M, ID)*GBD(M, NB)	CLAY	111
	SR(ID, M)=SR(ID, M)-XNT(M, ID)	CLAY	112
	RS=RS+XNT(M, ID)	CLAY	113
	NB=NBED(M)=ILAYR(M, ID)	CLAY	114
	TOTAL=GBA(M, NB)/RHUSED(1)+GBR(M, NB)/RHUSED(2)+GBC(M, NB)/RHUSED(3)	CLAY	115
	XND=(1,-POR)/TOTAL	CLAY	116
	XNT(M, ID)=BDIV(M)*GBA(M, NB)*XND	CLAY	117
	IF (ILAYR(M, ID),LT,(NBED(M)-1)) GO TO 270	CLAY	118
	ILAYR(M, ID)=ILAYR(M, ID)+1	CLAY	119
	GO TO 280	CLAY	120
270	IF (SR(ID, M),GE,XNT(M, ID)) GO TO 200	CLAY	121
280	RSAV1(M)=RSAV1(M)+AMIN1(SR(ID, M), XNT(M, ID))*GBD(M, NB)	CLAY	122
	RS=RS+AMIN1(SR(ID, M), XNT(M, ID))	CLAY	123
	XNT(M, ID)=XNT(M, ID)+AMIN1(SR(ID, M), XNT(M, ID))	CLAY	124
	SR(ID, M)=RS	CLAY	125
290	SR(ID, M)=SR(ID, M)/T	CLAY	126
	RSAV1(M)=RSAV1(M)/T	CLAY	127
305	CONTINUE	CLAY	128
	RETURN	CLAY	129
	END	CLAY	130
	SUBROUTINE COMB(NROW, NCOL, ID)	COMB	2
C		COMB	3
C	THIS SUBROUTINE MULTIPLIES THE UNSYMMETRIC BAND MATRIX (S) TO THE	COMB	4
C	LOAD VECTOR (C) AND STORES THE RESULT IN (F).	COMB	5
	DOUBLE PRECISION DVAR	COMB	6
	COMMON /BLK1/CT(240),C(9,240)	COMB	7
	COMMON /BLK2/P(240,86),F(240)	COMB	8
	LEVEL 2,P,F	FETFIX4	14
	COMMON /BLK3/S(240,86)	COMB	9
	LEVEL 2,S	FETFIX4	15
C		COMB	10
	NBND=2*NCOL-1	COMB	11
	DO 100 I=1, NCOL	COMB	12
	F(I)=0.0	COMB	13
	DO 100 K=1, NBND	COMB	14
	DVAR=F(I)+S(I, K)*C(ID, K)	COMB	15
	F(I)=DVAR	COMB	16
100	CONTINUE	COMB	17
C		COMB	18
	LL=0	COMB	19
	NI=NCOL+1	COMB	20
	IF (NI .GT. NROW) RETURN	COMB	21
	DO 110 I=NI, NROW	COMB	22
	F(I)=0.0	COMB	23
	LL=LL+1	COMB	24
	DO 110 K=1, NBND	COMB	25
	L=LL+K	COMB	26
	IF (L .GT. NROW) GO TO 110	COMB	27
	DVAR=F(I)+S(I, K)*C(ID, L)	COMB	28
	F(I)=DVAR	COMB	29
110	CONTINUE	COMB	30
C	IF (ID.LT.5) GO TO 995	COMB	31
C	WRITE(6,996)	COMB	32
C	WRITE(6,997) (F(I), I=1, NROW)	COMB	33
C	WRITE(6,998)	COMB	34
C	WRITE(6,999) (C(ID, I), I=1, NROW)	COMB	35
C996	FORMAT(/, 10X, "F MATRIX IN COMB", /)	COMB	36
C997	FORMAT(12(1PE10.3))	COMB	37

C998	FORMAT(/,10X,"C IN COMB",/)	COMB	38
C999	FORMAT(12(1PE10,3))	COMB	39
C995	CONTINUE	COMB	40
C	RETURN	COMB	41
	END	COMB	42
	SUBROUTINE DISOLV (M,AREA,ID,ION,T)	COMB	43
C		DISOLV	2
C	THIS ROUTINE CALCULATES COEFFICIENTS OF DECAY AND	DISOLV	3
C	SOURCE TERMS IN THE DISSOLVED CHEMICAL TRANSPORT CONVECTION-	DISOLV	4
C	DIFFUSION EQUATION	DISOLV	5
C		DISOLV	6
	COMMON /BLK1/CT(240),C(9,240)	DISOLV	7
	COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)	DISOLV	8
	COMMON /BLK7/NOD(240,6),X(240),Y(240)	DISOLV	9
	COMMON /BLK9/DX(100),DY(100),EX(100),EY(100),ALFA(100),BETA(100)	DISOLV	10
	9 HS(100)	DISOLV	11
	COMMON /BLK10/PEL(6,6),SEL(6,6),REL(6)	DISOLV	12
	COMMON /BLK11/D50(3,100),SD50(100),SR(3,100),SD(3,100)	DISOLV	13
	COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHOMAT,	DISOLV	14
	1 AKP(3)	DISOLV	15
	COMMON /BLK14/GBA(100,10),GBB(100,10),GBC(100,10),GBD(100,10),	DISOLV	16
	1 GBE(100,10),GBF(100,10),GBG(100,10),PDR	DISOLV	17
	COMMON /BLK15/ILAYR(100,3),XYS0(100),BDIV(100),NBED(100),BED(100),	DISOLV	18
	1 RHOSD(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)	DISOLV	19
	COMMON /BLK17/QLATE(240),QSARA(7,100),QPNT(7,240),CRATE(3,100),	DISOLV	20
	1 CD(4,240)	DISOLV	21
	LEVEL 2,QLATE,QSARA,QPNT,CRATE,CD	FETFIX4	22
	DIMENSION GBJ(3)	DISOLV	23
	DIMENSION GBI(3)	FETFIX8	24
C		DISOLV	25
	DATA EPSI/1.0E-12/	DISOLV	26
	A1=ACOF(1)	DISOLV	27
	A2=ACOF(2)	DISOLV	28
	A3=ACOF(3)	DISOLV	29
	B1=ACOF(4)	DISOLV	30
	B2=ACOF(5)	DISOLV	31
	B3=ACOF(6)	DISOLV	32
C		DISOLV	33
	N1=NOD(M,1)	DISOLV	34
	N2=NOD(M,2)	DISOLV	35
	N3=NOD(M,3)	DISOLV	36
	N4=NOD(M,4)	DISOLV	37
	N5=NOD(M,5)	DISOLV	38
	N6=NOD(M,6)	DISOLV	39
C		DISOLV	40
C		DISOLV	41
	H1 = H(N1)	DISOLV	42
	H2 = H(N2)	DISOLV	43
	H3 = H(N3)	DISOLV	44
C		DISOLV	45
C***	DECAY TERM ***	DISOLV	46
C		DISOLV	47
	ALFA(M) = ALMBDA	DISOLV	48
C**	ALFA(M) = ALMBDA+QLATE(M) IF QLATE(M) IS CONSTANT IN A ELEMENT	DISOLV	49
C		DISOLV	50
C***	SOURCE OR SINK TERM ***	DISOLV	51
C		DISOLV	52
	DO 100 I = 1,6	DISOLV	53
	REL(I) = 0.0	DISOLV	54
100	CONTINUE	DISOLV	55
C		DISOLV	55

C	ADSORPTION/DESORPTION WITH NON-MOVING BED SEDIMENT	DISOLV	56
C		DISOLV	57
	G8I(1) = G8A(M,NBED(M))	FETFIX8	2
	G8I(2) = G8B(M,NBED(M))	FETFIX8	3
	G8I(3) = G8C(M,NBED(M))	FETFIX8	4
	G8J(1) = G8D(M,NBED(M))	DISOLV	58
	G8J(2) = G8E(M,NBED(M))	DISOLV	59
	G8J(3) = G8F(M,NBED(M))	DISOLV	60
	A = 0.0	DISOLV	61
	B = 0.0	DISOLV	62
	DO 105 I = 1,3	DISOLV	63
	AA = RHOSD(I)*(1.0-POR)*BD50(M)*AKJ(I+6,M)	FETFIX6	26
	AA = AA*G8I(I)	FETFIX8	5
	A = -AA*AKJ(I,M)	DISOLV	65
	B = AA*G8J(I)	DISOLV	66
	CRATE(I,M) = 0.0	DISOLV	67
	IF (SR(ID,M).GT.0) GO TO 105	DISOLV	68
	IF (AKJ(I,M).EQ.0.OR.AKJ(I+6,M).EQ.0) GO TO 105	DISOLV	69
	A = A*AREA/180.	DISOLV	70
	B = B*AREA/3.	DISOLV	71
	REL(1) = REL(1) +A*(6.*C(ID,N1)-C(ID,N2)-C(ID,N3)-4.*C(ID,N5))	DISOLV	72
	REL(2) = REL(2) -A*(C(ID,N1)-6.*C(ID,N2)+C(ID,N3)+4.*C(ID,N6))	DISOLV	73
	REL(3) = REL(3) -A*(C(ID,N1)+C(ID,N2)-6.*C(ID,N3)+4.*C(ID,N4))	DISOLV	74
	REL(4) = REL(4) -A*(4.*C(ID,N3)-32.*C(ID,N4)-16.*C(ID,N5)	DISOLV	75
	1 -16.*C(ID,N6))+B	DISOLV	76
	1 REL(5) = REL(5) -A*(4.*C(ID,N1)-16.*C(ID,N4)-32.*C(ID,N5)	DISOLV	77
	1 -16.*C(ID,N6))+B	DISOLV	78
	REL(6) = REL(6) -A*(4.*C(ID,N2)-16.*C(ID,N4)-16.*C(ID,N5)	DISOLV	79
	1 -32.*C(ID,N6))+B	DISOLV	80
	CRATE(I,M) = (REL(1)+REL(2)+REL(3)+REL(4)+REL(5)+REL(6))	DISOLV	81
C		DISOLV	82
C		DISOLV	83
C	IF (M.NE.3) GO TO 105	DISOLV	84
C200	FORMAT(/,2X,"A = ",E12.3,4X,"REL(I) = ",6E12.3,/)	DISOLV	85
C201	FORMAT(/,2X,"CRATE(I) = ",E12.3,4X,"C(ID,N1) = ",6E12.3,/)	DISOLV	86
C	*WRITE(6,200) A,(REL(J),J=1,6)	DISOLV	87
C	*WRITE(6,201) CRATE(I,M),C(ID,N1),C(ID,N2),C(ID,N3),C(ID,N4),	DISOLV	88
C	1 C(ID,N5),C(ID,N6)	DISOLV	89
105	CONTINUE	DISOLV	90
	CRATE(3,M) = CRATE(3,M)-CRATE(2,M)	FETFIX6	27
	CRATE(2,M) = CRATE(2,M)-CRATE(1,M)	DISOLV	91
	CRATE(1,M) = CRATE(1,M)/AREA	DISOLV	93
	CRATE(2,M) = CRATE(2,M)/AREA	DISOLV	94
	CRATE(3,M) = CRATE(3,M)/AREA	DISOLV	95
C		DISOLV	96
C		DISOLV	97
C	DISSOLVED CONTAMINANT SOURCE	DISOLV	98
C		DISOLV	99
	A = AREA*QSARA(ID,M)/60.	DISOLV	100
C		DISOLV	101
	REL(1) = A*(2.*H1-H2-H3) +REL(1)	DISOLV	102
	REL(2) = -A*(H1-2.*H2+H3) +REL(2)	DISOLV	103
	REL(3) = -A*(H1+H2-2.*H3) +REL(3)	DISOLV	104
	REL(4) = A*(8.*H1+8.*H2+4.*H3) +REL(4)	DISOLV	105
	REL(5) = A*(4.*H1+8.*H2+8.*H3) +REL(5)	DISOLV	106
	REL(6) = A*(8.*H1+4.*H2+8.*H3) +REL(6)	DISOLV	107
C		DISOLV	108
C		DISOLV	109
C	ADSORPTION/DESORPTION WITH MOVING SEDIMENT	DISOLV	110
C		DISOLV	111
	DO 110 I = 1,3	DISOLV	112
	A = AKJ(I,M)*AKJ(I+3,M)*AREA/180.	DISOLV	113

	B	= AKJ(I+3,4)*AREA/1260.	DISOLV	114
C			DISOLV	115
	REL(1)	= REL(1)-A*(6.*H1*CD(I,N1)*C(ID,N1)-H2*CD(I,N2)*C(ID,N2)	DISOLV	116
	1	-H3*CD(I,N3)*C(ID,N3)-2.*(H2+H3)*CD(I,N5)*C(ID,N5))	DISOLV	117
	REL(2)	= REL(2)+A*(H1*CD(I,N1)*C(ID,N1)-6.*H2*CD(I,N2)*C(ID,N2)	DISOLV	118
	1	+H3*CD(I,N3)*C(ID,N3)+2.*(H1+H3)*CD(I,N6)*C(ID,N6))	DISOLV	119
	REL(3)	= REL(3)+A*(H1*CD(I,N1)*C(ID,N1)+H2*CD(I,N2)*C(ID,N2)	DISOLV	120
	1	-6.*H3*CD(I,N3)*C(ID,N3)+2.*(H1+H2)*CD(I,N4)*C(ID,N4))	DISOLV	121
	REL(4)	= REL(4)+A*(4.*H3*CD(I,N3)*C(ID,N3)-16.*(H1+H2)*CD(I,N4)*	DISOLV	122
	1	C(ID,N4)-6.*(H2+H3)*CD(I,N5)*C(ID,N5)-8.*(H1+H3)*CD(I,N6)*	DISOLV	123
	2	C(ID,N6))	DISOLV	124
	REL(5)	= REL(5)+A*(4.*H1*CD(I,N1)*C(ID,N1)-8.*(H1+H2)*CD(I,N4)*	DISOLV	125
	1	C(ID,N4)-16.*(H2+H3)*CD(I,N5)*C(ID,N5)-8.*(H1+H3)*CD(I,N6)*	DISOLV	126
	2	C(ID,N6))	DISOLV	127
	REL(6)	= REL(6)+A*(4.*H2*CD(I,N2)*C(ID,N2)-8.*(H1+H2)*CD(I,N4)*	DISOLV	128
	1	C(ID,N4)-8.*(H2+H3)*CD(I,N5)*C(ID,N5)-16.*(H1+H3)*CD(I,N6)*	DISOLV	129
	2	C(ID,N6))	DISOLV	130
C			DISOLV	131
	REL(1)	= REL(1) +8*((30.*H1+6.*H2+6.*H3)*C(ID+I,N1)	DISOLV	132
	1	-(4.*H1+4.*H2-H3)*C(ID+I,N2) -(4.*H1-H2+4.*H3)*C(ID+I,N3)	DISOLV	133
	2	+(12.*H1-8.*H2-4.*H3)*C(ID+I,N4)-(4.*H1+12.*H2+12.*H3)*C(ID+I,N5)	DISOLV	134
	3	+(12.*H1-4.*H2-8.*H3)*C(ID+I,N6))	DISOLV	135
	REL(2)	= REL(2) +8*((4.*H1+4.*H2-H3)*C(ID+I,N1)*(-1)	DISOLV	136
	1	+(6.*H1+30.*H2+6.*H3)*C(ID+I,N2)+(H1-4.*H2-4.*H3)*C(ID+I,N3)	DISOLV	137
	2	-(8.*H1-12.*H2+4.*H3)*C(ID+I,N4)-(4.*H1-12.*H2-8.*H3)*C(ID+I,N5)	DISOLV	138
	3	-(12.*H1+4.*H2+12.*H3)*C(ID+I,N6))	DISOLV	139
	REL(3)	= REL(3) +8*((4.*H1-H2+4.*H3)*C(ID+I,N1)*(-1)	DISOLV	140
	1	+(H1-4.*H2-4.*H3)*C(ID+I,N2) +(6.*H1+6.*H2+30.*H3)*C(ID+I,N3)	DISOLV	141
	2	-(12.*H1+12.*H2+4.*H3)*C(ID+I,N4)-(4.*H1+8.*H2-12.*H3)*C(ID+I,N5)	DISOLV	142
	3	-(8.*H1+4.*H2-12.*H3)*C(ID+I,N6))	DISOLV	143
	REL(4)	= REL(4) +8*((12.*H1-8.*H2-4.*H3)*C(ID+I,N1)	DISOLV	144
	1	-(8.*H1-12.*H2+4.*H3)*C(ID+I,N2)-(12.*H1+12.*H2+4.*H3)*C(ID+I,N3)	DISOLV	145
	2	+(96.*H1+96.*H2+32.*H3)*C(ID+I,N4)+(32.*H1+48.*H2+32.*H3)*	DISOLV	146
	3	C(ID+I,N5)+(48.*H1+32.*H2+32.*H3)*C(ID+I,N6))	DISOLV	147
	REL(5)	= REL(5) +8*((4.*H1+12.*H2+12.*H3)*C(ID+I,N1)*(-1)	DISOLV	148
	1	-(4.*H1-12.*H2+8.*H3)*C(ID+I,N2)-(4.*H1+8.*H2-12.*H3)*C(ID+I,N3)	DISOLV	149
	2	+(32.*H1+48.*H2+32.*H3)*C(ID+I,N4)+(32.*H1+96.*H2+96.*H3)*	DISOLV	150
	3	C(ID+I,N5)+(32.*H1+32.*H2+48.*H3)*C(ID+I,N6))	DISOLV	151
	REL(6)	= REL(6) +8*((12.*H1-4.*H2-8.*H3)*C(ID+I,N1)	DISOLV	152
	1	-(12.*H1+4.*H2+12.*H3)*C(ID+I,N2)-(8.*H1+4.*H2-12.*H3)*C(ID+I,N3)	DISOLV	153
	2	+(48.*H1+32.*H2+32.*H3)*C(ID+I,N4)+(32.*H1+32.*H2+48.*H3)*	DISOLV	154
	3	C(ID+I,N5)+(96.*H1+32.*H2+96.*H3)*C(ID+I,N6))	DISOLV	155
110		CONTINUE	DISOLV	156
C			DISOLV	157
C		POINT SOURCE QPNT(ID,NDS)	DISOLV	158
C			DISOLV	159
	B	= 1.0	DISOLV	160
C			DISOLV	161
	REL(1)	= REL(1) +8*((30.*H1+6.*H2+6.*H3)*QPNT(ID,N1)	DISOLV	162
	1	-(4.*H1+4.*H2-H3)*QPNT(ID,N2) -(4.*H1-H2+4.*H3)*QPNT(ID,N3)	DISOLV	163
	2	+(12.*H1-8.*H2-4.*H3)*QPNT(ID,N4)-(4.*H1+12.*H2+12.*H3)*	DISOLV	164
	3	QPNT(ID,N5)+(12.*H1-4.*H2-8.*H3)*QPNT(ID,N6))	DISOLV	165
	REL(2)	= REL(2) +8*((4.*H1+4.*H2-H3)*QPNT(ID,N1)*(-1)	DISOLV	166
	1	+(6.*H1+30.*H2+6.*H3)*QPNT(ID,N2)+(H1-4.*H2-4.*H3)*QPNT(ID,N3)	DISOLV	167
	2	-(8.*H1-12.*H2+4.*H3)*QPNT(ID,N4)-(4.*H1-12.*H2-8.*H3)*QPNT(ID,N5)	DISOLV	168
	3	-(12.*H1+4.*H2+12.*H3)*QPNT(ID,N6))	DISOLV	169
	REL(3)	= REL(3) +8*((4.*H1-H2+4.*H3)*QPNT(ID,N1)*(-1)	DISOLV	170
	1	+(H1-4.*H2-4.*H3)*QPNT(ID,N2) +(6.*H1+6.*H2+30.*H3)*QPNT(ID,N3)	DISOLV	171
	2	-(12.*H1+12.*H2+4.*H3)*QPNT(ID,N4)-(4.*H1+8.*H2-12.*H3)*	DISOLV	172
	3	QPNT(ID,N5)-(8.*H1+4.*H2-12.*H3)*QPNT(ID,N6))	DISOLV	173
	REL(4)	= REL(4) +8*((12.*H1-8.*H2-4.*H3)*QPNT(ID,N1)	DISOLV	174
	1	-(8.*H1-12.*H2+4.*H3)*QPNT(ID,N2)-(12.*H1+12.*H2+4.*H3)*	DISOLV	175

	2QPNT(ID,N3)+(96.*H1+96.*H2+32.*H3)*QPNT(ID,N4)+(32.*H1+48.*H2+32.*	DISOLV	176
	3H3)*QPNT(ID,N5)+(48.*H1+32.*H2+32.*H3)*QPNT(ID,N6))	DISOLV	177
	REL(5) = REL(5) +8*((4.*H1+12.*H2+12.*H3)*QPNT(ID,N1))*(-1)	DISOLV	178
	1=(4.*H1-12.*H2+8.*H3)*QPNT(ID,N2)-(4.*H1+8.*H2-12.*H3)*QPNT(ID,N3)	DISOLV	179
	2+(32.*H1+48.*H2+32.*H3)*QPNT(ID,N4)+(32.*H1+96.*H2+96.*H3)*	DISOLV	180
	3QPNT(ID,N5)+(32.*H1+32.*H2+48.*H3)*QPNT(ID,N6))	DISOLV	181
	REL(6) = REL(6) +8*((12.*H1-4.*H2-8.*H3)*QPNT(ID,N1)	DISOLV	182
	1=(12.*H1+4.*H2+12.*H3)*QPNT(ID,N2)-(8.*H1+4.*H2-12.*H3)*	DISOLV	183
	2QPNT(ID,N3)+(48.*H1+32.*H2+32.*H3)*QPNT(ID,N4)+(32.*H1+32.*H2+48.	DISOLV	184
	3*H3)*QPNT(ID,N5)+(96.*H1+32.*H2+96.*H3)*QPNT(ID,N6))	DISOLV	185
C		DISOLV	186
	RETURN	DISOLV	187
	END	DISOLV	188
	SUBROUTINE DUBOY(ID,M,II,QS)	DUBOY	2
C		DUBOY	3
C**	THIS SUBROUTINE COMPUTES THE SEDIMENT LOAD	DUBOY	4
C**	CAPACITY BY DUBOY'S FORMULA	DUBOY	5
C		DUBOY	6
	COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)	DUBOY	7
	COMMON /BLK11/D50(3,100),BD50(100),SR(3,100),SD(3,100)	DUBOY	8
C		DUBOY	9
C		DUBOY	10
C	DUBOY'S FORMULA MUST USE THE FOLLOWING UNITS	DUBOY	11
C	D50(J,M)..MEDIAN SAND DIAMETER IN MILLIMETERS J=1,2,3 M=1,NE	DUBOY	12
C	QS.....TOTAL SAND LOAD PER UNIT WIDTH IN LB/SEC-FT	DUBOY	13
C	STRESS(J)..BED SHEAR STRESS IN LB/SQ.FT	DUBOY	14
C	TAUC.....CRITICAL SHEAR STRESS AT WHICH SEDIMENT	DUBOY	15
C	MOVEMENT BEGINS, LBF/FT**2.	DUBOY	16
C		DUBOY	17
	J=ID	DUBOY	18
	D50TMP =D50(J,M)*1000.0	DUBOY	19
	TAU=0.2627*ALOG10(D50TMP)*ALOG10(D50TMP)+0.590*ALOG10(D50TMP	DUBOY	20
	1)-1.4962	DUBOY	21
	TAUC=10.0**(TAU)	DUBOY	22
	PSI=ALOG10(28.8)-0.7365*ALOG10(D50TMP)	DUBOY	23
	PSID=10.0**(PSI)	DUBOY	24
	QS=0.	DUBOY	25
	STRESS(II)=STRESS(II)/4.88243	DUBOY	26
	IF (STRESS(II).LE.TAUC) GO TO 999	DUBOY	27
	QS=PSID*STRESS(II)*(STRESS(II)-TAUC)	DUBOY	28
	QS=QS*128577.0	DUBOY	29
999	RETURN	DUBOY	30
	END	DUBOY	31
	SUBROUTINE ERFC(X,ANS)	ERFC	2
C		ERFC	3
C**	THIS SUBROUTINE CALCULATES THE ERROR FUNCTION OF ABS(X) TO WITHIN	ERFC	4
C**	3*10**-7. IT THEN CALCULATES THE COMPLIMENTARY ERROR FUNCTION	ERFC	5
C**	OF X AND RETURNS THAT VALUE AS ANS. ANS IS VALID FOR ALL REAL X.	ERFC	6
C		ERFC	7
	DIMENSION A(6)	ERFC	8
	DATA A(1),A(2),A(3),A(4),A(5),A(6)/.0705230784,.0422820123,	ERFC	9
	.0092705272,.0001520143,.0002765672,.0000430638/	ERFC	10
C		ERFC	11
C**	NOW CALCULATE ERF(X)	ERFC	12
C		ERFC	13
	XX=1.	ERFC	14
	ACCUM=1.	ERFC	15
	DO 10 I=1,6	ERFC	16
	XX=ABS(X)*XX	ERFC	17
	ACCUM=ACCUM+XX*A(I)	ERFC	18
10	CONTINUE	ERFC	19
C		ERFC	20

C**	THIS CHECK WORKS ON THE CDC 7000 SERIES.	ERFC	21
C		ERFC	22
C	IF(ACCUM .GT. 10.**20) GO TO 20	ERFC	23
C		ERFC	24
C**	THIS CHECK WORKS ON THE VAX 11/780.	ERFC	25
C		ERFC	26
	IF(ACCUM .GT. 200.0) GO TO 20	ERFC	27
	ERF=1.-1./ACCUM**16	ERFC	28
	GO TO 30	ERFC	29
20	ERF=1.	ERFC	30
C		ERFC	31
C**	NOW CALCULATE ANS = ERFC(X)	ERFC	32
C		ERFC	33
30	ANS=1.-ERF	ERFC	34
	IF (X .GE. 0.) RETURN	ERFC	35
	ANS=1.+ERF	ERFC	36
	RETURN	ERFC	37
	END	ERFC	38
	SUBROUTINE MATADD (M,NCOL,IO)	MATADD	2
C		MATADD	3
C	THIS SUBROUTINE ASSEMBLES THE GLOBAL SYSTEM MATRICES (P),(S) AND	MATADD	4
C	THE LOAD VECTOR R BY ADDING IN EACH ELEMENTAL MATRIX. THE	MATADD	5
C	SYSTEM MATRICES ARE STORED IN BAND FORM	MATADD	6
C		MATADD	7
	COMMON /BLK2/P(240,86),F(240)	MATADD	8
	LEVEL 2,P,F	FETFIX4	17
	COMMON /BLK3/S(240,86)	MATADD	9
	LEVEL 2,S	FETFIX4	18
	COMMON /BLK4/R(240),RPAST(240,7),NODBET,BETA1,AREA1	MATADD	10
	LEVEL 2,R,RPAST,NODBET,BETA1,AREA1	FETFIX4	19
	COMMON /BLK7/NOD(240,6),X(240),Y(240)	MATADD	11
	COMMON /BLK10/PEL(6,6),SEL(6,6),REL(6)	MATADD	12
	COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHOMAT,	MATADD	13
1	AKP(3)	MATADD	14
C		MATADD	15
C		MATADD	16
	NBND=2*NCOL-1	MATADD	17
	DO 110 J=1,6	MATADD	18
	NR=NOD(M,J)	MATADD	19
	DO 100 K=1,6	MATADD	20
	NC=NOD(M,K)-NR+1	MATADD	21
	MC=NCOL+NC-1	MATADD	22
	P(NR,MC)=P(NR,MC)+PEL(J,K)	MATADD	23
	S(NR,MC)=S(NR,MC)+SEL(J,K)	MATADD	24
100	CONTINUE	MATADD	25
	R(NR)=R(NR)+REL(J)	MATADD	26
110	CONTINUE	MATADD	27
	IF (ID.EQ.4.AND.M.EQ.1) R(NODBET) = R(NODBET)+BETA1	MATADD	28
C	IF (ID.EQ.5.AND.M.EQ.182) R(NODBET)=R(NODBET)+AKJ(1,M)*BETA1	MATADD	29
C	IF (ID.EQ.6.AND.M.EQ.182) R(NODBET)=R(NODBET)+AKJ(2,M)*BETA1	MATADD	30
C	IF (ID.EQ.7.AND.M.EQ.182) R(NODBET)=R(NODBET)+AKJ(3,M)*BETA1	MATADD	31
C		MATADD	32
	RETURN	MATADD	33
	END	MATADD	34
	SUBROUTINE PARTIC (M,AREA,IO,IDN,T)	PARTIC	2
C		PARTIC	3
C	THIS SUBROUTINE CALCULATES COEFFICIENTS OF	PARTIC	4
C	DECAY AND SOURCE TERMS IN THE PARTICULATE NUCLIDE TRANSPORT EQ.	PARTIC	5
C		PARTIC	6
	COMMON /BLK1/CT(240),C(9,240)	PARTIC	7
	COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)	PARTIC	8
	COMMON /BLK7/NOD(240,6),X(240),Y(240)	PARTIC	9

	COMMON /BLK9/DX(100),DY(100),EX(100),EY(100),ALFA(100),BETA(100)	PARTIC	10
9	,HS(100)	PARTIC	11
	COMMON /BLK10/PEL(6,6),SEL(6,6),REL(6)	PARTIC	12
	COMMON /BLK11/D50(3,100),BD50(100),SR(3,100),SD(3,100)	PARTIC	13
	COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHOWAT,	PARTIC	14
1	AKP(3)	PARTIC	15
	COMMON /BLK14/G8A(100,10),G8B(100,10),G8C(100,10),G8D(100,10),	PARTIC	16
1	G8E(100,10),G8F(100,10),G8G(100,10),POR	PARTIC	17
	COMMON /BLK15/ILAYR(100,3),XYSO(100),BDIV(100),NBED(100),BED(100),	PARTIC	18
1	RHOSED(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)	PARTIC	19
	COMMON /BLK17/QLATE(240),QSAWA(7,100),QPNT(7,240),CRATE(3,100),	PARTIC	20
1	CD(4,240)	PARTIC	21
	LEVEL 2,QLATE,QSARA,QPNT,CRATE,CD	FETFIX4	20
	DIMENSION CS(3)	PARTIC	22
	DATA EPSI/1.0E-12/	PARTIC	23
C		PARTIC	24
	A1=ACOF(1)	PARTIC	25
	A2=ACOF(2)	PARTIC	26
	A3=ACOF(3)	PARTIC	27
	B1=ACOF(4)	PARTIC	28
	B2=ACOF(5)	PARTIC	29
	B3=ACOF(6)	PARTIC	30
C		PARTIC	31
	N1=NOD(M,1)	PARTIC	32
	N2=NOD(M,2)	PARTIC	33
	N3=NOD(M,3)	PARTIC	34
	N4=NOD(M,4)	PARTIC	35
	N5=NOD(M,5)	PARTIC	36
	N6=NOD(M,6)	PARTIC	37
	H1 = H(N1)	PARTIC	38
	H2 = H(N2)	PARTIC	39
	H3 = H(N3)	PARTIC	40
C		PARTIC	41
C		PARTIC	42
C***	DECAY TERM ***	PARTIC	43
C		PARTIC	44
	ALFA(M) = ALMBDA	PARTIC	45
C**	ALFA(M) = ALMBDA+QLATE(M) IF QLATE(M) IS CONSTANT IN A ELEMENT	PARTIC	46
C	QLATE(M) IS A LATERAL INFLOW OF WATER PER UNIT VOLUME (1/SEC)	PARTIC	47
C		PARTIC	48
C		PARTIC	49
C***	SOURCE OR SINK TERM ***	PARTIC	50
C		PARTIC	51
	DO 100 I = 1,6	PARTIC	52
	REL(I) = 0.0	PARTIC	53
100	CONTINUE	PARTIC	54
C		PARTIC	55
C	DUE TO SEDIMENT EROSION AND/OR DEPOSITION	PARTIC	56
C		PARTIC	57
	CS(1) = RSAV1(M) * AREA/3.0	PARTIC	58
	CS(2) = RSAV2(M) * AREA/3.0	PARTIC	59
	CS(3) = RSAV3(M) * AREA/3.0	PARTIC	60
	REL(1) = 0.0	PARTIC	61
	REL(2) = 0.0	PARTIC	62
	REL(3) = 0.0	PARTIC	63
	REL(4) = CS(ID=4)	PARTIC	64
	REL(5) = REL(4)	PARTIC	65
	REL(6) = REL(4)	PARTIC	66
C		PARTIC	67
C	IF (M.EQ.77) WRITE(6,999) ID,(REL(I),I=1,6)	PARTIC	68
C999	FORMAT(2X,"ID,REL=",I5,6E12.5)	PARTIC	69
C	SORBED CONTAMINANT GENERATION	PARTIC	70

C	A	= AREA*QSARA(ID,M)/60.	PARTIC	71
	REL(1)	= REL(1)+A*(2.*H1-H2-H3)	PARTIC	72
	REL(2)	= REL(2)-A*(H1-2.*H2+H3)	PARTIC	73
	REL(3)	= REL(3)+A*(H1+H2-2.*H3)	PARTIC	74
	REL(4)	= REL(4)+A*(8.*H1+8.*H2+4.*H3)	PARTIC	75
	REL(5)	= REL(5)+A*(4.*H1+8.*H2+8.*H3)	PARTIC	76
	REL(6)	= REL(6)+A*(8.*H1+4.*H2+8.*H3)	PARTIC	77
C		IF(M.EQ.77) WRITE(6,999) ID,(REL(I),I=1,6)	PARTIC	78
C			PARTIC	79
	I=ID-4		PARTIC	80
	A	= AKJ(I,M)*AKJ(I+3,M)*AREA/180.	PARTIC	81
	B	= AKJ(I+3,M)*AREA/1260.	PARTIC	82
C			PARTIC	83
	REL(1)	= REL(1)+A*(6.*H1*CD(I,N1)*CD(4,N1)-H2*CD(I,N2)*CD(4,N2)	PARTIC	84
	1	+H3*CD(I,N3)*CD(4,N3)-2.*(H2+H3)*CD(I,N5)*CD(4,N5))	PARTIC	85
	REL(2)	= REL(2)-A*(H1*CD(I,N1)*CD(4,N1)-6.*H2*CD(I,N2)*CD(4,N2)	PARTIC	86
	1	+H3*CD(I,N3)*CD(4,N3)+2.*(H1+H3)*CD(I,N6)*CD(4,N6))	PARTIC	87
	REL(3)	= REL(3)-A*(H1*CD(I,N1)*CD(4,N1)+H2*CD(I,N2)*CD(4,N2)	PARTIC	88
	1	-6.*H3*CD(I,N3)*CD(4,N3)+2.*(H1+H2)*CD(I,N4)*CD(4,N4))	PARTIC	89
	REL(4)	= REL(4)-A*(4.*H3*CD(I,N3)*CD(4,N3)-16.*(H1+H2)*CD(I,N4)*	PARTIC	90
	1	CD(4,N4)-8.*(H2+H3)*CD(I,N5)*CD(4,N5)-8.*(H1+H3)*CD(I,N6)*	PARTIC	91
	2	CD(4,N6))	PARTIC	92
	REL(5)	= REL(5)-A*(4.*H1*CD(I,N1)*CD(4,N1)-8.*(H1+H2)*CD(I,N4)*	PARTIC	93
	1	CD(4,N4)-16.*(H2+H3)*CD(I,N5)*CD(4,N5)-8.*(H1+H3)*CD(I,N6)*	PARTIC	94
	2	CD(4,N6))	PARTIC	95
	REL(6)	= REL(6)-A*(4.*H2*CD(I,N2)*CD(4,N2)-8.*(H1+H2)*CD(I,N4)*	PARTIC	96
	1	CD(4,N4)-8.*(H2+H3)*CD(I,N5)*CD(4,N5)-16.*(H1+H3)*CD(I,N6)*	PARTIC	97
	2	CD(4,N6))	PARTIC	98
C			PARTIC	99
C		IF(M.EQ.77) WRITE(6,999) ID,(REL(J),J=1,6)	PARTIC	100
C		IF(M.EQ.77) WRITE(6,998) N2,I,H1,H2,H3,CD(I,N2),CD(4,N2),C(ID,N2)	PARTIC	101
C998		FORMAT(2X,"N2,I,H,CD(I,4, ID,N2)",2I5,6E12.5)	PARTIC	102
	REL(1)	= REL(1) -B*((30.*H1+6.*H2+6.*H3)*C(ID,N1)	PARTIC	103
	1	-(4.*H1+4.*H2-H3)*C(ID,N2) - (4.*H1-H2+4.*H3)*C(ID,N3)	PARTIC	104
	2	+(12.*H1-8.*H2-4.*H3)*C(ID,N4)-(4.*H1+12.*H2+12.*H3)*C(ID,N5)	PARTIC	105
	3	+(12.*H1-4.*H2-8.*H3)*C(ID,N6))	PARTIC	106
	REL(2)	= REL(2) -B*((4.*H1+4.*H2-H3)*C(ID,N1)*(-1)	PARTIC	107
	1	+(6.*H1+30.*H2+6.*H3)*C(ID,N2)+(H1-4.*H2-4.*H3)*C(ID,N3)	PARTIC	108
	2	-(8.*H1-12.*H2+4.*H3)*C(ID,N4)-(4.*H1-12.*H2+8.*H3)*C(ID,N5)	PARTIC	109
	3	-(12.*H1+4.*H2+12.*H3)*C(ID,N6))	PARTIC	110
	REL(3)	= REL(3) -B*((4.*H1-H2+4.*H3)*C(ID,N1)*(-1)	PARTIC	111
	1	+(H1-4.*H2-4.*H3)*C(ID,N2) +(6.*H1+6.*H2+30.*H3)*C(ID,N3)	PARTIC	112
	2	-(12.*H1+12.*H2+4.*H3)*C(ID,N4)-(4.*H1+8.*H2-12.*H3)*C(ID,N5)	PARTIC	113
	3	-(8.*H1+4.*H2-12.*H3)*C(ID,N6))	PARTIC	114
	REL(4)	= REL(4) -B*((12.*H1-8.*H2-4.*H3)*C(ID,N1)	PARTIC	115
	1	-(8.*H1-12.*H2+4.*H3)*C(ID,N2)-(12.*H1+12.*H2+4.*H3)*C(ID,N3)	PARTIC	116
	2	+(96.*H1+96.*H2+32.*H3)*C(ID,N4)+(32.*H1+48.*H2+32.*H3)*	PARTIC	117
	3	C(ID,N5)+(48.*H1+32.*H2+32.*H3)*C(ID,N6))	PARTIC	118
	REL(5)	= REL(5) -B*((4.*H1+12.*H2+12.*H3)*C(ID,N1)*(-1)	PARTIC	119
	1	-(4.*H1-12.*H2+8.*H3)*C(ID,N2)-(4.*H1+8.*H2-12.*H3)*C(ID,N3)	PARTIC	120
	2	+(32.*H1+48.*H2+32.*H3)*C(ID,N4)+(32.*H1+96.*H2+96.*H3)*	PARTIC	121
	3	C(ID,N5)+(32.*H1+32.*H2+48.*H3)*C(ID,N6))	PARTIC	122
	REL(6)	= REL(6) -B*((12.*H1-4.*H2-8.*H3)*C(ID,N1)	PARTIC	123
	1	-(12.*H1+4.*H2+12.*H3)*C(ID,N2)-(8.*H1+4.*H2-12.*H3)*C(ID,N3)	PARTIC	124
	2	+(48.*H1+32.*H2+32.*H3)*C(ID,N4)+(32.*H1+32.*H2+48.*H3)*	PARTIC	125
	3	C(ID,N5)+(96.*H1+32.*H2+96.*H3)*C(ID,N6))	PARTIC	126
C			PARTIC	127
C		IF(M.EQ.77) WRITE(6,999) ID,(REL(I),I=1,6)	PARTIC	128
C		POINT SOURCE CONTRIBUTION	PARTIC	129
C			PARTIC	130
	B	= 1.0	PARTIC	131
			PARTIC	132

C	REL(1) = REL(1)	+8*((30.*H1+6.*H2+6.*H3)*QPNT(ID,N1)	PARTIC	133
	1-(4.*H1+4.*H2+H3)*QPNT(ID,N2)	-(4.*H1+H2+4.*H3)*QPNT(ID,N3)	PARTIC	134
	2+(12.*H1+8.*H2+4.*H3)*QPNT(ID,N4)	-(4.*H1+12.*H2+12.*H3)*	PARTIC	135
	3QPNT(ID,N5)+(12.*H1+4.*H2+8.*H3)*QPNT(ID,N6))		PARTIC	136
	REL(2) = REL(2)	+8*((4.*H1+4.*H2+H3)*QPNT(ID,N1)*(-1)	PARTIC	137
	1+(6.*H1+30.*H2+6.*H3)*QPNT(ID,N2)	+(H1+4.*H2+4.*H3)*QPNT(ID,N3)	PARTIC	138
	2-(8.*H1+12.*H2+4.*H3)*QPNT(ID,N4)	-(4.*H1+12.*H2+8.*H3)*QPNT(ID,N5)	PARTIC	139
	3-(12.*H1+4.*H2+12.*H3)*QPNT(ID,N6))		PARTIC	140
	REL(3) = REL(3)	+8*((4.*H1+H2+4.*H3)*QPNT(ID,N1)*(-1)	PARTIC	141
	1+(H1+4.*H2+4.*H3)*QPNT(ID,N2)	+(6.*H1+6.*H2+30.*H3)*QPNT(ID,N3)	PARTIC	142
	2-(12.*H1+12.*H2+4.*H3)*QPNT(ID,N4)	-(4.*H1+8.*H2+12.*H3)*	PARTIC	143
	3QPNT(ID,N5)+(8.*H1+4.*H2+12.*H3)*QPNT(ID,N6))		PARTIC	144
	REL(4) = REL(4)	+8*((12.*H1+8.*H2+4.*H3)*QPNT(ID,N1)	PARTIC	145
	1-(8.*H1+12.*H2+4.*H3)*QPNT(ID,N2)	-(12.*H1+12.*H2+4.*H3)*	PARTIC	146
	2QPNT(ID,N3)+(96.*H1+96.*H2+32.*H3)*QPNT(ID,N4)	+(32.*H1+48.*H2+32.*	PARTIC	147
	3H3)*QPNT(ID,N5)+(48.*H1+32.*H2+32.*H3)*QPNT(ID,N6))		PARTIC	148
	REL(5) = REL(5)	+8*((4.*H1+12.*H2+12.*H3)*QPNT(ID,N1)*(-1)	PARTIC	149
	1-(4.*H1+12.*H2+8.*H3)*QPNT(ID,N2)	-(4.*H1+8.*H2+12.*H3)*QPNT(ID,N3)	PARTIC	150
	2+(32.*H1+48.*H2+32.*H3)*QPNT(ID,N4)	+(32.*H1+96.*H2+96.*H3)*	PARTIC	151
	3QPNT(ID,N5)+(32.*H1+32.*H2+48.*H3)*QPNT(ID,N6))		PARTIC	152
	REL(6) = REL(6)	+8*((12.*H1+4.*H2+8.*H3)*QPNT(ID,N1)	PARTIC	153
	1-(12.*H1+4.*H2+12.*H3)*QPNT(ID,N2)	-(8.*H1+4.*H2+12.*H3)*	PARTIC	154
	2QPNT(ID,N3)+(48.*H1+32.*H2+32.*H3)*QPNT(ID,N4)	+(32.*H1+32.*H2+48.*	PARTIC	155
	3H3)*QPNT(ID,N5)+(96.*H1+32.*H2+96.*H3)*QPNT(ID,N6))		PARTIC	156
			PARTIC	157
			PARTIC	158
C	IF(M.EQ.77) WRITE(6,999) ID,(REL(I),I=1,6)		PARTIC	159
C	RETURN		PARTIC	160
	END		PARTIC	161
	SUBROUTINE PMATRX (M,AREA)		PMATRX	2
C			PMATRX	3
C	THIS SUBROUTINE CONSTRUCTS THE SYMMETRIC ELEMENTAL MATRIX "PEL"		PMATRX	4
C	FOR THE M-TH TRIANGULAR ELEMENT.		PMATRX	5
C			PMATRX	6
	COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)		PMATRX	7
	COMMON /BLK7/NOD(240,6),X(240),Y(240)		PMATRX	8
	COMMON /BLK10/PEL(6,6),SEL(6,6),REL(6)		PMATRX	9
C			PMATRX	10
	N1 = NOD(M,1)		PMATRX	11
	N2 = NOD(M,2)		PMATRX	12
	N3 = NOD(M,3)		PMATRX	13
	H1 = H(N1)		PMATRX	14
	H2 = H(N2)		PMATRX	15
	H3 = H(N3)		PMATRX	16
	A = AREA/1260.		PMATRX	17
C			PMATRX	18
	PEL(1,1) = A*(30.*H1+6.*H2+6.*H3)		PMATRX	19
	PEL(1,2) = -A*(4.*H1+4.*H2+H3)		PMATRX	20
	PEL(1,3) = -A*(4.*H1+H2+4.*H3)		PMATRX	21
	PEL(1,4) = A*(12.*H1+8.*H2+4.*H3)		PMATRX	22
	PEL(1,5) = -A*(4.*H1+12.*H2+12.*H3)		PMATRX	23
	PEL(1,6) = A*(12.*H1+4.*H2+8.*H3)		PMATRX	24
	PEL(2,2) = A*(6.*H1+30.*H2+6.*H3)		PMATRX	25
	PEL(2,3) = A*(H1+4.*H2+4.*H3)		PMATRX	26
	PEL(2,4) = -A*(8.*H1+12.*H2+4.*H3)		PMATRX	27
	PEL(2,5) = -A*(4.*H1+12.*H2+8.*H3)		PMATRX	28
	PEL(2,6) = -A*(12.*H1+4.*H2+12.*H3)		PMATRX	29
	PEL(3,3) = A*(6.*H1+6.*H2+30.*H3)		PMATRX	30
	PEL(3,4) = -A*(12.*H1+12.*H2+4.*H3)		PMATRX	31
	PEL(3,5) = -A*(4.*H1+8.*H2+12.*H3)		PMATRX	32
	PEL(3,6) = -A*(8.*H1+4.*H2+12.*H3)		PMATRX	33
	PEL(4,4) = A*(96.*H1+96.*H2+32.*H3)		PMATRX	34

	PEL(4,5) = A*(32.*H1+48.*H2+32.*H3)	PMATRX	35
	PEL(4,6) = A*(48.*H1+32.*H2+32.*H3)	PMATRX	36
	PEL(5,5) = A*(32.*H1+96.*H2+96.*H3)	PMATRX	37
	PEL(5,6) = A*(32.*H1+32.*H2+48.*H3)	PMATRX	38
	PEL(6,6) = A*(96.*H1+32.*H2+96.*H3)	PMATRX	39
C	DO 100 I=1,5	PMATRX	40
	II=I+1	PMATRX	41
	DO 100 J=II,6	PMATRX	42
	PEL(J,I)=PEL(I,J)	PMATRX	43
100	CONTINUE	PMATRX	44
C		PMATRX	45
	RETURN	PMATRX	46
	END	PMATRX	47
	SUBROUTINE RMATRX (M,AREA,IO,INO)	PMATRX	48
C		RMATRX	2
C		RMATRX	3
C	THIS SUBROUTINE CONSTRUCTS THE ELEMENTAL LOAD VECTOR (REL) FOR	RMATRX	4
C	THE M-TH TRIANGULAR ELEMENT.	RMATRX	5
	COMMON /BLK6/ VX(240),VY(240),H(240),STRESS(3)	RMATRX	6
	COMMON /BLK7/NOD(240,6),X(240),Y(240)	RMATRX	7
	COMMON /BLK8/KBC,LBC,MBC(7,120),NBC(7,120),OBC(7,120),BC(7,120),	RMATRX	8
	1 KODE(10)	RMATRX	9
	COMMON /BLK9/DX(100),DY(100),EX(100),EY(100),ALFA(100),BETA(100)	RMATRX	10
	9 ,HS(100)	RMATRX	11
	COMMON /BLK10/PEL(6,6),SEL(6,6),REL(6)	RMATRX	12
	DIMENSION Z(6)	RMATRX	13
C		RMATRX	14
C	SOURCE TERM CONTRIBUTION	RMATRX	15
C		RMATRX	16
	N1 = NOD(M,1)	RMATRX	17
	N2 = NOD(M,2)	RMATRX	18
	N3 = NOD(M,3)	RMATRX	19
	H1 = H(N1)	RMATRX	20
	H2 = H(N2)	RMATRX	21
	H3 = H(N3)	RMATRX	22
C		RMATRX	23
	IF (INC.GT.0) GO TO 105	RMATRX	24
	DO 100 I=1,6	RMATRX	25
	REL(I)=0.0	RMATRX	26
	Z(I)=BETA(M)*AREA/3.0	RMATRX	27
	IF (I.LE.3) Z(I)=0.0	RMATRX	28
	Z(I) = Z(I) * (H1+H2+H3)/3.0	RMATRX	29
100	CONTINUE	RMATRX	30
C		RMATRX	31
C	INSERT THE DERIVATIVE BOUNDARY CONDITIONS	RMATRX	32
C		RMATRX	33
C	CONTINUE	RMATRX	34
105	IF (KBC.LE.0) GO TO 120	RMATRX	35
	DO 110 K=4,6	RMATRX	36
	NSIDE=NOD(M,K)	RMATRX	37
	DO 110 I=1,KBC	RMATRX	38
	IF (NSIDE.NE.MBC(IO,I)) GO TO 110	RMATRX	39
	FBC1=0.	RMATRX	40
	FBC2=0.	RMATRX	41
	FBC3=0.	RMATRX	42
	IF (K.EQ.5) FBC1=OBC(IO,I)	RMATRX	43
	IF (K.EQ.4) FBC3=OBC(IO,I)	RMATRX	44
	N1=NOD(M,1)	RMATRX	45
	N2=NOD(M,2)	RMATRX	46
	IF (K.EQ.6) FBC2=OBC(IO,I)	RMATRX	47
	N3=NOD(M,3)	RMATRX	48
		RMATRX	49

	S1=SQRT((X(N3)-X(N2))**2+(Y(N3)-Y(N2))**2)	RMATRX	50
	S2=SQRT((X(N3)-X(N1))**2+(Y(N3)-Y(N1))**2)	RMATRX	51
	S3=SQRT((X(N2)-X(N1))**2+(Y(N2)-Y(N1))**2)	RMATRX	52
C		RMATRX	53
	REL(1)=(FBC2*S2+FBC3*S3)/6.*(H1+H2+H3)/3.+REL(1)	RMATRX	54
	REL(2)=(FBC3*S3+FBC1*S1)/6.*(H1+H2+H3)/3.+REL(2)	RMATRX	55
	REL(3)=(FBC1*S1+FBC2*S2)/6.*(H1+H2+H3)/3.+REL(3)	RMATRX	56
	REL(4)=2.*FBC3*S3/3.*(H1+H2+H3)/3.+REL(4)	RMATRX	57
	REL(5)=2.*FBC1*S1/3.*(H1+H2+H3)/3.+REL(5)	RMATRX	58
	REL(6)=2.*FBC2*S2/3.*(H1+H2+H3)/3.+REL(6)	RMATRX	59
110	CONTINUE	RMATRX	60
C		RMATRX	61
120	CONTINUE	RMATRX	62
	DO 130 I=1,6	RMATRX	63
	REL(I)=REL(I)+Z(I)	RMATRX	64
130	CONTINUE	RMATRX	65
C		RMATRX	66
C		RMATRX	67
	RETURN	RMATRX	68
	END	RMATRX	69
	SUBROUTINE RVEL (AT,RAT,NCNDS,VFREQ)	RVEL	2
C		RVEL	3
C**	THIS SUBROUTINE READS DISCHARGE AND DEPTH DATA FROM THE DIRECT	RVEL	4
C**	ACCESS FILES GENERATED BY CAPE. LOGICAL UNITS LUQ AND LUH MUST BE	RVEL	5
C**	CONNECTED TO THE APPROPRIATE FILES THRU JOB CONTROL.	RVEL	6
C		RVEL	7
	COMMON /BLK6/ VX(240), VY(240), H(240), STRESS(3)	RVEL	8
	COMMON /FILE1/DUM(2000)	FETFIX7	1
	LEVEL 2,DUM	FETFIX7	2
C		RVEL	9
	DIMENSION NOD(135)	RVEL	10
C		RVEL	11
	IF (AT .GT. RAT) GO TO 200	RVEL	12
C		RVEL	13
C**	THE FOLLOWING SEGMENT SPECIFIES THE CORRELATION BETWEEN THE CAPE	RVEL	14
C**	AND FETRA NODE NUMBERS. READ IN THE FETRA NODE NUMBERS THAT	RVEL	15
C**	CORRESPOND TO THE CAPE NODES. THESE VALUES MUST BE READ IN AN	RVEL	16
C**	ASCENDING CAPE NODE NUMBER ORDER.	RVEL	17
C		RVEL	18
C**	AMAX IS THE TIME IN DAYS WHERE THE VELOCITIES WILL	RVEL	19
C**	BE STARTED OVER.	RVEL	20
C		RVEL	21
	READ (5,4000) AMAX	RVEL	22
4000	FORMAT (F10.0)	RVEL	23
	READ (5,5000) (NOD(I),I=1,NCNDS)	RVEL	24
5000	FORMAT (16I5)	RVEL	25
C		RVEL	26
C**	THE VALUES IN THE "DEFINE FILE" STATEMENT ARE DEPENDENT	RVEL	27
C**	UPON THE WAY THE FILES WERE CREATED IN THE GENERATING	RVEL	28
C**	PROGRAM.	RVEL	29
C		RVEL	30
	LUQ = 7	RVEL	31
	LUH = 8	RVEL	32
C		RVEL	33
	NREC = 1	RVEL	34
C	DEFINE FILE 7 (338,912,U,NREC)	RVEL	35
C	DEFINE FILE 8 (338,456,U,NREC)	RVEL	36
C	DEFINE FILE 7 (50,520,U,NREC)	RVEL	37
C	DEFINE FILE 8 (50,260,U,NREC)	RVEL	38
C		RVEL	39
200	A = AT	RVEL	40
	IF (A .GT. AMAX) A = AMOD(A,AMAX)	RVEL	41

AII = (A + VFREQ / 2.) / VFREQ + 1.	RVEL	42
II = AII	RVEL	43
IIVX=(II - 1) * NCNDS + 2 + 1	FETFIX5	2
IIM=(II - 1) * NCNDS + 1	FETFIX5	3
C READ (LUQ*II) (VX(NOD(I)), VY(NOD(I)), I=1,NCNDS)	RVEL	44
C READ (LUH*II) (H(NOD(I)),I=1,NCNDS)	RVEL	45
C	RVEL	46
C** SINCE THE DISCHARGE FILE GENERATED BY CAFE ACTUALLY CONTAINS	RVEL	47
C** DISCHARGES PER UNIT WIDTH, THESE VALUES MUST BE CONVERTED TO	RVEL	48
C** VELOCITIES. ALSO MUST CONVERT SECONDS TO DAYS.	RVEL	49
NCNDS2=NCNDS * 2	FETFIX5	4
CALL RDABSF (LUQ,DUM(1),NCNDS2,IIVX)	FETFIX5	5
IF(UNIT(LUQ)) 220,210,210	FETFIX5	6
C	FETFIX5	7
210 CONTINUE	FETFIX5	8
C	FETFIX5	9
WRITE(6,20010)	FETFIX5	10
20010 FORMAT(5X,"DIRECT ACCESS ERROR ON VELOCITY FILE")	FETFIX5	11
STOP	FETFIX5	12
C	FETFIX5	13
220 CONTINUE	FETFIX5	14
C	FETFIX5	15
ICNT=1	FETFIX5	16
DO 225 I=1,NCNDS	FETFIX5	17
VX(NOD(I))=DUM(ICNT)	FETFIX5	18
ICNT=ICNT + 1	FETFIX5	19
VY(NOD(I))=DUM(ICNT)	FETFIX5	20
ICNT=ICNT + 1	FETFIX5	21
225 CONTINUE	FETFIX5	22
C	FETFIX5	23
CALL RDABSF (LUH,DUM(1),NCNDS,IIM)	FETFIX5	24
IF(UNIT(LUH))240,230,230	FETFIX5	25
C	FETFIX5	26
230 CONTINUE	FETFIX5	27
C	FETFIX5	28
WRITE (6,20020)	FETFIX5	29
20020 FORMAT(5X,"DIRECT ACCESS ERROR ON DEPTH FILE")	FETFIX5	30
STOP	FETFIX5	31
C	FETFIX5	32
240 CONTINUE	FETFIX5	33
C	FETFIX5	34
ICNT=1	FETFIX5	35
DO 245 I=1,NCNDS	FETFIX5	36
H(NOD(I))=DUM(ICNT)	FETFIX5	37
ICNT=ICNT + 1	FETFIX5	38
245 CONTINUE	FETFIX5	39
C	FETFIX5	40
C	RVEL	50
DO 300 I=1,NCNDS	RVEL	51
VX(NOD(I)) = VX(NOD(I)) / H(NOD(I)) * 3600. * 24.	RVEL	52
VY(NOD(I)) = VY(NOD(I)) / H(NOD(I)) * 3600. * 24.	RVEL	53
300 CONTINUE	RVEL	54
RETURN	RVEL	58
END	RVEL	59
SUBROUTINE RWAVE (AT,WFREQ)	RWAVE	2
C	RWAVE	3
C** THIS SUBROUTINE DETERMINES WHICH FILE TO READ THE WAVE	RWAVE	4
C** CHARACTERISTICS FROM, BASED ON THE WIND VELOCITY AND	RWAVE	5
C** DIRECTION READ IN SUBROUTINE RWIND.	RWAVE	6
C	RWAVE	7
C CHARACTER*5 ZTYP	FETFIX1	12
C	RWAVE	9

	COMMON /WIND/ WVEL(700), WANG(700)	RWAVE	10
	COMMON/WAVE/ NODE(240), NC(135), A(135,10), K(135,10)	RWAVE	11
1	, W(135,10), HB(135), ALPHAC(135), WAVE, N, D, VIS	RWAVE	12
2	, DPTH(135)	RWAVE	13
	COMMON /ZTYPE/ ZTYP(135)	RWAVE	14
	COMMON/FILE/ DUMM(2000),NVAR(2,100)	FETFIX5	41
	DIMENSION IDUMM(2000)	FETFIX5	42
	LEVEL 2, IDUMM,DUMM,NVAR	FETFIX5	43
	EQUIVALENCE (IDUMM(1),DUMM(1))	FETFIX5	44
C		FETFIX5	45
C	LOGICAL WAVE	RWAVE	15
C	REAL K	RWAVE	16
C		RWAVE	17
C		RWAVE	18
C		RWAVE	19
	DIMENSION NDS(135), HINFS(20,18), DEPTH(20,18), ALFS(20,18),	RWAVE	20
1	HBS(20,18), WLS(20,18)	RWAVE	21
	CALL RDABSF (3, IDUMM(1),2,1)	FETFIX5	46
	IF(UNIT(3)) 10,5,5	FETFIX5	47
S	CONTINUE	FETFIX5	48
C		FETFIX5	49
	WRITE(6,20010)	FETFIX5	50
20010	FORMAT(" DIRECT ACCESS ERROR IN RWAVE")	FETFIX5	51
	STOP	FETFIX5	52
C		FETFIX5	53
10	CONTINUE	FETFIX5	54
C		FETFIX5	55
	NREC=IDUMM(1)	FETFIX5	56
	NVLOC=IDUMM(2)	FETFIX5	57
C		FETFIX5	58
	CALL RDABSF(3,NVAR(1,1),NREC * 2,NVLOC)	FETFIX5	59
C		FETFIX5	60
	IF(UNIT(3)) 12,5,5	FETFIX5	61
C		FETFIX5	62
12	CONTINUE	FETFIX5	63
C		FETFIX5	64
C		RWAVE	22
	ACCOUNT = (AT + WFREQ / 2.) / WFREQ + 1.	RWAVE	23
	NCOUNT = ACCOUNT	RWAVE	24
	ANG = WANG(NCOUNT)	RWAVE	25
	VEL = WVEL(NCOUNT)	RWAVE	26
C		RWAVE	27
C**	CHECK IF THE WIND IS BLOWING FROM OVERLAND. IF IT IS, ASSUME	RWAVE	28
C**	THERE ARE NO WAVES.	RWAVE	29
C		RWAVE	30
	IF((ANG .LT. 11.25).OR.(ANG .GT. 191.25)) GO TO 1000	RWAVE	31
	IF (ANG .GE. 33.75) GO TO 20	RWAVE	32
C		RWAVE	33
C**	NNE WAVES.	RWAVE	34
C		RWAVE	35
	OPEN (UNIT=3,NAME="NNEWAVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RWAVE	36
C	OPEN (UNIT=4,NAME="NNE SURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RWAVE	37
	GO TO 90	RWAVE	38
20	IF (ANG .GE. 56.25) GO TO 30	RWAVE	39
C		RWAVE	40
C**	NE WAVES.	RWAVE	41
C		RWAVE	42
	OPEN (UNIT=3,NAME="NEWAVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RWAVE	43
C	OPEN (UNIT=4,NAME="NESURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RWAVE	44
	GO TO 90	RWAVE	45
30	IF (ANG .GE. 78.75) GO TO 40	RWAVE	46
C		RWAVE	47

C**	ENE WAVES.	RHAVE	48
C		RHAVE	49
C	OPEN (UNIT#3,NAME="ENE#AVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RHAVE	50
C	OPEN (UNIT#4,NAME="ENESURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RHAVE	51
	GO TO 90	RHAVE	52
	40 IF (ANG ,GE. 101.25) GO TO 50	RHAVE	53
C		RHAVE	54
C**	E WAVES.	RHAVE	55
C		RHAVE	56
C	OPEN (UNIT#3,NAME="E#AVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RHAVE	57
C	OPEN (UNIT#4,NAME="ESURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RHAVE	58
	GO TO 90	RHAVE	59
	50 IF (ANG ,GE. 123.75) GO TO 60	RHAVE	60
C		RHAVE	61
C**	ESE WAVES.	RHAVE	62
C		RHAVE	63
C	OPEN (UNIT#3,NAME="ESE#AVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RHAVE	64
C	OPEN (UNIT#4,NAME="ESESURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RHAVE	65
	GO TO 90	RHAVE	66
	60 IF (ANG ,GE. 146.25) GO TO 70	RHAVE	67
C		RHAVE	68
C**	SZ WAVES.	RHAVE	69
C		RHAVE	70
C	OPEN (UNIT#3,NAME="SE#AVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RHAVE	71
C	OPEN (UNIT#4,NAME="SESURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RHAVE	72
	GO TO 90	RHAVE	73
	70 IF (ANG ,GE. 168.75) GO TO 80	RHAVE	74
C		RHAVE	75
C**	SSE WAVES.	RHAVE	76
C		RHAVE	77
C	OPEN (UNIT#3,NAME="SSE#AVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RHAVE	78
C	OPEN (UNIT#4,NAME="SSESURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RHAVE	79
	GO TO 90	RHAVE	80
C		RHAVE	81
C**	S WAVES.	RHAVE	82
C		RHAVE	83
	80 CONTINUE	RHAVE	84
C	OPEN (UNIT#3,NAME="S#AVZON.DAT",TYPE="OLD",ACCESS="DIRECT")	RHAVE	85
C	OPEN (UNIT#4,NAME="SSURF.DT2",TYPE="OLD",ACCESS="DIRECT")	RHAVE	86
C		RHAVE	87
C**	THE FOLLOWING 2 EQUATIONS FOR CALCULATING WAVE PERIOD AND	RHAVE	88
C**	DEEPWATER WAVE HEIGHT ARE BASED ON THE METHOD DEVELOPED BY	RHAVE	89
C**	HASSELMANN ET AL, 1976 AS USED BY THE U. S. ARMY CORPS OF	RHAVE	90
C**	ENGINEERS, THE EQUATIONS ARE FOR FULLY DEVELOPED WAVES.	RHAVE	91
C		RHAVE	92
	90 PER = VEL/1.372	RHAVE	93
	HINF = .02478*VEL**2	RHAVE	94
C		RHAVE	95
C**	IF THE DEEPWATER WAVE HEIGHT IS LESS THAN .5 FT.	RHAVE	96
C**	OR THE WAVE PERIOD IS LESS THAN 2. SEC.	RHAVE	97
C**	ASSUME THERE ARE NO WAVES.	RHAVE	98
C		RHAVE	99
	IF (HINF#3.2808 ,LT. .5) GO TO 1000	RHAVE	100
	IF (PER ,LT. 2.) GO TO 1000	RHAVE	101
	IF (PER ,GE. 4.) GO TO 120	RHAVE	102
	PER = 3.	RHAVE	103
	GO TO 160	RHAVE	104
120	IF (PER ,GE. 6.) GO TO 130	RHAVE	105
	PER = 5.	RHAVE	106
	GO TO 160	RHAVE	107
130	IF (PER ,GE. 8.) GO TO 140	RHAVE	108
	PER = 7.	RHAVE	109

GO TO 160	RWAVE	110
140 IF (PER .GE. 10.) GO TO 150	RWAVE	111
PER = 9.	RWAVE	112
GO TO 160	RWAVE	113
150 PER = 11.	RWAVE	114
160 II = (PER-1)/2	RWAVE	115
C READ (3*II) NV, (NDS(I), A(I,1), K(I,1), I=1, NV)	RWAVE	116
NV=NVAR(1,II)	FETFIX5	65
NVLOC=NVAR(2,II)	FETFIX5	66
CALL RDABSF(3, IDUMM(1), NV * 3, NVLOC)	FETFIX5	67
IF(UNIT(3)) 165, 162, 162	FETFIX5	68
C	FETFIX5	69
162 CONTINUE	FETFIX5	70
C	FETFIX5	71
WRITE(6, 20010)	FETFIX5	72
STOP	FETFIX5	73
C	FETFIX5	74
165 CONTINUE	FETFIX5	75
C	FETFIX5	76
ICNT=1	FETFIX5	77
DO 170 I=1, NV	FETFIX5	78
NDS(I)=IDUMM(ICNT)	FETFIX5	79
ICNT=ICNT + 1	FETFIX5	80
A(I,1)=DUMM(ICNT)	FETFIX5	81
ICNT=ICNT + 1	FETFIX5	82
K(I,1)=DUMM(ICNT)	FETFIX5	83
ICNT=ICNT + 1	FETFIX5	84
170 CONTINUE	FETFIX5	85
C	FETFIX5	86
PI = ACOS(-1.)	RWAVE	117
DO 300 I=1, NV	RWAVE	118
NC(I) = 1	RWAVE	119
NODE(NDS(I)) = I	RWAVE	120
A(I,1) = A(I,1)*MINF/2.	RWAVE	121
K(I,1) = 2.*PI/K(I,1)*3.2808	RWAVE	122
H(I,1) = 2.*PI/PER	RWAVE	123
ZTYP(I) = "WAVE,"	RWAVE	124
C WRITE (6, 4000) I, A(I,1), K(I,1), H(I,1)	RWAVE	125
C4000 FORMAT (' I, A(I,1), K(I,1), H(I,1) = ', I5, 1P3E12, 4)	RWAVE	126
300 CONTINUE	RWAVE	127
C	RWAVE	128
C** THE FOLLOWING SEGMENT READS THE SURF ZONE DATA.	RWAVE	129
C** IT ASSUMES THAT THE SURF ZONE DATA WAS GENERATED	RWAVE	130
C** FOR DEEPWATER WAVE HEIGHTS FROM 1 TO 16 FEET IN	RWAVE	131
C** 1 FOOT INCREMENTS.	RWAVE	132
C	RWAVE	133
NUMNOD = 18	RWAVE	134
NUMHT = 18	RWAVE	135
C READ (4*II) (NDS(I), (HINFS(I,J), DEPTH(I,J), HBS(I,J), ALFS(I,J),	RWAVE	136
C 1 HLS(I,J), J=1, NUMHT), I=1, NUMNOD)	FETFIX1	13
C	FETFIX5	67
NUMDS=5 * NUMHT * NUMNOD + NUMNOD	FETFIX5	88
NLOC=(II - 1) * NUMDS + 1	FETFIX5	89
CALL RDABSF (4, IDUMM(1), NUMDS, NLOC)	FETFIX5	90
IF(UNIT(4)) 305, 302, 302	FETFIX5	91
C	FETFIX5	92
302 CONTINUE	FETFIX5	93
C	FETFIX5	94
WRITE(6, 20010)	FETFIX5	95
STOP	FETFIX5	96
C	FETFIX5	97
305 CONTINUE	FETFIX5	98

<pre> C ICNT=1 DO 320 I=1,NUMNOD NDS(I)=IDUMM(ICNT) ICNT=ICNT + 1 DO 310 J=1,NUMHT HINFS(I,J)=DUMM(ICNT) ICNT=ICNT + 1 DEPTH(I,J)=DUMM(ICNT) ICNT=ICNT + 1 HBS(I,J)=DUMM(ICNT) ICNT=ICNT + 1 ALFS(ICNT)=DUMM(ICNT) ICNT=ICNT + 1 WLS(ICNT)=DUMM(ICNT) ICNT=ICNT + 1 310 CONTINUE C 320 CONTINUE C L = NV + 1 DO 500 I=1,NUMNOD NC(L) = 1 NODE(NDS(I)) = L ZTYP(L) = "SURF," DO 400 J=1,NUMHT DELH = HINFS(I,J) - HINF * 3.2808 400 IF (ABS(DELH) .LE. .5) GO TO 350 GO TO 410 350 ALPHAC(L) = ALFS(I,J) HB(L) = HBS(I,J) / 3.2808 K(L,1) = 2. * PI / WLS(I,J) * 3.2808 DPTH(L) = DEPTH(I,J) / 3.2808 GO TO 490 410 CONTINUE C C** IF NO SURF DATA IS APPLICABLE TO THE NODE C** ASSUME THE WATER IS CALM. C ALPHAC(L)=0. HB(L)=0. K(L,1)=9999. WRITE (6,6000) NDS(I), HINF 6000 FORMAT (1X,"SURF ZONE DATA NOT AVAILABLE FOR NODE ",I5/ 1 1X,"DEEPWATER WAVE HEIGHT = ",1PE10.3) WRITE(6,6010) 6010 FORMAT(1X,"THE WATER AT THIS NODE IS ASSUMED CALM.") 490 CONTINUE C WRITE (6,4001) L, ALPHAC(L), HB(L), K(L,1), DPTH(L) C4001 FORMAT (" L, ALPHAC(L), HB(L), K(L,1), DPTH(L) = ",I5, C 1 1P4E12.4) L = L + 1 500 CONTINUE GO TO 2000 1000 DO 1010 I=1,409 1010 NODE(I) = 0 WRITE (6,2010) AT,ANG,VEL 2010 FORMAT (/ " AT TIME =",F8.4," DAYS, THE WAVES ARE TOO SMALL" 1 " TO CAUSE APPRECIABLE TRANSPORT,"/" WIND DIRECTION =", 2 F8.4," DEGREES FROM TRUE NORTH,"/" WIND VELOCITY =", F8.4, 3 " M/SEC.") 2000 NCOUNT = NCOUNT+1 </pre>	<pre> FETFIX5 99 FETFIX5 100 FETFIX5 101 FETFIX5 102 FETFIX5 103 FETFIX5 104 FETFIX5 105 FETFIX5 106 FETFIX5 107 FETFIX5 108 FETFIX5 109 FETFIX5 110 FETFIX5 111 FETFIX5 112 FETFIX5 113 FETFIX5 114 FETFIX5 115 FETFIX5 116 FETFIX5 117 FETFIX5 118 RWAVE 138 RWAVE 139 RWAVE 140 RWAVE 141 RWAVE 142 RWAVE 143 RWAVE 144 RWAVE 145 RWAVE 146 RWAVE 147 RWAVE 148 RWAVE 149 RWAVE 150 RWAVE 151 RWAVE 152 RWAVE 153 RWAVE 154 RWAVE 155 RWAVE 156 RWAVE 157 RWAVE 158 RWAVE 159 RWAVE 160 RWAVE 161 RWAVE 162 RWAVE 163 RWAVE 164 RWAVE 165 RWAVE 166 RWAVE 167 RWAVE 168 RWAVE 169 RWAVE 170 RWAVE 171 RWAVE 172 RWAVE 173 RWAVE 174 RWAVE 175 RWAVE 176 RWAVE 177 RWAVE 178 RWAVE 179 </pre>
--	---

CALL CLOSE(3)	RWAVE	180
CALL CLOSE(4)	RWAVE	181
RETURN	RWAVE	182
END	RWAVE	183
SUBROUTINE RWIND (WFREQ,LW)	RWIND	2
C	RWIND	3
C** THIS SUBROUTINE READS THE WIND DATA REQUIRED FOR COMPUTATION	RWIND	4
C** OF THE WAVE CHARACTERISTICS.	RWIND	5
C	RWIND	6
COMMON /WIND/ WVEL(700), WANG(700)	RWIND	7
C	RWIND	8
C** LW = NUMBER OF WIND DATA POINTS TO BE INPUT. THE FIRST WIND	RWIND	9
C** DATA POINT WILL BE USED AT TIME=0, DAYS.	RWIND	10
C** WFREQ = TIME INTERVAL IN DAYS BETWEEN EACH WIND DATA POINT.	RWIND	11
C** WANG = DIRECTION FROM WHICH WIND IS BLOWING, DEGREES FROM	RWIND	12
C** TRUE NORTH MEASURED CW.	RWIND	13
C** WVEL = WIND VELOCITY, M/SEC.	RWIND	14
C	RWIND	15
READ (5,5000) LW,WFREQ	RWIND	16
READ (5,5010) ((WVEL(I), WANG(I)),I=1,LW)	RWIND	17
C	RWIND	18
C WRITE (6,6000) (WVEL(I),WANG(I),I=1,LW)	RWIND	19
C6000 FORMAT (' ***** WVEL(I),WANG(I) = ',8F10.2)	RWIND	20
RETURN	RWIND	21
5000 FORMAT (15,F10.0)	RWIND	22
5010 FORMAT (8E10.2)	RWIND	23
END	RWIND	24
SUBROUTINE SAND (M, ID, T, AREA)	SAND	2
C	SAND	3
C THIS SUBROUTINE COMPUTES CAPACITY OF SEDIMENT LOAD	SAND	4
C AND THEN SUBTRACT IT FROM THE ACTUAL LOAD TO OBTAIN THE AMOUNT OF	SAND	5
C SEDIMENT RESUSPENSION OR DEPOSITION	SAND	6
C	SAND	7
C CHARACTER*5 ZTYP	FETFIX1	14
REAL K	SAND	9
COMMON NDS	SAND	10
COMMON /BLK1/CT(240),C(9,240)	SAND	11
COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)	SAND	12
COMMON /BLK7/NOD(240,6),X(240),Y(240)	SAND	13
COMMON /BLK11/D50(3,100),BD50(100),BR(3,100),SD(3,100)	SAND	14
COMMON /BLK14/GBA(100,10),GBB(100,10),GBC(100,10),GBD(100,10),	SAND	15
1 GBE(100,10),GBF(100,10),GBG(100,10),PQR	SAND	16
COMMON /BLK15/ILAYR(100,3),XYS0(100),BDIV(100),NBED(100),BED(100),	SAND	17
1 RHOSED(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)	SAND	18
COMMON/WAVE/ NODE(240), NC(135), A(135,10), K(135,10),	SAND	19
1 MW(135,10), MB(135), ALPHAC(135), WAVE, N, D	SAND	20
2 , VIS, DPTH8(135)	SAND	21
COMMON /ZTYPE/ ZTYP(135)	SAND	22
LOGICAL WAVE	SAND	23
DIMENSION QS(3),QSA(3)	SAND	24
DIMENSION QSS(3)	FETFIX8	6
DIMENSION NCHECK(240), QCHECK(240)	SAND	25
DATA EPSI /1.E-10/	SAND	26
C	SAND	27
C** NCHECK IS USED TO ENSURE THAT THE SEDIMENT CAPACITY SUBROUTINES	SAND	28
C** (DUBOY, WAVSAN, AND SURFTR) ARE CALLED ONLY ONCE FOR EACH NODE	SAND	29
C** AT A GIVEN TIME STEP. QCHECK STORES THE RESULT FROM EACH NODE.	SAND	30
C	SAND	31
PI = ACOS(-1.0)	SAND	32
IF(M.NE.1) GO TO 3	SAND	33
DO 2 I=1,NDS	SAND	34
NCHECK(I)=0	SAND	35

2	CONTINUE	SAND	36
3	SD(ID,M)=0.0	SAND	37
	SR(ID,M)=0.0	SAND	38
	RS=0.0	SAND	39
	RS=0.0	SAND	40
	ILAYR(M, ID)=0	SAND	41
	TOTAL = GBA(M,NBED(M))/RHOSD(1)+GBR(M,NBED(M))/RHOSD(2)+	SAND	42
1	GBC(M,NBED(M))/RHOSD(3)	SAND	43
	XDTOP=(1.-PUR)/TOTAL	SAND	44
	XNT(M, ID)=XYSO(M)*GBC(M,NBED(M))*XDTOP	SAND	45
	DO 400 I=1,3	SAND	46
	NI=NOD(M,I)	SAND	47
	IF(NCHECK(NI).EQ.0) GO TO 5	SAND	48
	QS(I)=QCHECK(NI)	SAND	49
	GO TO 400	SAND	50
5	N=NODE(NI)	SAND	51
	QS(I) = 0.0	SAND	52
	QSS(I) = 0.0	FETFIX8	7
	D=D50(ID,M)	SAND	53
	UEXT=SQRT(VX(NI)**2+VY(NI)**2)	SAND	54
	IF(N.GT.0) GO TO 10	SAND	55
	CALL DUBOY(ID,M,I,QSS(I))	FETFIX8	8
10	IF(ZTYP(N).EQ."HAVE,") CALL WAVSAN(ID,H(NI),UEXT, QS(I))	SAND	58
C	IF(ZTYP(N).EQ."SURF,") AND.HB(N).GT.1.UE=10)	SAND	59
C	1 CALL SURFTR(ID,H(NI),UEXT, QS(I))	SAND	60
	IF(ZTYP(N).EQ."SURF,") CALL SURFTR(ID,H(NI),UEXT, QS(I))	SAND	61
399	CONTINUE	SAND	62
	NCHECK(NI)=1	SAND	63
	QCHECK(NI) = QS(I)+QSS(I)	FETFIX8	9
400	CONTINUE	SAND	65
	OCTH=0.	SAND	66
	S=0.	SAND	67
	SN=0.	SAND	68
	DO 100 I=1,3	SAND	69
	IF(I=2) 60,70,80	SAND	70
		SAND	71
C		SAND	72
C**	N1 IS THE NODE AT WHICH THE SCOUR OR EROSION COEFFICIENT IS	SAND	73
C**	BEING CALCULATED.	SAND	74
C**	N2 IS THE NEXT CORNER NODE COUNTER=CLOCKWISE.	SAND	75
C**	N3 IS THE SECOND CORNER NODE COUNTER=CLOCKWISE.	SAND	76
C		SAND	77
60	N1=NOD(M,1)	SAND	78
	N2=NOD(M,2)	SAND	79
	N3=NOD(M,3)	SAND	80
	I1=1	SAND	81
	I2=2	SAND	82
	I3=3	SAND	83
	GO TO 90	SAND	84
70	N1=NOD(M,2)	SAND	85
	N2=NOD(M,3)	SAND	86
	N3=NOD(M,1)	SAND	87
	I1=2	SAND	88
	I2=3	SAND	89
	I3=1	SAND	90
	GO TO 90	SAND	91
80	N1=NOD(M,3)	SAND	92
	N2=NOD(M,1)	SAND	93
	N3=NOD(M,2)	SAND	94
	I1=3	SAND	95
	I2=1	SAND	96
	I3=2	SAND	97
90	CONTINUE	SAND	97

VM=SQRT(VX(N1)**2+VY(N1)**2)	SAND	98
IF(VM .EQ. 0.) GO TO 91	SAND	99
CTH=QS(I)/VM	SAND	100
GO TO 93	SAND	101
C	SAND	102
C** IF VM IS ZERO, ASSUME THAT EVENTUALLY ALL THE SUSPENDED	SAND	103
C** SEDIMENT WILL SETTLE OUT.	SAND	104
C	SAND	105
91 CTH=0.	SAND	106
93 CAC=C(ID,N1)*H(N1)	SAND	107
DCTH=DCTH+(CTH-CAC)	SAND	108
IF(VM .EQ. 0.) GO TO 100	SAND	109
QSA(I)=C(ID,N1)*H(N1)*VM	SAND	110
A1=Y(N2)-Y(N1)	SAND	111
A2=X(N2)-X(N1)	SAND	112
A3=Y(N3)-Y(N1)	SAND	113
A4=X(N3)-X(N1)	SAND	114
C	SAND	115
C** ALPHA1 IS THE ANGLE BETWEEN THE X-AXIS AND THE ELEMENT SIDE, N1	SAND	116
C** TO N2.	SAND	117
C** ALPHA2 IS THE ANGLE BETWEEN THE X-AXIS AND THE ELEMENT SIDE N1	SAND	118
C** TO N3.	SAND	119
C** BETA IS THE ANGLE BETWEEN THE X-AXIS AND THE VELOCITY VECTOR AT N1	SAND	120
C** DEN1 IS THE DISTANCE BETWEEN N1 AND N2.	SAND	121
C** DEN2 IS THE DISTANCE BETWEEN N1 AND N3.	SAND	122
C	SAND	123
ALPHA1=ATAN2(A1,A2)	SAND	124
IF(ALPHA1 .LT. 0.) ALPHA1=ALPHA1+2.*PI	SAND	125
ALPHA2=ATAN2(A3,A4)	SAND	126
IF(ALPHA2 .LT. 0.) ALPHA2=ALPHA2+2.*PI	SAND	127
BETA=ATAN2(VY(N1),VX(N1))	SAND	128
IF(BETA .LT. 0.) BETA=BETA+2.*PI	SAND	129
DEN1 =SQRT((X(N1)-X(N2))**2+(Y(N1)-Y(N2))**2)	SAND	130
DEN2 =SQRT((X(N1)-X(N3))**2+(Y(N1)-Y(N3))**2)	SAND	131
ANGLE=BETA-ALPHA1	SAND	132
C	SAND	133
C** CHECK IF THERE IS A VELOCITY COMPONENT TOWARDS N2.	SAND	134
C	SAND	135
IF(COS(ANGLE) .LE. 0.) GO TO 92	SAND	136
SN=SN+1.	SAND	137
S=S+(QS(I2)-QSA(I1))*COS(ANGLE)/DEN1	SAND	138
92 ANGLE =BETA-ALPHA2	SAND	139
C	SAND	140
C** CHECK IF THERE IS A VELOCITY COMPONENT TOWARDS N3.	SAND	141
C	SAND	142
IF(COS(ANGLE) .LE. 0.) GO TO 100	SAND	143
SN=SN+1.	SAND	144
S=S+(QS(I3)-QSA(I1))*COS(ANGLE)/DEN2	SAND	145
100 CONTINUE	SAND	146
S2=S	SAND	147
IF(SN .GT. 0.) S2=S2/SN	SAND	148
DCTH=DCTH/3.	SAND	149
C	SAND	150
C** CHECK TO SEE IF THE TIME STEP IS TOO LARGE.	SAND	151
C** S2*T INDICATES THE AMOUNT OF SEDIMENT THAT WILL BE SCOURED OR	SAND	152
C** DEPOSITED DURING THE NEXT TIME STEP.	SAND	153
C** DCTH INDICATES THE DIFFERENCE BETWEEN THE THEORETICAL AND ACTUAL	SAND	154
C** SEDIMENT CAPACITY ON A PER UNIT HORIZONTAL AREA BASIS.	SAND	155
C	SAND	156
IF(DCTH .EQ. 0.) GO TO 110	SAND	157
RATIO=S2*T/DCTH	SAND	158
IF(RATIO .LE. 1.) GO TO 110	SAND	159

	DT=DCTH/S2	SAND	160
	WRITE(6,6000) M, S2	SAND	161
	WRITE(6,6001) DCTH, DT	SAND	162
110	CONTINUE	SAND	163
C		SAND	164
	IF (S2) 130,305,150	SAND	165
C	DEPOSITION	SAND	166
130	ILAYR(M, ID)=1	SAND	167
	SD(ID, M)=(-1)*S2	SAND	168
	N1 = NOD(M, 1)	SAND	169
	N2 = NOD(M, 2)	SAND	170
	N3 = NOD(M, 3)	SAND	171
	N4 = NOD(M, 4)	SAND	172
	N5 = NOD(M, 5)	SAND	173
	N6 = NOD(M, 6)	SAND	174
	H1 = H(N1)	SAND	175
	H2 = H(N2)	SAND	176
	H3 = H(N3)	SAND	177
	HAVG = (H1+H2+H3)/3.	SAND	178
	CAVG3 = (C(ID, N1)*H1+C(ID, N2)*H2+C(ID, N3)*H3+C(ID, N4)*(H1+H2)/2.	SAND	179
1	+C(ID, N5)*(H2+H3)/2.+C(ID, N6)*(H1+H3)/2.)/(6.*HAVG)	SAND	180
	IF (CAVG3, LE, EP8I) GO TO 305	SAND	181
	CAVG7 = (C(ID+4, N1)*H1+C(ID+4, N2)*H2+C(ID+4, N3)*H3+C(ID+4, N4)*	SAND	182
1	(H1+H2)/2.+C(ID+4, N5)*(H2+H3)/2.+C(ID+4, N6)*(H1+H3)/2.)/	SAND	183
2	(6.*HAVG)	SAND	184
	RSV3(M) = -SD(ID, M)*CAVG7/CAVG3	SAND	185
	GO TO 305	SAND	186
C		SAND	187
C	RESUSPENSION	SAND	188
C		SAND	189
150	SR(ID, M)=S2	SAND	190
C		SAND	191
C	CHECK THE AVAILABILITY OF SAND IN THE BED TO BE SCOURED	SAND	192
C	TO COMPUTE THE AVAILABILITY OF SAND IN BED LAYERS	SAND	193
C	A NUMBER OF BED LAYERS SCOURED IS PREDETERMINED BY OTHER SEDIMENT	SAND	194
C	(CLAY OR OTHER SAND)	SAND	195
	NB = NBED(M)	SAND	196
	SR(ID, M)=SR(ID, M)*T	SAND	197
	IF (SR(ID, M).GT.XNT(M, ID).AND.ILAYR(M, 1).GT.0) GO TO 200	SAND	198
	SR(ID, M)=AMIN1(SR(ID, M), XNT(M, ID))	SAND	199
	XNT(M, ID)*XNT(M, ID)=SR(ID, M)	SAND	200
	RSV3(M)=SR(ID, M)*GBF(M, NBED(M))	SAND	201
	GO TO 290	SAND	202
200	ILAYR(M, ID)=ILAYR(M, ID)+1	SAND	203
	RSV3(M)=RSV3(M)+XNT(M, ID)*GBF(M, NB)	SAND	204
	SR(ID, M)=SR(ID, M)-XNT(M, ID)	SAND	205
	RS=RS+XNT(M, ID)	SAND	206
	NB=NBED(M)-ILAYR(M, ID)	SAND	207
	IF (NB, LE, 0) GO TO 285	SAND	208
	TOTAL=GBA(M, NB)/RHOSED(1)+GBB(M, NB)/RHOSED(2)+GBC(M, NB)/RHOSED(3)	SAND	209
	XND=(1.-POR)/TOTAL	SAND	210
	XNT(M, ID)=BDIV(M)*GBC(M, NB)*XND	SAND	211
	IF (ILAYR(M, ID) .GE. ILAYR(M, 1)) GO TO 280	SAND	212
	IF (SR(ID, M) .GE. XNT(M, ID)) GO TO 200	SAND	213
280	RS=RS+AMIN1(SR(ID, M), XNT(M, ID))	SAND	214
	RSV3(M)=RSV3(M)+AMIN1(SR(ID, M), XNT(M, ID))*GBF(M, NB)	SAND	215
285	XNT(M, ID)*XNT(M, ID)=AMIN1(SR(ID, M), XNT(M, ID))	SAND	216
	SR(ID, M)=RS	SAND	217
290	SR(ID, M)=SR(ID, M)/T	SAND	218
	RSV3(M)=RSV3(M)/T	SAND	219
305	CONTINUE	SAND	220
	RETURN	SAND	221

6000	FORMAT(1X,"***WARNING***",/,	SAND	222
	\$1X,"THE COMPUTED DEPOSITION RATE FOR ELEMENT ",I3," IS ",1PE12.4,	SAND	223
	" KGF/DAY=***2.")	SAND	224
6001	FORMAT(1X,"THE THEORETICAL CAPACITY DIFFERENCE IS ",1PE12.4," KGF/	SAND	225
	\$M**2",/,	SAND	226
	\$1X,"THEREFORE THE TIME STEP SHOULD BE REDUCED TO A MAXIMUM OF ",	SAND	227
	\$E12.4," DAY.")	SAND	228
	END	SAND	229
	SUBROUTINE SAND2 (M,IO,T,AREA)	SAND2	2
C		SAND2	3
C	THIS SUBROUTINE COMPUTES THE SEDIMENT LOAD CAPACITY	SAND2	4
C	AND THEN SUBTRACT IT FROM THE ACTUAL LOAD TO OBTAIN THE AMOUNT OF	SAND2	5
C	SEDIMENT RESUSPENSION OR DEPOSITION	SAND2	6
C	THIS ROUTINE DECIDES NO. OF BED LAYERS TO BE SCoured	SAND2	7
C	SUBROUTINE CLAY OR THIS DECIDE NO. OF SCoured BED LAYERS	SAND2	8
C		SAND2	9
C	CHARACTER*5 ZTYP	FETFIX1	15
C	REAL K	SAND2	11
		SAND2	12
	COMMON NOS	SAND2	13
	COMMON /BLK1/CT(240),C(9,240)	SAND2	14
	COMMON /BLK6/VX(240),VY(240),M(240),STRESS(3)	SAND2	15
	COMMON /BLK7/NOD(240,6),X(240),Y(240)	SAND2	16
	COMMON /BLK11/D50(3,100),RDSO(100),SR(3,100),SD(3,100)	SAND2	17
	COMMON /BLK14/GBA(100,10),GRB(100,10),GBC(100,10),GBD(100,10),	SAND2	18
1	GBE(100,10),GRF(100,10),GRG(100,10),POR	SAND2	19
	COMMON /BLK15/ILAYR(100,3),XYSO(100),RDIIV(100),NBED(100),BED(100),	SAND2	20
1	RHOSED(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)	SAND2	21
	COMMON/WAVE/ NODE(240), NC(135), A(135,10), K(135,10),	SAND2	22
1	HW(135,10), HB(135), ALPHAC(135), WAVE, N, D	SAND2	23
2	, VIS, OPTHS(135)	SAND2	24
	COMMON /ZTYPE/ ZTYP(135)	SAND2	25
	LOGICAL WAVE	SAND2	26
	DIMENSION QS(3), QSA(3)	SAND2	27
	DIMENSION QSS(3)	FETFIX8	10
	DIMENSION NCHECK(240), QCHECK(240)	SAND2	28
	DATA EPSI /1.E=10/	SAND2	29
C		SAND2	30
C**	NCHECK IS USED TO ENSURE THAT THE SEDIMENT CAPACITY SUBROUTINES	SAND2	31
C**	(DUBOY, HAVSAN, AND SURFTR) ARE CALLED ONLY ONCE FOR EACH NODE	SAND2	32
C**	AT A GIVEN TIME STEP. QCHECK STORES THE RESULT FROM EACH NODE.	SAND2	33
C		SAND2	34
	PI=ACOS(-1.)	SAND2	35
	IF(M.NE.1) GO TO 3	SAND2	36
	DO 2 I=1,NOS	SAND2	37
	NCHECK(I)=0	SAND2	38
2	CONTINUE	SAND2	39
3	SD(ID,M)=0.0	SAND2	40
	SR(ID,M)=0.0	SAND2	41
	RSV1(M)=0.	SAND2	42
	RS=0.0	SAND2	43
	ILAYR(M,IO)=0	SAND2	44
	TOTAL = GBA(M,NBED(M))/RHOSED(1)+GBB(M,NBED(M))/RHOSED(2)+	SAND2	45
1	GBC(M,NBED(M))/RHOSED(3)	SAND2	46
	XDTP=(1.-POR)/TOTAL	SAND2	47
	XNT(M,IO)=XYSO(M)*GBA(M,NBED(M))*XDTP	SAND2	48
	DO 400 I=1,3	SAND2	49
	NI=NOD(M,I)	SAND2	50
	IF(NCHECK(NI).EQ.0) GO TO 5	SAND2	51
	QS(I)=QCHECK(NI)	SAND2	52
	GO TO 400	SAND2	53
5	N=NODE(NI)	SAND2	54

QS(I) = 0.0	SAND2	55
QSS(I) = 0.0	FETFIX8	11
D=050(ID,M)	SAND2	56
UEXT=SQRT(VX(NI)**2+VY(NI)**2)	SAND2	57
IF(N.GT.0) GO TO 10	SAND2	58
CALL DUBOY(ID,M,I,QSS(I))	FETFIX8	12
10 IF(ZTYP(N).EQ."WAVE,") CALL WAVSAN(ID,H(NI),UEXT,QS(I))	SAND2	61
IF(ZTYP(N).EQ."SURF,".AND.HB(N).GT.1.0E-10.AND.DPTH(N)	SAND2	62
1.GE.H(NI)) CALL SURFTR(ID,H(NI),UEXT,QS(I))	SAND2	63
C IF(ZTYP(N).EQ."SURF,") CALL SURFTR(ID,H(NI),UEXT,US(I))	SAND2	64
C WRITE (6,6020) M,N,I,NI,QS(I),ZTYP(N)	SAND2	65
C6020 FORMAT (" **** M, N, I, NI, QS(I), ZTYP(N) = ",4I5,1PE12.4,1X,A5)	SAND2	66
399 CONTINUE	SAND2	67
NCHECK(NI)=1	SAND2	68
QCHECK(NI) = QS(I) + QSS(I)	FETFIX8	13
400 CONTINUE	SAND2	70
DCTH=0.	SAND2	71
S=0.	SAND2	72
SN=0.	SAND2	73
DO 100 I=1,3	SAND2	74
IF(I=2) GO,70,80	SAND2	75
C	SAND2	76
C** N1 IS THE NODE AT WHICH THE SCOUR OR EROSION COEFFICIENT IS	SAND2	77
C** BEING CALCULATED.	SAND2	78
C** N2 IS THE NEXT CORNER NODE COUNTER=CLOCKWISE.	SAND2	79
C** N3 IS THE SECOND CORNER NODE COUNTER=CLOCKWISE.	SAND2	80
C	SAND2	81
60 N1=NOD(M,1)	SAND2	82
N2=NOD(M,2)	SAND2	83
N3=NOD(M,3)	SAND2	84
I1=1	SAND2	85
I2=2	SAND2	86
I3=3	SAND2	87
GO TO 90	SAND2	88
70 N1=NOD(M,2)	SAND2	89
N2=NOD(M,3)	SAND2	90
N3=NOD(M,1)	SAND2	91
I1=2	SAND2	92
I2=3	SAND2	93
I3=1	SAND2	94
GO TO 90	SAND2	95
80 N1=NOD(M,3)	SAND2	96
N2=NOD(M,1)	SAND2	97
N3=NOD(M,2)	SAND2	98
I1=3	SAND2	99
I2=1	SAND2	100
I3=2	SAND2	101
90 CONTINUE	SAND2	102
VM=SQRT(VX(N1)**2+VY(N1)**2)	SAND2	103
IF(VM.EQ.0.) GO TO 91	SAND2	104
CTH=QS(I)/VM	SAND2	105
GO TO 93	SAND2	106
C	SAND2	107
C** IF VM IS ZERO, ASSUME THAT EVENTUALLY ALL THE SUSPENDED	SAND2	108
C** SEDIMENT WILL SETTLE OUT.	SAND2	109
C	SAND2	110
91 CTH=0.	SAND2	111
93 CAC=C(ID,N1)*H(N1)	SAND2	112
DCTH=DCTH+(CTH=CAC)	SAND2	113
IF(VM.EQ.0.) GO TO 100	SAND2	114
QSA(I)=C(ID,N1)*H(N1)*VM	SAND2	115
A1=Y(N2)-Y(N1)	SAND2	116

A2=X(N2)-X(N1)	SAND2	117
A3=Y(N3)-Y(N1)	SAND2	118
A4=X(N3)-X(N1)	SAND2	119
C	SAND2	120
C** ALPHA1 IS THE ANGLE BETWEEN THE X-AXIS AND THE ELEMENT SIDE, N1	SAND2	121
C** TO N2.	SAND2	122
C** ALPHA2 IS THE ANGLE BETWEEN THE X-AXIS AND THE ELEMENT SIDE N1	SAND2	123
C** TO N3.	SAND2	124
C** BETA IS THE ANGLE BETWEEN THE X-AXIS AND THE VELOCITY VECTOR AT N1	SAND2	125
C** DEN1 IS THE DISTANCE BETWEEN N1 AND N2.	SAND2	126
C** DEN2 IS THE DISTANCE BETWEEN N1 AND N3.	SAND2	127
C	SAND2	128
ALPHA1=ATAN2(A1,A2)	SAND2	129
IF(ALPHA1 .LT. 0.) ALPHA1=ALPHA1+2.*PI	SAND2	130
ALPHA2=ATAN2(A3,A4)	SAND2	131
IF(ALPHA2 .LT. 0.) ALPHA2=ALPHA2+2.*PI	SAND2	132
BETA=ATAN2(VY(N1),VX(N1))	SAND2	133
IF(BETA .LT. 0.) BETA=BETA+2.*PI	SAND2	134
DEN1 =SQRT((X(N1)-X(N2))**2+(Y(N1)-Y(N2))**2)	SAND2	135
DEN2 =SQRT((X(N1)-X(N3))**2+(Y(N1)-Y(N3))**2)	SAND2	136
ANGLE=BETA-ALPHA1	SAND2	137
C WRITE (6,7000) M,N1,N2,N3,ALPHA1,ALPHA2,BETA,ANGLE	SAND2	138
C7000 FORMAT ("M,N1,N2,N3,ALPHA1,ALPHA2,BETA,ANGLE = ",6I5,1P4E12.4)	SAND2	139
C	SAND2	140
C** CHECK IF THERE IS A VELOCITY COMPONENT TOWARDS N2.	SAND2	141
C	SAND2	142
IF(COS(ANGLE) .LE. 0.) GO TO 92	SAND2	143
SN=SN+1.	SAND2	144
S=S+(QS(I2)-QSA(I1))*COS(ANGLE)/DEN1	SAND2	145
C WRITE (6,1198) M,N1,N2,N3,I1,I2,S,QSA(I1),QS(I2)	SAND2	146
C1198 FORMAT (" M,N1,N2,N3,I1,I2,S,QSA(I1),QS(I2) = ",6I5,1P3E12.4)	SAND2	147
92 ANGLE =BETA-ALPHA2	SAND2	148
C	SAND2	149
C** CHECK IF THERE IS A VELOCITY COMPONENT TOWARDS N3.	SAND2	150
C	SAND2	151
IF(COS(ANGLE) .LE. 0.) GO TO 100	SAND2	152
SN=SN+1.	SAND2	153
S=S+(QS(I3)-QSA(I1))*COS(ANGLE)/DEN2	SAND2	154
C WRITE (6,1199) M,N1,N2,N3,I1,I3,S,QSA(I1),QS(I3)	SAND2	155
C1199 FORMAT (" M,N1,N2,N3,I1,I3,S,QSA(I1),QS(I3) = ",6I5,1P3E12.4)	SAND2	156
100 CONTINUE	SAND2	157
S2=S	SAND2	158
IF(SN .GT. 0.) S2=S2/SN	SAND2	159
DCTH=DCTH/3.	SAND2	160
C	SAND2	161
C** CHECK TO SEE IF THE TIME STEP IS TOO LARGE.	SAND2	162
C** S2*T INDICATES THE AMOUNT OF SEDIMENT THAT WILL BE SCOURED OR	SAND2	163
C** DEPOSITED DURING THE NEXT TIME STEP.	SAND2	164
C** DCTH INDICATES THE DIFFERENCE BETWEEN THE THEORETICAL AND ACTUAL	SAND2	165
C** SEDIMENT CAPACITY ON A PER UNIT HORIZONTAL AREA BASIS.	SAND2	166
C	SAND2	167
IF(DCTH .EQ. 0.) GO TO 110	SAND2	168
RATIO=S2*T/DCTH	SAND2	169
IF(RATIO .LE. 1.) GO TO 110	SAND2	170
DT=DCTH/S2	SAND2	171
S2=DCTH/DT/2.0	SAND2	172
WRITE(6,6000) M, S2	SAND2	173
WRITE(6,6001) DCTH, DT	SAND2	174
110 CONTINUE	SAND2	175
C	SAND2	176
IF (S2) 130,305,150	SAND2	177
C DEPOSITION	SAND2	178

130	ILAYR(M, ID)=-1	SAND2	179
	SD(ID, M)=(-1)*S2	SAND2	180
	N1 = NOD(M, 1)	SAND2	181
	N2 = NOD(M, 2)	SAND2	182
	N3 = NOD(M, 3)	SAND2	183
	N4 = NOD(M, 4)	SAND2	184
	N5 = NOD(M, 5)	SAND2	185
	N6 = NOD(M, 6)	SAND2	186
	H1 = H(N1)	SAND2	187
	H2 = H(N2)	SAND2	188
	H3 = H(N3)	SAND2	189
	HAVG = (H1+H2+H3)/3.	SAND2	190
	CAVG1 = (C(ID, N1)*H1+C(ID, N2)*H2+C(ID, N3)*H3+C(ID, N4)*(H1+H2)/2.	SAND2	191
	1 +C(ID, N5)*(H2+H3)/2.+C(ID, N6)*(H1+H3)/2.)/(6.*HAVG)	SAND2	192
	IF (CAVG1.LE.EPSI) GO TO 305	SAND2	193
	CAVG5 = (C(ID+4, N1)*H1+C(ID+4, N2)*H2+C(ID+4, N3)*H3+C(ID+4, N4)*	SAND2	194
	1 (H1+H2)/2.+C(ID+4, N5)*(H2+H3)/2.+C(ID+4, N6)*(H1+H3)/2.)/	SAND2	195
	2 (6.*HAVG)	SAND2	196
	RSAV1(M) = -SD(ID, M)*CAVG5/CAVG1	SAND2	197
	GO TO 305	SAND2	198
C		SAND2	199
C	RESUSPENSION	SAND2	200
C		SAND2	201
150	SR(ID, M)=S2	SAND2	202
C		SAND2	203
C	TO COMPUTE A NUMBER OF LAYERS SCOURED IN ORDER TO RESUSPEND THE	SAND2	204
C	APPROPRIATE AMOUNT OF SAND	SAND2	205
C	ASSUME SAND IS MOST DIFFICULT TO SCOUR	SAND2	206
C	SR(ID, M)=SR(ID, M)*T	SAND2	207
	ILAYR(M, ID)=0	SAND2	208
	NB = NBED(M)	SAND2	209
	IF (SR(ID, M).GT.XNT(M, ID)) GO TO 200	SAND2	210
	RSAV1(M)=SR(ID, M)*GRD(M, NBED(M))	SAND2	211
	XNT(M, ID)=XNT(M, ID)-SR(ID, M)	SAND2	212
	GO TO 290	SAND2	213
200	ILAYR(M, ID)=ILAYR(M, ID)+1	SAND2	214
	IF (ILAYR(M, ID).EQ.NBED(M)) GO TO 280	SAND2	215
	RSAV1(M)=RSAV1(M)+XNT(M, ID)*GRD(M, NB)	SAND2	216
	SR(ID, M)=SR(ID, M)+XNT(M, ID)	SAND2	217
	RS=RS+XNT(M, ID)	SAND2	218
	NB=NBED(M)-ILAYR(M, ID)	SAND2	219
	TOTAL=GBA(M, NB)/RHOSED(1)+GBB(M, NB)/RHOSED(2)+GBC(M, NB)/RHOSED(3)	SAND2	220
	XND=(1.-POR)/TOTAL	SAND2	221
	XNT(M, ID)=HDIV(M)*GBA(M, NB)*XND	SAND2	222
	IF (ILAYR(M, ID).LT.(NBED(M)-1)) GO TO 270	SAND2	223
	ILAYR(M, ID)=ILAYR(M, ID)+1	SAND2	224
	GO TO 280	SAND2	225
270	IF (SR(ID, M).GE.XNT(M, ID)) GO TO 200	SAND2	226
280	RSAV1(M)=RSAV1(M)+AMIN1(SR(ID, M), XNT(M, ID))*GRD(M, NB)	SAND2	227
	RS=RS+AMIN1(SR(ID, M), XNT(M, ID))	SAND2	228
	XNT(M, ID)=XNT(M, ID)-AMIN1(SR(ID, M), XNT(M, ID))	SAND2	229
	SR(ID, M)=RS	SAND2	230
290	SR(ID, M)=SR(ID, M)/T	SAND2	231
	RSAV1(M)=RSAV1(M)/T	SAND2	232
305	CONTINUE	SAND2	233
	RETURN	SAND2	234
6000	FORMAT(1X, "*****WARNING***", /,	SAND2	235
	\$1X, "THE COMPUTED DEPOSITION RATE FOR ELEMENT ", I3, " IS ", 1PE12.4,	SAND2	236
	\$" KGF/DAY=M**2.")	SAND2	237
6001	FORMAT(1X, "THE THEORETICAL CAPACITY DIFFERENCE IS ", 1PE12.4, " KGF/	SAND2	238
	\$M**2", /,	SAND2	239
	\$1X, "THEREFORE THE TIME STEP SHOULD BE REDUCED TO A MAXIMUM OF ",	SAND2	240


```

SE12.4," DAY.")
END
SUBROUTINE SEDIME (M,AREA,ID)
C
C THIS ROUTINE CALCULATES COEFFICIENTS OF DECAY AND
C SOURCE TERMS IN THE SEDIMENT TRANSPORT CONVECTION-DIFFUSION
C EQUATION
C
COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)
COMMON /BLK7/NOD(240,6),X(240),Y(240)
COMMON /BLK9/OX(100),OY(100),EX(100),EY(100),ALFA(100),BETA(100)
9
,HS(100)
COMMON /BLK10/REL(6,6),SEL(6,6),REL(6)
COMMON /BLK11/D50(3,100),dD50(100),SR(3,100),SD(3,100)
COMMON /BLK17/QLATE(240),QSARA(7,100),QPNT(7,240),CRATE(3,100),
1 CD(4,240)
LEVEL 2,QLATE,QSARA,QPNT,CRATE,CD
C
C
N1=NOD(M,1)
N2=NOD(M,2)
N3=NOD(M,3)
N4=NOD(M,4)
N5=NOD(M,5)
N6=NOD(M,6)
H1 = H(N1)
H2 = H(N2)
H3 = H(N3)
C
C*** DECAY TERM ***
ALFA(M) = 0.0
C** ALFA(M) = QLATE(M) IF QLATE(M) IS CONSTANT WITHIN A ELEMENT
C
C*** SOURCE OR SINK TERM ***
C AREA SOCECE QSARA(ID,M)
A = AREA*QSARA(ID,M)/60.
B = AREA*(SR(ID,M)-SD(ID,M))/3.
C
REL(1) = A*(2.*H1-H2-H3)
REL(2) = -A*(H1-2.*H2+H3)
REL(3) = -A*(H1+H2-2.*H3)
REL(4) = B+A*(8.*H1+8.*H2+4.*H3)
REL(5) = B+A*(4.*H1+8.*H2+8.*H3)
REL(6) = B+A*(8.*H1+4.*H2+8.*H3)
C
C POINT SOURCE QPNT(ID,NDS)
B = 1.0
C
REL(1) = REL(1) +B*((30.*H1+6.*H2+6.*H3)*QPNT(ID,N1)
1=(4.*H1+4.*H2+H3)*QPNT(ID,N2) -(4.*H1-H2+4.*H3)*QPNT(ID,N3)
2+(12.*H1-8.*H2-4.*H3)*QPNT(ID,N4)-(4.*H1+12.*H2+12.*H3)*
3QPNT(ID,N5)+(12.*H1-4.*H2-8.*H3)*QPNT(ID,N6))
REL(2) = REL(2) +B*((4.*H1+4.*H2+H3)*QPNT(ID,N1)*(-1)
1+(6.*H1+30.*H2+6.*H3)*QPNT(ID,N2)+(H1-4.*H2-4.*H3)*QPNT(ID,N3)
2=(8.*H1-12.*H2+4.*H3)*QPNT(ID,N4)-(4.*H1-12.*H2+8.*H3)*QPNT(ID,N5)
3=(12.*H1+4.*H2+12.*H3)*QPNT(ID,N6))
REL(3) = REL(3) +B*((4.*H1-H2+4.*H3)*QPNT(ID,N1)*(-1)
1+(H1-4.*H2-4.*H3)*QPNT(ID,N2) +(6.*H1+6.*H2+30.*H3)*QPNT(ID,N3)
2=(12.*H1+12.*H2+4.*H3)*QPNT(ID,N4)-(4.*H1+8.*H2-12.*H3)*
3QPNT(ID,N5)-(8.*H1+4.*H2-12.*H3)*QPNT(ID,N6))
REL(4) = REL(4) +B*((12.*H1-8.*H2+4.*H3)*QPNT(ID,N1)
1=(8.*H1-12.*H2+4.*H3)*QPNT(ID,N2)-(12.*H1+12.*H2+4.*H3)*

```

```

SAND2 241
SAND2 242
SEDIME 2
SEDIME 3
SEDIME 4
SEDIME 5
SEDIME 6
SEDIME 7
SEDIME 8
SEDIME 9
SEDIME 10
SEDIME 11
SEDIME 12
SEDIME 13
SEDIME 14
SEDIME 15
FETFIX4 21
SEDIME 16
SEDIME 17
SEDIME 18
SEDIME 19
SEDIME 20
SEDIME 21
SEDIME 22
SEDIME 23
SEDIME 24
SEDIME 25
SEDIME 26
SEDIME 27
SEDIME 28
SEDIME 29
SEDIME 30
SEDIME 31
SEDIME 32
SEDIME 33
SEDIME 34
SEDIME 35
SEDIME 36
SEDIME 37
SEDIME 38
SEDIME 39
SEDIME 40
SEDIME 41
SEDIME 42
SEDIME 43
SEDIME 44
SEDIME 45
SEDIME 46
SEDIME 47
SEDIME 48
SEDIME 49
SEDIME 50
SEDIME 51
SEDIME 52
SEDIME 53
SEDIME 54
SEDIME 55
SEDIME 56
SEDIME 57
SEDIME 58
SEDIME 59
SEDIME 60

```

```

2QPNT(ID,N3)+(96.*H1+96.*H2+32.*H3)*QPNT(ID,N4)+(32.*H1+48.*H2+32.*H3)*QPNT(ID,N5)+(48.*H1+32.*H2+32.*H3)*QPNT(ID,N6))
REL(5) = REL(5) +R*((4.*H1+12.*H2+12.*H3)*QPNT(ID,N1)*(-1)
1=(4.*H1+12.*H2+8.*H3)*QPNT(ID,N2)-(4.*H1+8.*H2+12.*H3)*QPNT(ID,N3)
2+(32.*H1+48.*H2+32.*H3)*QPNT(ID,N4)+(32.*H1+96.*H2+96.*H3)*
3QPNT(ID,N5)+(32.*H1+32.*H2+48.*H3)*QPNT(ID,N6))
REL(6) = REL(6) +R*((12.*H1+4.*H2+8.*H3)*QPNT(ID,N1)
1=(12.*H1+4.*H2+12.*H3)*QPNT(ID,N2)-(8.*H1+4.*H2+12.*H3)*
2QPNT(ID,N3)+(48.*H1+32.*H2+32.*H3)*QPNT(ID,N4)+(32.*H1+32.*H2+48.*
3*H3)*QPNT(ID,N5)+(96.*H1+32.*H2+96.*H3)*QPNT(ID,N6))
C RETURN
C END
C SUBROUTINE SHEAR (M)
C THIS ROUTINE CALCULATES BED SHEAR STRESS AND SHEAR VELOCITY FOR
C A SEDIMENT LADEN FLOW
C REF. HYDRAULICS OF SEDIMENT TRANSPORT BY W.H.GRAF, EQ.8.49
COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)
COMMON /BLK7/NOD(240,6),X(240),Y(240)
COMMON /BLK11/D50(3,100),BD50(100),SR(3,100),SD(3,100)
COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHONAT,
AKP(3)
C USTAR.....SHEAR VELOCITY
C STRESS(J)..BED SHEAR STRESS
C D50(J,H)..MEDIA SIZE DIAMETER OF SEDIMENT J=1,3 M=1,NE
C BD50(J)...TOTAL AVERAGE OF BED SEDIMENT DIAMETER
C RHONAT SPECIFIC WT. OF WATER IN KG(FORCE)/M**3 (1000 KG(F)/M**
C DENSWT...WATER DENSITY IN KG(FORCE)=DAY**2/M**4
C AKAPPA....KARMAN CONSTANT
C
AKAPPA=0.4
N1=NOD(M,1)
N2=NOD(M,2)
N3=NOD(M,3)
TV1=SQRT(VX(N1)**2+VY(N1)**2)
TV2=SQRT(VX(N2)**2+VY(N2)**2)
TV3=SQRT(VX(N3)**2+VY(N3)**2)
USTAR1= TV1 /((17.66+(ALOG10(H(N1)/(96.5*BD50(M)))))*2.3/AKAPPA)
USTAR2= TV2 /((17.66+(ALOG10(H(N2)/(96.5*BD50(M)))))*2.3/AKAPPA)
USTAR3= TV3 /((17.66+(ALOG10(H(N3)/(96.5*BD50(M)))))*2.3/AKAPPA)
DENSWT=RHONAT/(9.8*(3600.*24.))**2)
STRESS(1)=DENSWT*USTAR1**2
STRESS(2)=DENSWT*USTAR2**2
STRESS(3)=DENSWT*USTAR3**2
RETURN
END
SUBROUTINE SILT (M, ID, T)
C THIS SUBROUTINE COMPUTES THE AMOUNT OF RESUSPENSION, SR, OR
C DIPOSITION, SD, OF SILT
C FOR SYMBOLS, SEE SUBROUTINE CLAY
COMMON /BLK1/CT(240),C(9,240)
COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)
COMMON /BLK7/NOD(240,6),X(240),Y(240)
COMMON /BLK11/D50(3,100),BD50(100),SR(3,100),SD(3,100)
COMMON /BLK13/WS(3,100),CRSTRS(3,100),CDSTRS(3,100),ERODA(3,100)
COMMON /BLK14/GBA(100,10),GRH(100,10),GBC(100,10),GBD(100,10),
GBE(100,10),GAF(100,10),GBG(100,10),PCR
C COMMON /BLK15/ILAYR(100,3),XYSO(100),BDIV(100),NBED(100),BED(100),
C RHQSED(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)
C DIMENSION SI(3)
SILT 2
SILT 3
SILT 4
SILT 5
SILT 6
SILT 7
SILT 8
SILT 9
SILT 10
SILT 11
SILT 12
SILT 13
SILT 14
SILT 15
SILT 16

```

	DATA EPSI/1.0E-10/	SILT	17
	SD(ID,M)=0.0	SILT	18
	SR(ID,M)=0.0	SILT	19
	IF (ID,EQ,2) RSAV2(M)=0.	SILT	20
	IF (ID,EQ,3) RSAV3(M)=0.0	SILT	21
	RS=0.0	SILT	22
	ILAYR(M, ID)=0	SILT	23
	TOTAL = GBA(M,NBED(M))/RHOSED(1)+GBB(M,NBED(M))/RHOSED(2)+	SILT	24
	GBC(M,NBED(M))/RHOSED(3)	SILT	25
	XDTOP=(1.-POR)/TOTAL	SILT	26
	IF (ID,EQ,2) XNT(M, ID)=XYS0(M)*GBB(M,NBED(M))*XDTOP	SILT	27
	IF (ID,EQ,3) XNT(M, ID)=XYS0(M)*GBC(M,NBED(M))*XDTOP	SILT	28
	DO 999 II=1,3	SILT	29
	S1(II)=0.0	SILT	30
	IF (STRESS(II).LE.CRSTRS(ID,M).AND.STRESS(II).GE.CDSTRS(ID,M)) GO	SILT	31
	1 TO 999	SILT	32
	IF (STRESS(II).GT.CRSTRS(ID,M)) GO TO 100	SILT	33
C	DEPOSITION	SILT	34
	S1(II)=WS(ID,M)*C(ID,NOD(M,II))*(1.0-(STRESS(II)/CDSTRS(ID,M))	SILT	35
	1)/H(NOD(M,II))*(-1)	SILT	36
	GO TO 999	SILT	37
C		SILT	38
C	RESUSPENSION	SILT	39
C		SILT	40
100	S1(II)=ERQDA(ID,M)*(STRESS(II)/CRSTRS(ID,M)-1.0)	SILT	41
999	CONTINUE	SILT	42
	S2=(S1(1)+S1(2)+S1(3))/3.0	SILT	43
	IF (S2) 300,305,302	SILT	44
300	CONTINUE	SILT	45
C		SILT	46
C		SILT	47
C	SEDIMENT DEPOSITION	SILT	48
C		SILT	49
	SD(ID,M)=(-1)*S2	SILT	50
	N1 = NOD(M,1)	SILT	51
	N2 = NOD(M,2)	SILT	52
	N3 = NOD(M,3)	SILT	53
	N4 = NOD(M,4)	SILT	54
	N5 = NOD(M,5)	SILT	55
	N6 = NOD(M,6)	SILT	56
	H1 = H(N1)	SILT	57
	H2 = H(N2)	SILT	58
	H3 = H(N3)	SILT	59
	HAVG = (H1+H2+H3)/3.0	SILT	60
	CAVG2 = (C(ID,N1)*H1+C(ID,N2)*H2+C(ID,N3)*H3+C(ID,N4)*(H1+H2)/	SILT	61
1	2.0+C(ID,N5)*(H2+H3)/2.0+C(ID,N6)*(H1+H3)/2.0)/(6.0*	SILT	62
2	HAVG)	SILT	63
	IF(CAVG2.LE.EPSI) GO TO 305	SILT	64
	CAVG6 = (C(ID+4,N1)*H1+C(ID+4,N2)*H2+C(ID+4,N3)*H3	SILT	65
1	+C(ID+4,N4)*(H1+H2)/2.0+C(ID+4,N5)*(H2+H3)/2.0	SILT	66
2	+C(ID+4,N6)*(H1+H3)/2.0)/(6.0*HAVG)	SILT	67
	IF (ID,EQ,2) RSAV2(M) =-SD(ID,M)*CAVG6/CAVG2	SILT	68
	IF (ID,EQ,3) RSAV3(M) =-SD(ID,M)*CAVG6/CAVG2	SILT	69
	ILAYR(M, ID)=1	SILT	70
	GO TO 305	SILT	71
302	CONTINUE	SILT	72
C		SILT	73
C	SEDIMENT EROSION	SILT	74
C		SILT	75
	SR(ID,M)=S2	SILT	76
C		SILT	77
C	TO COMPUTE THE AVAILABILITY OF SILT IN BED LAYERS	SILT	78

C	NUMBER OF LAYERS SCOURED IS DETERMINED BY OTHER SEDIMENT (CLAY	SILT	79
C	OR SAND2)	SILT	80
C		SILT	81
	NB=NBED(M)	SILT	82
	SR(ID,M)=SR(ID,M)*T	SILT	83
	IF (SR(ID,M).GT.XNT(M, ID).AND, ILAYR(M,1).GT.0) GO TO 200	SILT	84
	SR(ID,M)=AMIN1(SR(ID,M), XNT(M, ID))	SILT	85
	XNT(M, ID)=XNT(M, ID)-SR(ID,M)	SILT	86
	IF (ID.EQ.2) RSAV2(M)=SR(ID,M)*GBE(M, NBED(M))	SILT	87
	IF (ID.EQ.3) RSAV3(M)=SR(ID,M)*GBF(M, NBED(M))	SILT	88
	GO TO 290	SILT	89
200	ILAYR(M, ID)=ILAYR(M, ID)+1	SILT	90
	IF (ID.EQ.2) RSAV2(M)=RSAV2(M)+XNT(M, ID)*GBE(M, NB)	SILT	91
	IF (ID.EQ.3) RSAV3(M)=RSAV3(M)+XNT(M, ID)*GBF(M, NB)	SILT	92
	SR(ID,M)=SR(ID,M)-XNT(M, ID)	SILT	93
	RS=RS+XNT(M, ID)	SILT	94
	NB=NBED(M)-ILAYR(M, ID)	SILT	95
	IF (NB.LE.0) GO TO 285	SILT	96
	TOTAL=GBA(M, NB)/RH0SED(1)+GBB(M, NB)/RH0SED(2)+GBC(M, NB)/RH0SED(3)	SILT	97
	XND=(1.-POR)/TOTAL	SILT	98
	IF (ID.EQ.2) XNT(M, ID)=BDIV(M)*GBB(M, NB)*XND	SILT	99
	IF (ID.EQ.3) XNT(M, ID)=BDIV(M)*GBC(M, NB)*XND	SILT	100
	IF (ILAYR(M, ID) .GE. ILAYR(M, 1)) GO TO 280	SILT	101
	IF (SR(ID,M).GE.XNT(M, ID)) GO TO 200	SILT	102
280	RS=RS+AMIN1(SR(ID,M), XNT(M, ID))	SILT	103
	IF (ID.EQ.2) RSAV2(M)=RSAV2(M)+AMIN1(SR(ID,M), XNT(M, ID))*GBE(M, NB)	SILT	104
	IF (ID.EQ.3) RSAV3(M)=RSAV3(M)+AMIN1(SR(ID,M), XNT(M, ID))*GBF(M, NB)	SILT	105
285	XNT(M, ID)=XNT(M, ID)-AMIN1(SR(ID,M), XNT(M, ID))	SILT	106
	SR(ID,M)=RS	SILT	107
290	SR(ID,M)=SR(ID,M)/T	SILT	108
	IF (ID.EQ.2) RSAV2(M)=RSAV2(M)/T	SILT	109
	IF (ID.EQ.3) RSAV3(M)=RSAV3(M)/T	SILT	110
305	CONTINUE	SILT	111
	RETURN	SILT	112
	END	SILT	113
	SUBROUTINE SMATRX (M, AREA, INO)	SMATRX	2
C		SMATRX	3
C	THIS SUBROUTINE CONSTRUCTS CONVECTION, DIFFUSION, AND DECAY	SMATRX	4
C	TERMS OF THE MATRIX (SEL) FOR THE M-TH TRIANGULAR ELEMENT	SMATRX	5
C		SMATRX	6
C**	THIS IS THE 3-NODE VERSION OF SMATRX.	SMATRX	7
C**	WITH LINEAR INTERPOLATING POLYNOMIALS.	SMATRX	8
C		SMATRX	9
	COMMON /BLK6/VX(240), VY(240), H(240), STRESS(3)	SMATRX	10
	COMMON /BLK7/NOD(240, 6), X(240), Y(240)	SMATRX	11
	COMMON /BLK9/DX(100), DY(100), EX(100), EY(100), ALFA(100), BETA(100)	SMATRX	12
9	, HS(100)	SMATRX	13
	COMMON /BLK10/PEL(6, 6), SEL(6, 6), REL(6)	SMATRX	14
	COMMON /BLK12/ACOF(6), U(6), V(6), D(2), AKJ(9, 100), ALMBDA, RHO*AT,	SMATRX	15
1	AKP(3)	SMATRX	16
C		SMATRX	17
C	DIMENSION SSEL(6, 6)	SMATRX	18
		SMATRX	19
	A1=ACOF(1)	SMATRX	20
	A2=ACOF(2)	SMATRX	21
	A3=ACOF(3)	SMATRX	22
	B1=ACOF(4)	SMATRX	23
	B2=ACOF(5)	SMATRX	24
	B3=ACOF(6)	SMATRX	25
C		SMATRX	26
	N1=NOD(M, 1)	SMATRX	27
	N2=NOD(M, 2)	SMATRX	28

	N3=100(M,3)	SMATRX	29
C	D1=DX(M)/(60.0*AREA)	SMATRX	30
	D2=DY(M)/(60.0*AREA)	SMATRX	31
	U1=VX(N1)	SMATRX	32
	U2=VX(N2)	SMATRX	33
	U3=VX(N3)	SMATRX	34
	V1=VY(N1)	SMATRX	35
	V2=VY(N2)	SMATRX	36
	V3=VY(N3)	SMATRX	37
	RATE=ALFA(M)	SMATRX	38
C		SMATRX	39
	H1 = H(N1)	SMATRX	40
	H2 = H(N2)	SMATRX	41
	H3 = H(N3)	SMATRX	42
C**	DIFFUSION TERMS	SMATRX	43
	SEL(1,1) = (D1*B1*B1+D2*A1*A1)*(9.*H1+3.*H2+3.*H3)	SMATRX	44
	SEL(1,2) = -(D1*B1*B2+D2*A1*A2)*(2.*H1+2.*H2+H3)	SMATRX	45
	SEL(1,3) = -(D1*B1*B3+D2*A1*A3)*(2.*H1+H2+2.*H3)	SMATRX	46
	SEL(1,4) = (D1*B1*B1+D2*A1*A1)*(3.*H1-2.*H2-H3)	SMATRX	47
1	+(D1*B1*B2+D2*A1*A2)*(14.*H1+3.*H2+3.*H3)	SMATRX	48
	SEL(1,5) = (D1*B1*B2+D2*A1*A2)*(3.*H1-H2-2.*H3)	SMATRX	49
1	+(D1*B1*B3+D2*A1*A3)*(3.*H1-2.*H2-H3)	SMATRX	50
	SEL(1,6) = (D1*B1*B1+D2*A1*A1)*(3.*H1-H2-2.*H3)	SMATRX	51
1	+(D1*B1*B3+D2*A1*A3)*(14.*H1+3.*H2+3.*H3)	SMATRX	52
	SEL(2,2) = (D1*B2*B2+D2*A2*A2)*(3.*H1+9.*H2+3.*H3)	SMATRX	53
	SEL(2,3) = -(D1*B2*B3+D2*A2*A3)*(H1+2.*H2+2.*H3)	SMATRX	54
	SEL(2,4) = (D1*B1*B2+D1*A1*A2)*(3.*H1+14.*H2+3.*H3)	SMATRX	55
1	-(D1*B2*B2+D2*A2*A2)*(2.*H1-3.*H2+H3)	SMATRX	56
	SEL(2,5) = (D1*B2*B3+D2*A2*A3)*(3.*H1+14.*H2+3.*H3)	SMATRX	57
1	-(D1*B2*B2+D2*A2*A2)*(H1-3.*H2+2.*H3)	SMATRX	58
	SEL(2,6) = (D1*B1*B2+D2*A1*A2)*(H1-3.*H2+2.*H3)	SMATRX	59
1	-(D1*B2*B3+D2*A2*A3)*(H1-3.*H2+2.*H3)	SMATRX	60
	SEL(3,3) = (D1*B3*B3+D2*A3*A3)*(3.*H1+3.*H2+9.*H3)	SMATRX	61
	SEL(3,4) = -(D1*B1*B3+D2*A1*A3)*(H1+2.*H2-3.*H3)	SMATRX	62
1	-(D1*B2*B3+D2*A2*A3)*(2.*H1+H2-3.*H3)	SMATRX	63
	SEL(3,5) = (D1*B2*B3+D2*A2*A3)*(3.*H1+3.*H2+14.*H3)	SMATRX	64
1	-(D1*B3*B3+D2*A3*A3)*(H1+2.*H2-3.*H3)	SMATRX	65
	SEL(3,6) = (D1*B1*B3+D2*A1*A3)*(3.*H1+3.*H2+14.*H3)	SMATRX	66
1	-(D1*B3*B3+D2*A3*A3)*(2.*H1+H2-3.*H3)	SMATRX	67
	SEL(4,4) = (D1*B1*B1+D2*A1*A1)*(8.*H1+24.*H2+8.*H3)	SMATRX	68
1	+(D1*B1*B2+D2*A1*A2)*(16.*H1+16.*H2+3.*H3)	SMATRX	69
2	+(D1*B2*B2+D2*A2*A2)*(24.*H1+8.*H2+8.*H3)	SMATRX	70
	SEL(4,5) = (D1*B1*B2+D2*A1*A2)*(4.*H1+8.*H2+8.*H3)	SMATRX	71
1	+(D1*B1*B3+D2*A1*A3)*(8.*H1+24.*H2+8.*H3)	SMATRX	72
2	+(D1*B2*B2+D2*A2*A2)*(8.*H1+4.*H2+8.*H3)	SMATRX	73
3	+(D1*B2*B3+D2*A2*A3)*(8.*H1+8.*H2+4.*H3)	SMATRX	74
	SEL(4,6) = (D1*B1*B1+D2*A1*A1)*(4.*H1+8.*H2+8.*H3)	SMATRX	75
1	+(D1*B1*B2+D2*A1*A2)*(8.*H1+4.*H2+8.*H3)	SMATRX	76
2	+(D1*B1*B3+D2*A1*A3)*(8.*H1+8.*H2+4.*H3)	SMATRX	77
3	+(D1*B2*B3+D2*A2*A3)*(24.*H1+8.*H2+8.*H3)	SMATRX	78
	SEL(5,5) = (D1*B2*B2+D2*A2*A2)*(8.*H1+8.*H2+24.*H3)	SMATRX	79
1	+(D1*B2*B3+D2*A2*A3)*(8.*H1+16.*H2+16.*H3)	SMATRX	80
2	+(D1*B3*B3+D2*A3*A3)*(8.*H1+24.*H2+8.*H3)	SMATRX	81
	SEL(5,6) = (D1*B1*B2+D2*A1*A2)*(8.*H1+8.*H2+24.*H3)	SMATRX	82
1	+(D1*B1*B3+D2*A1*A3)*(4.*H1+8.*H2+8.*H3)	SMATRX	83
2	+(D1*B2*B3+D2*A2*A3)*(8.*H1+4.*H2+8.*H3)	SMATRX	84
3	+(D1*B3*B3+D2*A3*A3)*(8.*H1+8.*H2+4.*H3)	SMATRX	85
	SEL(6,6) = (D1*B1*B1+D2*A1*A1)*(8.*H1+8.*H2+24.*H3)	SMATRX	86
1	+(D1*B1*B3+D2*A1*A3)*(16.*H1+8.*H2+16.*H3)	SMATRX	87
2	+(D1*B3*B3+D2*A3*A3)*(24.*H1+8.*H2+8.*H3)	SMATRX	88
C		SMATRX	89
		SMATRX	90


```

DO 100 I=1,5
II      = I+1
DO 100 J=II,6
SEL(J,I) = SEL(I,J)
CONTINUE
100
C
C
C
C**      ADEVECTION TERMS
C
BU11     =(B1*U1+A1*V1)/2520.
BU12     =(B1*U2+A1*V2)/2520.
BU13     =(B1*U3+A1*V3)/2520.
BU21     =(B2*U1+A2*V1)/2520.
BU22     =(B2*U2+A2*V2)/2520.
BU23     =(B2*U3+A2*V3)/2520.
BU31     =(B3*U1+A3*V1)/2520.
BU32     =(B3*U2+A3*V2)/2520.
BU33     =(B3*U3+A3*V3)/2520.
C
SSEL(1,1) =
1 +BU12*(12.*H1+6.*H2+3.*H3)
SSEL(2,1) =
1 -BU12*(8.*H1+30.*H2+4.*H3)
1 -BU12*(4.*H1-2.*H2+5.*H3)
SSEL(4,1) =
1 +BU12*(40.*H1+12.*H2+4.*H3)
SSEL(5,1) =
1 +BU12*(4.*H1-36.*H2-24.*H3)
SSEL(6,1) =
1 +BU12*(20.*H1+4.*H2+4.*H3)
SSEL(1,2) =
1 -BU22*(8.*H1+22.*H2+5.*H3)
SSEL(2,2) =
1 +BU22*(12.*H1+102.*H2+12.*H3)
SSEL(3,2) =
1 -BU22*(5.*H1+22.*H2+8.*H3)
SSEL(4,2) =
1 +BU22*(40.*H1+108.*H2+20.*H3)
SSEL(5,2) =
1 +BU22*(20.*H1+108.*H2+40.*H3)
SSEL(6,2) =
1 +BU22*(4.*H1+20.*H2+4.*H3)
SSEL(1,3) =
1 -BU32*(5.*H1-2.*H2+4.*H3)
SSEL(2,3) =
1 -BU32*(4.*H1+30.*H2+8.*H3)
SSEL(3,3) =
1 +BU32*(3.*H1+6.*H2+12.*H3)
SSEL(4,3) =
1 -BU32*(24.*H1+36.*H2-4.*H3)
SSEL(5,3) =
1 +BU32*(4.*H1+12.*H2+40.*H3)
SSEL(6,3) =
1 +BU32*(4.*H1+4.*H2+20.*H3)
SSEL(1,4) =
1 +BU21*(144.*H1+12.*H2+12.*H3)
2 +BU22*(12.*H1-8.*H2-4.*H3)
3 +BU23*(12.*H1-4.*H2-8.*H3)
SSEL(2,4) =
1 -BU21*(36.*H1+8.*H2+12.*H3)
+BU11*(102.*H1+12.*H2+12.*H3)
+BU13*(12.*H1+6.*H2+3.*H3)
-BU11*(22.*H1+8.*H2+5.*H3)
-BU13*(5.*H1+4.*H2-2.*H3)
-BU11*(22.*H1+5.*H2+8.*H3)
-BU13*(8.*H1+4.*H2+30.*H3)
+BU11*(108.*H1+40.*H2+20.*H3)
+BU13*(20.*H1+4.*H2+4.*H3)
+BU11*(20.*H1+4.*H2+4.*H3)
+BU13*(4.*H1-36.*H2-24.*H3)
+BU11*(108.*H1+20.*H2+40.*H3)
+BU13*(40.*H1+4.*H2+12.*H3)
-BU21*(30.*H1+8.*H2+4.*H3)
-BU23*(4.*H1+5.*H2-2.*H3)
+BU21*(6.*H1+12.*H2+3.*H3)
+BU23*(3.*H1+12.*H2+6.*H3)
+BU21*(2.*H1-5.*H2-4.*H3)
+BU23*(4.*H1+8.*H2+30.*H3)
+BU21*(12.*H1+40.*H2+4.*H3)
+BU23*(4.*H1+20.*H2+4.*H3)
+BU21*(4.*H1+40.*H2+12.*H3)
+BU23*(36.*H1-4.*H2+24.*H3)
+BU23*(24.*H1-4.*H2+36.*H3)
-BU31*(30.*H1+4.*H2+8.*H3)
-BU33*(8.*H1+5.*H2+22.*H3)
+BU31*(2.*H1-4.*H2-5.*H3)
+BU33*(5.*H1+8.*H2+22.*H3)
+BU31*(6.*H1+3.*H2+12.*H3)
+BU33*(12.*H1+12.*H2+102.*H3)
+BU31*(36.*H1+24.*H2-4.*H3)
+BU33*(4.*H1+4.*H2+20.*H3)
+BU31*(4.*H1+4.*H2+20.*H3)
+BU33*(20.*H1+40.*H2+108.*H3)
+BU31*(12.*H1+4.*H2+40.*H3)
+BU33*(40.*H1+20.*H2+108.*H3)
+BU11*(12.*H1-8.*H2-4.*H3)
-BU12*(8.*H1+36.*H2+12.*H3)
-BU13*(4.*H1+12.*H2+12.*H3)
-BU11*(8.*H1-12.*H2+4.*H3)
+BU12*(12.*H1+144.*H2+12.*H3)
SMATRX 91
SMATRX 92
SMATRX 93
SMATRX 94
SMATRX 95
SMATRX 96
SMATRX 97
SMATRX 98
SMATRX 99
SMATRX 100
SMATRX 101
SMATRX 102
SMATRX 103
SMATRX 104
SMATRX 105
SMATRX 106
SMATRX 107
SMATRX 108
SMATRX 109
SMATRX 110
SMATRX 111
SMATRX 112
SMATRX 113
SMATRX 114
SMATRX 115
SMATRX 116
SMATRX 117
SMATRX 118
SMATRX 119
SMATRX 120
SMATRX 121
SMATRX 122
SMATRX 123
SMATRX 124
SMATRX 125
SMATRX 126
SMATRX 127
SMATRX 128
SMATRX 129
SMATRX 130
SMATRX 131
SMATRX 132
SMATRX 133
SMATRX 134
SMATRX 135
SMATRX 136
SMATRX 137
SMATRX 138
SMATRX 139
SMATRX 140
SMATRX 141
SMATRX 142
SMATRX 143
SMATRX 144
SMATRX 145
SMATRX 146
SMATRX 147
SMATRX 148
SMATRX 149
SMATRX 150
SMATRX 151
SMATRX 152

```


2	-BU22*(8,*H1-12,*H2+4,*H3)	-BU13*(4,*H1-12,*H2+8,*H3)	SMATRX	153
3	-BU23*(12,*H1+4,*H2+12,*H3)		SMATRX	154
	SSEL(3,4) =		SMATRX	155
1	+BU21*(36,*H1+12,*H2+8,*H3)	+BU11*(12,*H1+12,*H2+4,*H3)	SMATRX	156
2	+BU22*(12,*H1+12,*H2+4,*H3)	+BU12*(12,*H1+36,*H2+8,*H3)	SMATRX	157
3	+BU23*(8,*H1+4,*H2-12,*H3)	+BU13*(4,*H1+8,*H2-12,*H3)	SMATRX	158
	SSEL(4,4) =		SMATRX	159
1	+BU21*(192,*H1+96,*H2+48,*H3)	+BU11*(96,*H1+96,*H2+32,*H3)	SMATRX	160
2	+BU22*(96,*H1+96,*H2+32,*H3)	+BU12*(96,*H1+192,*H2+48,*H3)	SMATRX	161
3	+BU23*(48,*H1+32,*H2+32,*H3)	+BU13*(32,*H1+48,*H2+32,*H3)	SMATRX	162
	SSEL(5,4) =		SMATRX	163
1	+BU21*(48,*H1+32,*H2+32,*H3)	+BU11*(32,*H1+48,*H2+32,*H3)	SMATRX	164
2	+BU22*(32,*H1+48,*H2+32,*H3)	+BU12*(48,*H1+192,*H2+96,*H3)	SMATRX	165
3	+BU23*(32,*H1+32,*H2+48,*H3)	+BU13*(32,*H1+96,*H2+96,*H3)	SMATRX	166
	SSEL(6,4) =		SMATRX	167
1	+BU21*(192,*H1+48,*H2+96,*H3)	+BU11*(48,*H1+32,*H2+32,*H3)	SMATRX	168
2	+BU22*(48,*H1+32,*H2+32,*H3)	+BU12*(32,*H1+48,*H2+32,*H3)	SMATRX	169
3	+BU23*(96,*H1+32,*H2+96,*H3)	+BU13*(32,*H1+32,*H2+48,*H3)	SMATRX	170
	SSEL(1,5) =		SMATRX	171
1	+BU31*(12,*H1+8,*H2-4,*H3)	+BU21*(12,*H1-4,*H2-8,*H3)	SMATRX	172
2	+BU32*(8,*H1+36,*H2+12,*H3)	+BU22*(4,*H1+12,*H2+12,*H3)	SMATRX	173
3	+BU33*(4,*H1+12,*H2+12,*H3)	+BU23*(8,*H1+12,*H2+36,*H3)	SMATRX	174
	SSEL(2,5) =		SMATRX	175
1	+BU31*(8,*H1-12,*H2+4,*H3)	+BU21*(12,*H1+4,*H2+12,*H3)	SMATRX	176
2	+BU32*(12,*H1+144,*H2+12,*H3)	+BU22*(4,*H1-12,*H2+8,*H3)	SMATRX	177
3	+BU33*(4,*H1-12,*H2+8,*H3)	+BU23*(12,*H1+8,*H2+36,*H3)	SMATRX	178
	SSEL(3,5) =		SMATRX	179
1	+BU31*(12,*H1+12,*H2+4,*H3)	+BU21*(8,*H1+4,*H2-12,*H3)	SMATRX	180
2	+BU32*(12,*H1+36,*H2+8,*H3)	+BU22*(4,*H1+8,*H2-12,*H3)	SMATRX	181
3	+BU33*(4,*H1+8,*H2-12,*H3)	+BU23*(12,*H1+12,*H2+144,*H3)	SMATRX	182
	SSEL(4,5) =		SMATRX	183
1	+BU31*(96,*H1+96,*H2+32,*H3)	+BU21*(48,*H1+32,*H2+32,*H3)	SMATRX	184
2	+BU32*(96,*H1+192,*H2+48,*H3)	+BU22*(32,*H1+48,*H2+32,*H3)	SMATRX	185
3	+BU33*(32,*H1+48,*H2+32,*H3)	+BU23*(32,*H1+32,*H2+48,*H3)	SMATRX	186
	SSEL(5,5) =		SMATRX	187
1	+BU31*(32,*H1+48,*H2+32,*H3)	+BU21*(32,*H1+32,*H2+48,*H3)	SMATRX	188
2	+BU32*(48,*H1+192,*H2+96,*H3)	+BU22*(32,*H1+96,*H2+96,*H3)	SMATRX	189
3	+BU33*(32,*H1+96,*H2+96,*H3)	+BU23*(48,*H1+96,*H2+192,*H3)	SMATRX	190
	SSEL(6,5) =		SMATRX	191
1	+BU31*(48,*H1+32,*H2+32,*H3)	+BU21*(96,*H1+32,*H2+96,*H3)	SMATRX	192
2	+BU32*(32,*H1+48,*H2+32,*H3)	+BU22*(32,*H1+32,*H2+48,*H3)	SMATRX	193
3	+BU33*(32,*H1+32,*H2+48,*H3)	+BU23*(96,*H1+48,*H2+192,*H3)	SMATRX	194
	SSEL(1,6) =		SMATRX	195
1	+BU31*(144,*H1+12,*H2+12,*H3)	+BU11*(12,*H1-4,*H2-8,*H3)	SMATRX	196
2	+BU32*(4,*H1+8,*H2-12,*H3)	+BU12*(12,*H1+12,*H2+4,*H3)	SMATRX	197
3	+BU33*(12,*H1-4,*H2-8,*H3)	+BU13*(8,*H1+12,*H2+36,*H3)	SMATRX	198
	SSEL(2,6) =		SMATRX	199
1	+BU31*(36,*H1+8,*H2+12,*H3)	+BU11*(12,*H1+4,*H2+12,*H3)	SMATRX	200
2	+BU32*(8,*H1-12,*H2+4,*H3)	+BU12*(4,*H1-12,*H2+8,*H3)	SMATRX	201
3	+BU33*(12,*H1+4,*H2+12,*H3)	+BU13*(12,*H1+8,*H2+36,*H3)	SMATRX	202
	SSEL(3,6) =		SMATRX	203
1	+BU31*(36,*H1+12,*H2+8,*H3)	+BU11*(8,*H1+4,*H2-12,*H3)	SMATRX	204
2	+BU32*(12,*H1+12,*H2+4,*H3)	+BU12*(4,*H1+8,*H2-12,*H3)	SMATRX	205
3	+BU33*(8,*H1+4,*H2-12,*H3)	+BU13*(12,*H1+12,*H2+144,*H3)	SMATRX	206
	SSEL(4,6) =		SMATRX	207
1	+BU31*(192,*H1+96,*H2+48,*H3)	+BU11*(48,*H1+32,*H2+32,*H3)	SMATRX	208
2	+BU32*(96,*H1+96,*H2+32,*H3)	+BU12*(96,*H1+48,*H2+32,*H3)	SMATRX	209
3	+BU33*(48,*H1+32,*H2+32,*H3)	+BU13*(32,*H1+32,*H2+48,*H3)	SMATRX	210
	SSEL(5,6) =		SMATRX	211
1	+BU31*(48,*H1+32,*H2+32,*H3)	+BU11*(32,*H1+32,*H2+48,*H3)	SMATRX	212
2	+BU32*(32,*H1+48,*H2+32,*H3)	+BU12*(32,*H1+96,*H2+96,*H3)	SMATRX	213
3	+BU33*(32,*H1+32,*H2+48,*H3)	+BU13*(48,*H1+96,*H2+192,*H3)	SMATRX	214

	SSEL(6,6) =	+BU11*(96.*H1+32.*H2+96.*H3)	SMATRX	215
1	+BU31*(192.*H1+48.*H2+96.*H3)	+BU12*(32.*H1+32.*H2+48.*H3)	SMATRX	216
2	+BU32*(48.*H1+32.*H2+32.*H3)	+BU13*(96.*H1+48.*H2+192.*H3)	SMATRX	217
3	+BU33*(96.*H1+32.*H2+96.*H3)		SMATRX	218
C			SMATRX	219
C**	DECAY TERMS		SMATRX	220
C			SMATRX	221
C	A = AREA/1260.		SMATRX	222
C			SMATRX	223
	DO 130 J=1,6		SMATRX	224
	DO 130 I=1,6		SMATRX	225
	SEL(I,J) = SEL(I,J)+SSEL(I,J)+RATE*PEL(I,J)		SMATRX	226
130	CONTINUE		SMATRX	227
	DO 140 J=1,6		FETFIX2	1
	DO 140 I=1,6		FETFIX2	2
	IF (I.EQ.J) GO TO 140		FETFIX2	3
	IF (SSEL(I,J).LE.0) GO TO 140		FETFIX2	4
	SEL(I,J) = SEL(I,J) - SSEL(I,J)		FETFIX2	5
	SEL(J,I) = SEL(J,I) - SSEL(I,J)		FETFIX2	6
	SEL(I,I) = SEL(I,I) + SSEL(I,J)		FETFIX2	7
	SEL(J,J) = SEL(J,J) + SSEL(I,J)		FETFIX2	8
140	CONTINUE		FETFIX2	9
	RETURN		SMATRX	228
	END		SMATRX	229
	FUNCTION SOL(H,G,W,TOL)		SOL	2
C			SOL	3
C**	THIS FUNCTION SUBPROGRAM SOLVES THE TRANSCENDENTAL EQ. FOR K.		SOL	4
C**	K=***2/(G*TANH(K*H))		SOL	5
C			SOL	6
	C=G/***2		SOL	7
C			SOL	8
C**	MAKE INITIAL ESTIMATE OF K USING K=W/CS		SOL	9
C**	WHERE CS IS THE SHALLOW WATER WAVE CELERITY.		SOL	10
C			SOL	11
	DUMMY=W/SQRT(G*H)		SOL	12
1	W=DUMMY		SOL	13
	DUMMY=1./(C*TANH(W*H))		SOL	14
	DIFF=ABS(W-DUMMY)		SOL	15
	IF(DIFF.GT.TOL) GO TO 1		SOL	16
	SOL=DUMMY		SOL	17
	RETURN		SOL	18
	END		SOL	19
	SUBROUTINE SPECS(NE,NDS,NCOL,NTP,T,ID,IDN,INO,KK,		SPECS	2
1	NPRNT,RSTRT,STORE,XXX,R,T)		SPECS	3
C			SPECS	4
C	THIS ROUTINE READS, CHECKS AND PRINTS THE PROBLEM SPECIFICATION.		SPECS	5
C			SPECS	6
C	THE UNITS OF THE INPUT DATA...METER,KILOGRAM(FORCE),DAY,PICOCURIE		SPECS	7
C	THE UNIT OF SEDIMENT CONC....(KG/CUBIC METER)		SPECS	8
C	THE UNIT OF DISSOLVED CHEMICAL....(KG OR PCI/CUBIC METER OF WATER)		SPECS	9
C	THE UNIT OF CHEMICAL ATTACHED TO SED....(KG OR PCI/CUBIC METER OF WA		SPECS	10
C	THE OUTPUT UNIT OF CHEMICAL ATTACHED TO SEDIMENT(KG OF PCI/KG OF		SPECS	11
C	SEDIMENT)		SPECS	12
C			SPECS	13
C			SPECS	14
C	CHARACTER*5 NAME,JCHECK,ZTYP,FLAG		FETFIX1	16
C	CHARACTER*10 SCHECK		FETFIX1	17
C	REAL K		SPECS	17
C			SPECS	18
	COMMON /BLK1/CT(240),C(9,240)		SPECS	19
	COMMON /BLK4/ R(240), RPAST(240,7), NODBET, BETA1, AREA1		SPECS	20
	LEVEL 2,R,RPAST,NODBET,BETA1,AREA1		FETFIX4	22

	COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)	SPECS	21
	COMMON /BLK7/NOO(240,6),X(240),Y(240)	SPECS	22
	COMMON /BLK8/KBC,LBC,MBC(7,120),NBC(7,120),OBC(7,120),BC(7,120),	SPECS	23
	1 KODE(10)	SPECS	24
	COMMON /BLK9/DX(100),DY(100),EX(100),EY(100),ALFA(100),BETA(100)	SPECS	25
	9 ,HS(100)	SPECS	26
	COMMON /BLK11/DSO(3,100),BDSO(100),SR(3,100),SD(3,100)	SPECS	27
	COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHOMAT,	SPECS	28
	1 AKP(3)	SPECS	29
	COMMON /BLK13/MS(3,100),CRSTRS(3,100),CDSTRS(3,100),ERODA(3,100)	SPECS	30
	COMMON /BLK14/G8A(100,10),G8B(100,10),G8C(100,10),G8D(100,10),	SPECS	31
	1 G8E(100,10),G8F(100,10),G8G(100,10),POR	SPECS	32
	COMMON /BLK15/ILAYR(100,3),XYSO(100),HOIV(100),NBED(100),HED(100),	SPECS	33
	1 RHOSD(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)	SPECS	34
	COMMON /BLK16/ STYP	SPECS	35
	COMMON /BLK17/QLATE(240),QSARA(7,100),QPNT(7,240),CRATE(3,100),	SPECS	36
	1 CD(4,240)	SPECS	37
	LEVEL 2,QLATE,QSARA,QPNT,CRATE,CD	FETFIX4	23
C		SPECS	38
C		SPECS	39
	COMMON/WAVE/ NODE(240), NC(135), A(135,10), K(135,10)	SPECS	40
	3 , W(135,10), HB(135), ALPHAC(135), WAVE, NNN, DD	SPECS	41
	3 , VIS, DPHS(135)	SPECS	42
	COMMON /ZTYPE/ ZTYP(135)	SPECS	43
	COMMON/OPTION/ CLAY, SAND	SPECS	44
C		SPECS	45
	LOGICAL NUICS, NUOOF, NUVEL, TVBCS, TVHYD, TVINP, RSTRT, STORE,	SPECS	46
	3 USSDI, NSSDI, UBSDI, NBSDI, URGEO, NBGEO, UBCON, NBCON	SPECS	47
	LOGICAL CLAY, SAND, WAVE, WAVR, VELR	SPECS	48
	DIMENSION ARAD(3),ALEFT(3),B2(3),B3(3),GBJ(3),CS(3)	SPECS	49
	DIMENSION NAME(32), FLAG(16)	SPECS	50
	DATA KU/410/,MU/410/,NU/43/	SPECS	51
C		SPECS	52
	IF(KKK.EQ.1) GO TO 100	SPECS	53
10	CONTINUE	SPECS	54
	*WRITE(6,888)KK, ID, NBED(74), RAT, T, C(ID, 188), GBG(74), NBED(74)	SPECS	55
888	FORMAT(2X,3I5,4E12.5)	SPECS	56
	IF(KKK.EQ.1) WRITE(6,2070) ID,INO	SPECS	57
	IF(.NOT.TVINP .AND. KKK.GT.1) RETURN	SPECS	58
	IF(.NOT.TVBCS .AND. KKK.GT.1) GO TO 20	SPECS	59
	GO TO 250	SPECS	60
20	CONTINUE	SPECS	61
	IF(.NOT.TVHYD .AND. KKK.GT.1) RETURN	SPECS	62
	IF(ID .GT. 1) RETURN	SPECS	63
	GO TO 270	SPECS	64
C		SPECS	65
C	CARD 1-2	SPECS	66
C	NAME(J)...TITLE OR COMMENT CARDS(2)	SPECS	67
100	READ (5,310) (NAME(J),J=1,32)	SPECS	68
	*WRITE (6,320)	SPECS	69
	*WRITE (6,320)	SPECS	70
	*WRITE (6,330) (NAME(J),J=1,32)	SPECS	71
C		SPECS	72
C	CARD 3	SPECS	73
C	ID.....IDENTIFICATION OF SIMULATION SUBSTANCES	SPECS	74
C	ID=1,2,3..SEDIMENT	SPECS	75
C	ID=4.....DISOLVED CHEMICAL	SPECS	76
C	ID=5,6,7..CHEMICAL ADSORBED BY SEDIMENT	SPECS	77
C	ID=8.....TOTAL AMOUNT OF SEDIMENT	SPECS	78
C	ID=9.....TOTAL AMOUNT OF CHEMICAL	SPECS	79
C	INO.....FLAG FOR COEFFICIENT SUBROUTINES FOR TRANSPORT CODE (=0)	SPECS	80
C		SPECS	81

	READ (5,325) ID,INO	SPECS	82
	IF (IU.GT.1) GO TO 125	SPECS	83
C		SPECS	84
C	CARD 4-5	SPECS	85
C		SPECS	86
C	IDN.....TOTAL NUMBER OF SIMULATION SUBSTANCES	SPECS	87
C	NE.....TOTAL NUMBER OF TRIANGULAR ELEMENTS	SPECS	88
C	NDS.....TOTAL NUMBER OF NODES	SPECS	89
C	NTP.....TOTAL NUMBER OF TIME PLANES	SPECS	90
C	(#0 FOR STEADY-STATE SOL.)	SPECS	91
C	NPRNT.....PRINT FREQUENCY	SPECS	92
C	T.....SIMULATION TIME STEP	SPECS	93
C**	STYP.....NUMERICAL SOLUTION TYPE	SPECS	94
C**	#0. IMPLIES EXPLICIT SOLUTION	SPECS	95
C**	#.5 IMPLIES CRANK-NICHOLSON SOLUTION(DEFAULT).	SPECS	96
C**	#1. IMPLIES IMPLICIT SOLUTION	SPECS	97
	READ(5,341) IDN,NE,NDS,NTP,NPRNT	SPECS	98
	READ(5,349) T,STYP,SCHECK	SPECS	99
	IF(SCHECK.EQ.#) STYP#.5	SPECS	100
	WRITE (6,520)	SPECS	101
	WRITE (6,550) IDN,NE,NDS,NTP,T,NPRNT	SPECS	102
	WRITE(6,2020) STYP	SPECS	103
	IF(STYP.EQ.0.) WRITE(6,2030)	SPECS	104
	IF(STYP.EQ..5) WRITE(6,2040)	SPECS	105
	IF(STYP.EQ.1.) WRITE(6,2050)	SPECS	106
	IF((STYP.LT.0.) .OR. (STYP.GT.1.)) WRITE(6,2060)	SPECS	107
	IF((STYP.LT.0.) .OR. (STYP.GT.1.)) STOP	SPECS	108
	IF(NTP.EQ.0) NPRNT=1	SPECS	109
	IF (NDS.LT.MU.AND.NE.LT.KU.AND.NDS.GT.6) GO TO 110	SPECS	110
	WRITE (6,370)	SPECS	111
	STOP	SPECS	112
110	CONTINUE	SPECS	113
120	CONTINUE	SPECS	114
	WRITE (6,520)	SPECS	115
C		SPECS	116
C		SPECS	117
C	CARD GROUP 6-7	SPECS	118
C		SPECS	119
C	NUGSA.....NUMBER OF AREA SOURCES	SPECS	120
C	QSARA(I,J).....SOURCE STRENGTH(PCI/CUBIC METER OF DISCHARGE/DAY)	SPECS	121
C	I.....CONSTITUENT NUMBER	SPECS	122
C	J.....SOURCE LOCATION (ELEMENT NUMBER)	SPECS	123
C	NUQSP.....NUMBER OF POINT SOURCES	SPECS	124
C	QPNT(I,J).....SOURCE STRENGTH(PCI/CUBIC METER OF DISCHARGE/DAY)	SPECS	125
C	I.....CONSTITUENT NUMBER	SPECS	126
C	J.....SOURCE LOCATION (NODE NUMBER)	SPECS	127
C		SPECS	128
	DO 700 J=1,NE	SPECS	129
	DO 700 I=1,ION	SPECS	130
	QSARA(I,J)=0.0	SPECS	131
700	CONTINUE	SPECS	132
	DO 701 J=1,NDS	SPECS	133
	DO 701 I=1,ION	SPECS	134
	QPNT(I,J)=0.0	SPECS	135
701	CONTINUE	SPECS	136
C		SPECS	137
C		SPECS	138
	READ(5,509) NUGSA	SPECS	139
	WRITE(6,705) NUGSA	SPECS	140
	IF (NUGSA.EQ.0) GO TO 704	FETFIX1	18
	WRITE(6,706)	SPECS	142
	DO 702 II=1,NUGSA	SPECS	143

	READ(5,435) I,J,GSARA(I,J)	SPECS	144
	WRITE(6,707) I,J,GSARA(I,J)	SPECS	145
702	CONTINUE	SPECS	146
704	CONTINUE	FETFIX1	19
	READ(5,509) NUQSP	SPECS	147
	WRITE(6,708) NUQSP	SPECS	148
	IF (NUQSP.EQ.0) GO TO 710	FETFIX1	20
	WRITE(6,709)	SPECS	150
	DO 703 II=1,NUQSP	SPECS	151
	READ(5,435) I,J,QPNT(I,J)	SPECS	152
	WRITE(6,707) I,J,QPNT(I,J)	SPECS	153
703	CONTINUE	SPECS	154
710	CONTINUE	FETFIX1	21
	WRITE(6,520)	SPECS	155
	WRITE(6,390)	SPECS	156
C		SPECS	157
C	CARD 8	SPECS	158
C	I.....ELEMENT NUMBER	SPECS	159
C	NOD(N,K)..ELEMENT-NODE TABLE (CONNECTIVITY)	SPECS	160
	DO 170 N=1,NE	SPECS	161
	READ (5,560) I,NOD(I,1),NOD(I,4),NOD(I,2),NOD(I,5),NOD(I,3),NOD(I,	SPECS	162
	6)	SPECS	163
	WRITE (6,570) I,(NOD(I,L),L=1,6)	SPECS	164
	IF (I.LT.1.OR.I.GT.NE) WRITE (6,540)	SPECS	165
	IF (I.LT.1.OR.I.GT.NE) STOP	SPECS	166
170	CONTINUE	SPECS	167
C		SPECS	168
C	COMPUTE THE BAND WIDTH (2*NCOL-1)	SPECS	169
C		SPECS	170
	NCOL=1	SPECS	171
	DO 180 N=1,NE	SPECS	172
	DO 180 I=1,6	SPECS	173
	DO 180 J=1,6	SPECS	174
	NN=NOD(N,I)-NOD(N,J)+1	SPECS	175
180	IF (NCOL=NN.LT.0) NCOL=NN	SPECS	176
	WRITE (6,400) NCOL	SPECS	177
	IF (NCOL.LE.NU) GO TO 190	SPECS	178
	WRITE (6,410)	SPECS	179
	STOP	SPECS	180
190	CONTINUE	SPECS	181
	WRITE (6,520)	SPECS	182
	WRITE(6,610)	SPECS	183
C		SPECS	184
C	CARD 9	SPECS	185
C	J.....CORNER NODE NUMBER	SPECS	186
C	X(J).....X-COORDINATE OF J-TH NODE	SPECS	187
C	Y(J).....Y-COORDINATE OF J-TH NODE	SPECS	188
C**	INPUT INFORMATION FOR CORNER NODES ONLY. FETRA INTERPOLATES	SPECS	189
C**	BETWEEN THE CORNER NODES TO FIND THE LOCATION OF MIDPOINTS.	SPECS	190
C**	A BLANK CARD MUST FOLLOW THE LAST NODE DATA CARD.	SPECS	191
C		SPECS	192
	NCNDS = 0	SPECS	193
192	READ(5,500) J, XJ, YJ, JCHECK	SPECS	194
	IF(JCHECK .EQ. " ") GO TO 200	SPECS	195
	X(J) = XJ	SPECS	196
	Y(J) = YJ	SPECS	197
	NCNDS = NCNDS + 1	SPECS	198
	WRITE(6,611) J, X(J), Y(J)	SPECS	199
	GO TO 192	SPECS	200
200	CONTINUE	SPECS	201
C**	CARD 10	SPECS	202
C		SPECS	203

C**	BY SPECIFYING THE FOLLOWING VALUES THE OPTIONS DESCRIBED	SPECS	204
C**	ARE ACTIVATED(SEPARATE THE OPTIONS WITH A COMMA). IF NO OPTIONS	SPECS	205
C**	ARE DESIRED, INPUT A BLANK CARD.	SPECS	206
C		SPECS	207
C**	NUCOF, FOR NONUNIFORM COEFFICIENTS(OX,DY,ALFA,BETA,HS-SEE	SPECS	208
C**	CARDS 15,16,AND 17).	SPECS	209
C**	NUVEL, FOR NONUNIFORM VELOCITY AND/OR DEPTH.	SPECS	210
C**	TVBCS, FOR TIME VARYING BOUNDARY CONDITIONS.	SPECS	211
C**	TVHYD, FOR TIME VARYING HYDRAULICS.	SPECS	212
C**	TVINP, FOR TIME VARYING INPUT DATA.	SPECS	213
C**	USSDI, FOR UNIFORM SUSPENDED SEDIMENT DIAMETER.	SPECS	214
C**	NSSDI, FOR NONUNIFORM SUSPENDED SEDIMENT DIAMETER.	SPECS	215
C**	UBSDI, FOR UNIFORM BED SEDIMENT DIAMETER.	SPECS	216
C**	NBSDI, FOR NONUNIFORM BED SEDIMENT DIAMETER.	SPECS	217
C**	UBGEO, FOR UNIFORM BED CONFIGURATION.	SPECS	218
C**	NBGEO, FOR NONUNIFORM BED CONFIGURATION.	SPECS	219
C**	UBCON, FOR UNIFORM BED CONCENTRATION.	SPECS	220
C**	NBCON, FOR NONUNIFORM BED CONCENTRATION.	SPECS	221
C		SPECS	222
125	CONTINUE	SPECS	223
	NUCOF=,FALSE.	SPECS	224
	NUVEL=,FALSE.	SPECS	225
	TVINP=,FALSE.	SPECS	226
	TVBCS=,FALSE.	SPECS	227
	TVHYD=,FALSE.	SPECS	228
	USSDI=,FALSE.	SPECS	229
	NSSDI=,FALSE.	SPECS	230
	UBSDI=,FALSE.	SPECS	231
	NBSDI=,FALSE.	SPECS	232
	UBGEO=,FALSE.	SPECS	233
	NBGEO=,FALSE.	SPECS	234
	UBCON=,FALSE.	SPECS	235
	NBCON=,FALSE.	SPECS	236
	RSTRT=,FALSE.	SPECS	237
	STORE=,FALSE.	SPECS	238
	READ(5,1025) (FLAG(I),I=1,12)	SPECS	239
	DO 25 I=1,12	SPECS	240
	IF(FLAG(I) .EQ. " ") GO TO 27	SPECS	241
	IF(FLAG(I) .EQ. "NUCOF") NUCOF=,TRUE.	SPECS	242
	IF(FLAG(I) .EQ. "NUVEL") NUVEL=,TRUE.	SPECS	243
	IF(FLAG(I) .EQ. "TVINP") TVINP=,TRUE.	SPECS	244
	IF(FLAG(I) .EQ. "TVBCS") TVBCS=,TRUE.	SPECS	245
	IF(FLAG(I) .EQ. "TVHYD") TVHYD=,TRUE.	SPECS	246
	IF(FLAG(I) .EQ. "USSDI") USSDI=,TRUE.	SPECS	247
	IF(FLAG(I) .EQ. "NSSDI") NSSDI=,TRUE.	SPECS	248
	IF(FLAG(I) .EQ. "UBSDI") UBSDI=,TRUE.	SPECS	249
	IF(FLAG(I) .EQ. "NBSDI") NBSDI=,TRUE.	SPECS	250
	IF(FLAG(I) .EQ. "UBGEO") UBGEO=,TRUE.	SPECS	251
	IF(FLAG(I) .EQ. "NBGEO") NBGEO=,TRUE.	SPECS	252
	IF(FLAG(I) .EQ. "UBCON") UBCON=,TRUE.	SPECS	253
	IF(FLAG(I) .EQ. "NBCON") NBCON=,TRUE.	SPECS	254
	IF(FLAG(I) .EQ. "RSTRT") RSTRT=,TRUE.	SPECS	255
	IF(FLAG(I) .EQ. "STORE") STORE=,TRUE.	SPECS	256
25	CONTINUE	SPECS	257
27	CONTINUE	SPECS	258
	WRITE(6,520)	SPECS	259
	WRITE(6,6000)	SPECS	260
	IF(NUCOF) WRITE(6,6015)	SPECS	261
	IF(NUVEL) WRITE(6,6020)	SPECS	262
	IF(TVBCS) WRITE(6,6025)	SPECS	263
	IF(TVHYD) WRITE(6,6030)	SPECS	264
	IF(TVINP) WRITE(6,6035)	SPECS	265

	IF(USSDI) WRITE(6,6040)	SPECS	266
	IF(NSSDI) WRITE(6,6045)	SPECS	267
	IF(UBSDI) WRITE(6,6050)	SPECS	268
	IF(NBSDI) WRITE(6,6055)	SPECS	269
	IF(UBGEO) WRITE(6,6060)	SPECS	270
	IF(NBGEO) WRITE(6,6065)	SPECS	271
	IF(UBCON) WRITE(6,6070)	SPECS	272
	IF(NBCON) WRITE(6,6075)	SPECS	273
C		SPECS	274
C**	CARD 10A	SPECS	275
C		SPECS	276
C**	BY SPECIFYING THE FOLLOWING VALUES THE OPTIONS DESCRIBED ARE	SPECS	277
C**	ACTIVATED(SEPARATE THE OPTIONS WITH A COMMA), IF NO OPTIONS	SPECS	278
C**	ARE DESIRED INPUT A BLANK CARD.	SPECS	279
C		SPECS	280
C**	CLAY, REQUIRED IF INO(SEE CARD 3) IS GREATER THAN ZERO	SPECS	281
C**	AND CLAY IS THE LIMITING BED LAYER.	SPECS	282
C**	SAND, REQUIRED IF INO(SEE CARD 3) IS GREATER THAN ZERO	SPECS	283
C**	AND SAND IS THE LIMITING BED LAYER.	SPECS	284
C**	HAVE, REQUIRED IF THE CONTAMINANT AND/OR SEDIMENT TRANSPORT	SPECS	285
C**	AT ANY OF THE NODES IS INFLUENCED BY A WAVE OR SURF	SPECS	286
C**	ENVIRONMENT.	SPECS	287
C**	HAVR, REQUIRED IF HAVE, IS SPECIFIED AND THE WAVE AND SURF	SPECS	288
C**	CHARACTERISTICS ARE TO BE READ FROM FILES CREATED BY	SPECS	289
C**	OTHER PROGRAMS.	SPECS	290
C**	VELR, REQUIRED IF THE VELOCITY FIELD IS TO BE READ FROM FILES	SPECS	291
C**	CREATED FROM OTHER PROGRAMS.	SPECS	292
C		SPECS	293
	CLAY=.FALSE.	SPECS	294
	HAVR=.FALSE.	SPECS	295
	SAND=.FALSE.	SPECS	296
	HAVE=.FALSE.	SPECS	297
	VELR=.FALSE.	SPECS	298
	READ(5,1030) (FLAG(I),I=1,16)	SPECS	299
	DO 30 I=1,10	SPECS	300
	IF(FLAG(I) .EQ. " ") GO TO 35	SPECS	301
	IF(FLAG(I) .EQ. "CLAY,") CLAY=.TRUE.	SPECS	302
	IF(FLAG(I) .EQ. "SAND,") SAND=.TRUE.	SPECS	303
	IF(FLAG(I) .EQ. "HAVE,") HAVE=.TRUE.	SPECS	304
	IF(FLAG(I) .EQ. "HAVR,") HAVR=.TRUE.	SPECS	305
	IF(FLAG(I) .EQ. "VELR,") VELR=.TRUE.	SPECS	306
30	CONTINUE	SPECS	307
35	CONTINUE	SPECS	308
	IF (INO.LE.0) GO TO 107	SPECS	309
C		SPECS	310
C	CARD 10-12	SPECS	311
C		SPECS	312
C	D50(J,M)..SUSPENDED SEDIMENT DIAMETER SIZE J=1,2,3	SPECS	313
C	BD50(M)..BED SEDIMENT SIZE	SPECS	314
C	CRSTRS(ID,M).CRITICAL SHEAR STRESS FOR SCOURING, ID=1,2,3	SPECS	315
C	CDSTRS(ID,M).CRITICAL SHEAR STRESS FOR DEPOSITION ID=1,2,3	SPECS	316
C	WS(ID,M).....FALL VELOCITY OF SEDIMENT ID=1,2,3 AT ELEMENT M	SPECS	317
C	ERODA(ID,M)..ERODABILITY COEFFICIENT ID=1,2,3 AT ELEMENT M	SPECS	318
C		SPECS	319
	IF (ID.GT.3) GO TO 107	SPECS	320
	IF(USSDI) GO TO 103	SPECS	321
	IF(NSSDI) GO TO 108	SPECS	322
	GO TO 104	SPECS	323
108	READ (5,503) (D50(ID,M),WS(ID,M),CRSTRS(ID,M),CDSTRS(ID,M),	SPECS	324
	ERODA(ID,M),M=1,NE)	SPECS	325
	WRITE(6,352)	SPECS	326
	WRITE(6,353) (ID,M,D50(ID,M),WS(ID,M),CRSTRS(ID,M),CDSTRS(ID,M),	SPECS	327

	1ERODA(ID,M),M=1,NE)	SPECS	328
	GO TO 104	SPECS	329
103	READ (5,503) D50(ID,1),WS(ID,1),CRSTRS(ID,1),CDSTRS(ID,1),	SPECS	330
	1ERODA(ID,1)	SPECS	331
	*WRITE(6,352)	SPECS	332
	*WRITE(6,355) ID,D50(ID,1),WS(ID,1),CRSTRS(ID,1),CDSTRS(ID,1),	SPECS	333
	1ERODA(ID,1)	SPECS	334
	DO 140 M=2,NE	SPECS	335
	D50(ID,M)=D50(ID,1)	SPECS	336
	WS(ID,M)=WS(ID,1)	SPECS	337
	CRSTRS(ID,M)=CRSTRS(ID,1)	SPECS	338
	CDSTRS(ID,M)=CDSTRS(ID,1)	SPECS	339
	ERODA(ID,M)=ERODA(ID,1)	SPECS	340
140	CONTINUE	SPECS	341
	IF (ID.GT.1) GO TO 107	SPECS	342
104	IF(UBSDI) GO TO 105	SPECS	343
	IF(NBSDI) GO TO 109	SPECS	344
	GO TO 107	SPECS	345
109	READ (5,503) (B050(M),M=1,NE)	SPECS	346
	GO TO 107	SPECS	347
105	READ (5,505) B050(1)	SPECS	348
	DO 106 I=1,NE	SPECS	349
	B050(I)=B050(1)	SPECS	350
106	CONTINUE	SPECS	351
107	CONTINUE	SPECS	352
C		SPECS	353
C	CARD 13=14	SPECS	354
C		SPECS	355
C	UNICS, REQUIRED FOR UNIFORM INITIAL CONDITIONS; THE CARD FOLLOW	SPECS	356
C	THIS SHOULD INCLUDE C(I,1) WHERE I=1. THIS 2-CARD-SET IS	SPECS	357
C	REPEATED FOR I=2,9	SPECS	358
C		SPECS	359
C	IF FLAG(1) DOES NOT EQUAL "UNICS," NONUNIFORM INITIAL CO	SPECS	360
C	DITIONS ARE ASSUMED. C(I,J) FOR J=1,NDS IS READ FOR THI	SPECS	361
C	CASE.	SPECS	362
C	C(ID,J)... INITIAL CONDITION AND SOL. ALREADY SOLVED AT PRESENT	SPECS	363
C	TIME STEP OR SOL. AT PREVIOUS TIME STEP	SPECS	364
C		SPECS	365
	IF (ID.GT.1) GO TO 205	SPECS	366
	DO 50 I=1,9	SPECS	367
	DO 50 J=1,NDS	SPECS	368
50	C(I,J)=0.0	SPECS	369
	DO 150 I=ID,IDN	SPECS	370
	READ(5,1025) FLAG(1)	SPECS	371
	*WRITE(6,520)	SPECS	372
	*WRITE(6,6100) I	SPECS	373
	IF(FLAG(1).EQ."UNICS") GO TO 130	SPECS	374
	*WRITE(6,6110)	SPECS	375
	READ (5,510) (C(I,J),J=1,NDS)	SPECS	376
	GO TO 150	SPECS	377
130	READ (5,505) C(I,1)	SPECS	378
	*WRITE(6,6120)	SPECS	379
	DO 145 J=1,NDS	SPECS	380
145	C(I,J)=C(I,1)	SPECS	381
150	CONTINUE	SPECS	382
205	*WRITE (6,420)	SPECS	383
	*WRITE (6,490)	SPECS	384
	NPRT=NDS/6+1	SPECS	385
	DO 160 I=1,NPRT	SPECS	386
	NST=(I-1)*6+1	SPECS	387
	IF (NST.GT.NDS) GO TO 160	SPECS	388
	II=NST+5	SPECS	389

	IF (II.GT.NDS, II=NDS	SPECS	390
	WRITE (6,530) (J,C(ID,J),J=NST,II)	SPECS	391
160	CONTINUE	SPECS	392
C		SPECS	393
	IF(ID .GT. 1) GO TO 10	SPECS	394
	IF(.NOT. NUOOF) GO TO 210	SPECS	395
C		SPECS	396
C	CARD 15=16=17	SPECS	397
C		SPECS	398
C	DX(J).....DISPERSION COEFFICIENT = X COMPONENT	SPECS	399
C	DY(J).....DISPERSION COEFFICIENT = Y COMPONENT	SPECS	400
C	ALMBDA....DECAY RATE	SPECS	401
C	AKJ(I)....ADSORPTION OR DESORPTION RATE, I=1,9	SPECS	402
C	AKP(I)....EQUAL 1 IF ADSORPTION OR DESORPTION OCCURS	SPECS	403
C	EQUAL 0 IF NO ADSORPTION OR DESORPTION OCCURS	SPECS	404
C	RHOSED(J).SPECIFIC WT. OF J-TH SEDIMENT	SPECS	405
C	POR.....POROSITY	SPECS	406
C	RHOWAT....SPECIFIC WT. OF WATER(1000KG(FORCE)/M**3)	SPECS	407
C**	VIS.....KINEMATIC VISCOSITY OF WATER, M**2/SEC	SPECS	408
C	ALFA (J)..DECAY TERM	SPECS	409
C	BETA(J)...SOURCE OR SINK TERM	SPECS	410
C	HS(J).....ELEMENT THICKNESS	SPECS	411
C		SPECS	412
	IF (INO.LE.0) GO TO 208	SPECS	413
	READ(S,351) ALMBDA,(AKP(I),I=1,3),BETA1,AREA1,NOOBET	SPECS	414
	DO 206 I=1,9	SPECS	415
	READ(S,336) (AKJ(I,J),J=1,NE)	SPECS	416
206	CONTINUE	SPECS	417
	READ(S,350) (RHOSED(I),I=1,3), RHOWAT, POR, VIS	SPECS	418
	WRITE(6,383)	SPECS	419
	WRITE(6,381) ALMBDA,AKP(1),AKP(2),AKP(3),BETA1,AREA1,NOOBET	FETFIX1	22
	DO 207 I=1,9	SPECS	421
	WRITE(6,385) I	SPECS	422
	WRITE(6,386) (AKJ(I,J),J=1,NE)	SPECS	423
207	CONTINUE	SPECS	424
	WRITE(6,384)	SPECS	425
	WRITE(6,381) (RHOSED(I),I=1,3), RHOWAT, POR, VIS	SPECS	426
208	READ (S,350)(DX(J),DY(J),ALFA(J),BETA(J),HS(J),J=1,NE)	SPECS	427
	WRITE (6,380) (J,DX(J),DY(J),ALFA(J),BETA(J),HS(J),	SPECS	428
	IJ=1,NE)	SPECS	429
	WRITE (6,520)	SPECS	430
	GO TO 240	SPECS	431
C		SPECS	432
210	IF (INO.LE.0) GO TO 209	SPECS	433
	READ(S,351) ALMBDA,(AKP(I),I=1,3),BETA1,AREA1,NOOBET	SPECS	434
	DO 212 I=1,9	SPECS	435
	READ(S,336) AKJ(I,1)	SPECS	436
	DO 212 J=1,NE	SPECS	437
	AKJ(I,J) = AKJ(I,1)	SPECS	438
212	CONTINUE	SPECS	439
	READ(S,350) (RHOSED(I),I=1,3), RHOWAT, POR, VIS	SPECS	440
	WRITE(6,383)	SPECS	441
	WRITE(6,381) ALMBDA,AKP(1),AKP(2),AKP(3),BETA1,AREA1,NOOBET	FETFIX1	23
	DO 213 I=1,9	SPECS	443
	WRITE(6,385) I	SPECS	444
	WRITE(6,386) (AKJ(I,J),J=1,NE)	SPECS	445
213	CONTINUE	SPECS	446
	WRITE(6,384)	SPECS	447
	WRITE(6,381) (RHOSED(I),I=1,3), RHOWAT, POR, VIS	SPECS	448
209	READ (S,350) DX(1),DY(1),ALFA(1),BETA(1),HS(1)	SPECS	449
	WRITE (6,580)	SPECS	450
	WRITE (6,380) NE,DX(1),DY(1),ALFA(1),BETA(1),HS(1)	SPECS	451

	WRITE (6,520)	SPECS	452
	DO 222 J=1,NE	SPECS	453
	DX(J)=DX(1)	SPECS	454
	DY(J)=DY(1)	SPECS	455
	ALFA(J)=ALFA(1)	SPECS	456
	BETA(J)=BETA(1)	SPECS	457
	HS(J)=HS(1)	SPECS	458
222	CONTINUE	SPECS	459
240	CONTINUE	SPECS	460
	IF(INO .LE. 0) GO TO 10	SPECS	461
C		SPECS	462
C**	CARD 18	SPECS	463
C		SPECS	464
C**	FOR ELEMENT M=	SPECS	465
C**	NBED(M) =INITIAL NUMBER OF BED LAYERS.	SPECS	466
C**	BDIV(M) =THICKNESS OF BED LAYERS(USED FOR ALL LAYERS	SPECS	467
C**	EXCEPT TOP LAYER).	SPECS	468
C**	XYSO(M) =THICKNESS OF TOP BED LAYER.	SPECS	469
C**	BED(M) =INITIAL TOTAL BED THICKNESS.	SPECS	470
C		SPECS	471
	IF(UBGEO) GO TO 122	SPECS	472
	IF(NBGEO) GO TO 121	SPECS	473
	GO TO 124	SPECS	474
121	READ(5,335) (NBED(M),BDIV(M),XYSO(M),BED(M),M=1,NE)	SPECS	475
	WRITE(6,532)	SPECS	476
	WRITE(6,535)(M,NBED(M),BDIV(M),XYSO(M),BED(M),BD50(M),M=1,NE)	SPECS	477
	GO TO 126	SPECS	478
122	READ(5,335) NBED(1),BDIV(1),XYSO(1),BED(1)	SPECS	479
	WRITE(6,531)	SPECS	480
	WRITE(6,537)NBED(1),BDIV(1),XYSO(1),BED(1),BD50(1)	SPECS	481
	DO 131 M=1,NE	SPECS	482
	NBED(M)=NBED(1)	SPECS	483
	BDIV(M)=BDIV(1)	SPECS	484
	XYSO(M)=XYSO(1)	SPECS	485
	BED(M)=BED(1)	SPECS	486
131	CONTINUE	SPECS	487
126	IF(UBCON) GO TO 129	SPECS	488
	IF(NBCON) GO TO 128	SPECS	489
	GO TO 124	SPECS	490
C		SPECS	491
C**	CARD 19	SPECS	492
C		SPECS	493
C**	***NOTE=BED LAYERS ARE NUMBERED FROM DEEPEST TO SHALLOWEST***	SPECS	494
C		SPECS	495
C**	FOR ELEMENT M, BED LAYER J=	SPECS	496
C**	GBA(M,J)=WEIGHT FRACTION OF CLAY IF CLAY, IS THE	SPECS	497
C**	LIMITING BED LAYER.	SPECS	498
C**	=WEIGHT FRACTION OF SAND IF SAND IS THE	SPECS	499
C**	THE LIMITING BED LAYER.	SPECS	500
C**	GBB(M,J)=WEIGHT FRACTION OF SILT.	SPECS	501
C**	GBC(M,J)=WEIGHT FRACTION OF SAND IF CLAY IS THE	SPECS	502
C**	LIMITING BED LAYER.	SPECS	503
C**	=WEIGHT FRACTION OF SILT IF SAND IS THE	SPECS	504
C**	LIMITING BED LAYER.	SPECS	505
C**	GBD(M,J)=CHEMICAL CONCENTRATION OF CONTAMINANT PER UNIT WEIGHT	SPECS	506
C**	OF SEDIMENT(CLAY OR SAND DEPENDING ON THE SEDIMENT TYPE	SPECS	507
C**	THAT GBA(M,J) APPLIES TO).	SPECS	508
C**	GBE(M,J)=CHEMICAL CONCENTRATION OF CONTAMINANT PER UNIT	SPECS	509
C**	WEIGHT OF SILT.	SPECS	510
C**	GBF(M,J)=CHEMICAL CONCENTRATION OF CONTAMINANT PER UNIT WEIGHT	SPECS	511
C**	OF SEDIMENT(CLAY OR SILT DEPENDING ON THE SEDIMENT TYPE	SPECS	512
C**	THAT GBC(M,J) APPLIES TO).	SPECS	513

C		SPECS	514
C**	***NOTE=APPROPRIATE UNITS FOR GBD, GBE, AND GBF ARE	SPECS	515
C**	PCI/KG OR KG/KG.***	SPECS	516
C		SPECS	517
126	DO 127 M=1,NE	SPECS	518
	NBED3=NBED(M)	SPECS	519
	READ(5,336) (GBA(M,J),GBB(M,J),GBC(M,J),GBD(M,J),GBE(M,J),GBF(M,J)	SPECS	520
	1),J=1,NBED3)	SPECS	521
127	CONTINUE	SPECS	522
	DO 195 M=1,NE	SPECS	523
	NBED3=NBED(M)	SPECS	524
	DO 195 J=1,NBED3	SPECS	525
	GBG(M,J)=GBA(M,J)*GBD(M,J)+GBB(M,J)*GBE(M,J)+GBC(M,J)*GBF(M,J)	SPECS	526
195	CONTINUE	SPECS	527
	WRITE(6,533)	SPECS	528
	DO 196 M=1,NE	SPECS	529
	NBED3=NBED(M)	SPECS	530
	WRITE(6,536) (J,GBA(M,J),GBB(M,J),GBC(M,J),GBD(M,J),GBE(M,J),	SPECS	531
	1GBF(M,J),GBG(M,J),J=1,NBED3)	SPECS	532
196	CONTINUE	SPECS	533
	GO TO 124	SPECS	534
129	READ(5,336) GBA(1,1),GBB(1,1),GBC(1,1),GBD(1,1),GBE(1,1),GBF(1,1)	SPECS	535
	GBG(1,1)=GBA(1,1)*GBD(1,1)+GBB(1,1)*GBE(1,1)+GBC(1,1)*GBF(1,1)	SPECS	536
221	DO 123 M=1,NE	SPECS	537
	NBEDM=NBED(1)	SPECS	538
	DO 123 J=1,NBEM	SPECS	539
	GBA(M,J)=GBA(1,1)	SPECS	540
	GBB(M,J)=GBB(1,1)	SPECS	541
	GBC(M,J)=GBC(1,1)	SPECS	542
	GBD(M,J)=GBD(1,1)	SPECS	543
	GBE(M,J)=GBE(1,1)	SPECS	544
	GBF(M,J)=GBF(1,1)	SPECS	545
	GBG(M,J)=GBG(1,1)	SPECS	546
123	CONTINUE	SPECS	547
	WRITE(6,539)	SPECS	548
	WRITE(6,538) GBA(1,1),GBB(1,1),GBC(1,1),GBD(1,1),GBE(1,1),GBF(1,1)	SPECS	549
	1,GBG(1,1)	SPECS	550
124	CONTINUE	SPECS	551
	GO TO 10	SPECS	552
C		SPECS	553
C	CARD 20	SPECS	554
C		SPECS	555
C	READ THE BOUNDARY CONDITIONS	SPECS	556
C		SPECS	557
C	LBC.....NUMBER OF SPECIFIED BOUNDARY CONDITION NODES	SPECS	558
C	KBC.....NUMBER OF DERIVATIVE BOUNDARY CONDITION NODES	SPECS	559
C	NBC(ID,J)....BOUNDARY NODE NUMBER	SPECS	560
C	BC(ID,J)....BOUNDARY NODE VALUE (SPECIFIED)	SPECS	561
C	MBC(ID,J)....BOUNDARY NODE NUMBER	SPECS	562
C	DBC(ID,J)....BOUNDARY NODE VALUE (DERIVATIVE)	SPECS	563
C		SPECS	564
250	CONTINUE	SPECS	565
	READ(5,341)LBC,KBC	SPECS	566
	JMAX=MAX0(KBC,LBC)	SPECS	567
	IF(JMAX .LE. 0) GO TO 20	SPECS	568
	NCHECK=1	SPECS	569
	MCHECK=1	SPECS	570
	IF(LBC .EQ. 0) NCHECK = 0	SPECS	571
	IF(KBC .EQ. 0) MCHECK = 0	SPECS	572
	DO 260 J=1,JMAX	SPECS	573
	READ(5,600) NBC(ID,J),BC(ID,J),MBC(ID,J),DBC(ID,J)	SPECS	574
C		SPECS	575

C**	B,C. NODES MUST BE INPUT IN A NUMERICALLY INCREASING SEQUENCE	SPECS	576
C		SPECS	577
	IF(NBC(ID,J) .LT. NCHECK) WRITE(6,2000) NBC(ID,J)	SPECS	578
	IF(MBC(ID,J) .LT. MCHECK) WRITE(6,2010) MBC(ID,J)	SPECS	579
	NCHECK=MBC(ID,J)	SPECS	580
	MCHECK=MBC(ID,J)	SPECS	581
260	CONTINUE	SPECS	582
	WRITE(6,460)	SPECS	583
	WRITE(6,490)	SPECS	584
	WRITE(6,470) (NBC(ID,J),BC(ID,J),J=1,LBC)	SPECS	585
	IF(KBC .LE. 0) GO TO 20	SPECS	586
	WRITE(6,590)	SPECS	587
	WRITE(6,470) (MBC(ID,J),DBC(ID,J),J=1,KBC)	SPECS	588
265	CONTINUE	SPECS	589
C		SPECS	590
	GO TO 20	SPECS	591
270	CONTINUE	SPECS	592
	IF(ID.EQ.1.AND.KK.EQ.1.AND.RSTRT) READ(1) RAT, KK, C, NBED,	FETFIX1	24
	1 XY30, BED, GBA, GBB, GBC, GBD, GBE, GBF, GBG	SPECS	594
	DO 275 J=1, NDS	SPECS	595
	H(J) = 0.0	SPECS	596
	VX(J) = 0.0	SPECS	597
	VY(J) = 0.0	SPECS	598
275	CONTINUE	SPECS	599
	IF(.NOT. NUVEL) GO TO 280	SPECS	600
C		SPECS	601
C	READ THE FLOW FIELD	SPECS	602
C		SPECS	603
C	CARD 21	SPECS	604
C		SPECS	605
C	VX(J).....X-COMPONENT OF VELOCITY	SPECS	606
C	VY(J).....Y-COMPONENT OF VELOCITY	SPECS	607
C	H(J).....FLOW DEPTH	SPECS	608
	IF (VELR) GO TO 295	SPECS	609
	READ(5,430) (H(J), VX(J), VY(J), J=1, NDS)	SPECS	610
	GO TO 300	SPECS	611
280	CONTINUE	SPECS	612
	READ(5,430) H(1), VX(1), VY(1)	SPECS	613
	DO 290 J=1, NDS	SPECS	614
	H(J)=H(1)	SPECS	615
	VX(J)=VX(1)	SPECS	616
	VY(J)=VY(1)	SPECS	617
290	CONTINUE	SPECS	618
	GO TO 300	SPECS	619
295	IF(ID.EQ.1.AND.KKK.EQ.1) READ(5,336) VFREQ	SPECS	620
	AT = RAT+T*(KKK-1)	SPECS	621
	IF (ID.EQ.1 .AND. AMOD(AT, VFREQ) .LT. T) CALL RVEL(AT, RAT,	SPECS	622
	1 NCNDS, VFREQ)	SPECS	623
300	CONTINUE	SPECS	624
	IF (MOD(KK, NPRNT).NE.0 .AND. KK.NE.NTP) GO TO 305	SPECS	625
	WRITE(6,440)	SPECS	626
	WRITE(6,480)	SPECS	627
	NDSM2 = NDS - 2	SPECS	628
	WRITE(6,450) (J, VX(J), VY(J), H(J), J+1, VX(J+1), VY(J+1), H(J+1),	SPECS	629
	1 J+2, VX(J+2), VY(J+2), H(J+2), J=1, NDSM2, 3)	SPECS	630
305	CONTINUE	SPECS	631
C		SPECS	632
C	IF (KK.EQ.2) WRITE (6,7005) KK, ID, (FLAG(IJK), IJK=1,6)	SPECS	633
C	IF (KK.EQ.1.AND.ID.EQ.7) WRITE (6,7005) KK, ID, (FLAG(IJK), IJK=1,6)	SPECS	634
C7005	FORMAT (2I5,6(1X,A5))	SPECS	635
	IF(.NOT. WAVE) GO TO 309	SPECS	636
	IF (WAVR) GO TO 312	SPECS	637

C		SPECS	638
C**	CARD 22,A,B,C	SPECS	639
C		SPECS	640
C**	THIS SEGMENT READS THE DATA REQUIRED FOR CALCULATION OF	SPECS	641
C**	WAVE SEDIMENT TRANSPORT. INPUT DATA AT THE CORNER NODES ONLY.	SPECS	642
C		SPECS	643
C**	NOONO =THE NODE NUMBER(CORNER NODES ONLY).	SPECS	644
C**	ZTYP =WAVE ZONE TYPE	SPECS	645
C**	=WAVE, IF NODE IS BEYOND SURF ZONE	SPECS	646
C**	=SURF, IF NODE IS IN SURF ZONE	SPECS	647
C**	=WAVC, IF NODE IS IN A WAVE ZONE AND THE WAVE	SPECS	648
C**	CHARACTERISTICS ARE TO BE CALCULATED.	SPECS	649
C**	NC =NUMBER OF WAVE CHARACTERISTICS TO READ FOR	SPECS	650
C**	THIS NODE(DEFAULT = 1).	SPECS	651
C**	WV =WIND VELOCITY, M/SEC.	SPECS	652
C**	DM =MEAN FETCH DEPTH, M.	SPECS	653
C**	F =EFFECTIVE FETCH LENGTH, M.	SPECS	654
C**	A =WAVE AMPLITUDE, M.	SPECS	655
C**	K =WAVE NUMBER=2*PI/WAVE LENGTH, 1./M.	SPECS	656
C**	w =WAVE FREQUENCY=2*PI/WAVE PERIOD, 1./SEC.	SPECS	657
C**	HB =WAVE HEIGHT AT BREAKING, M.	SPECS	658
C**	ALPHAC =ANGLE BETWEEN WAVE RAY AND THE GRADIENT OF THE	SPECS	659
C**	BOTTOM BATHYMETRY, DEGREES.	SPECS	660
C		SPECS	661
	HEAD(5,1025) FLAG(1)	SPECS	662
	WRITE(6,520)	SPECS	663
	IF (FLAG(1) .EQ. "WIND") GO TO 1315	SPECS	664
	WRITE(6,6125)	SPECS	665
	READ(5,6126) WNDVEL	SPECS	666
	WRITE(6,6127) WNDVEL	SPECS	667
	GO TO 6131	SPECS	668
1315	WRITE(6,6130)	SPECS	669
6131	WRITE(6,2075)	SPECS	670
	DO 306 I=1,NDS	SPECS	671
306	NODE(I)=0	SPECS	672
	NP=1	SPECS	673
307	READ(5,1000) NODNO,ZTYP(NP),NC(NP)	SPECS	674
	IF(ABS(EOF(5)).NE.0.0) GO TO 309	SPECS	675
	IF (NODNO.EQ.0) GO TO 309	SPECS	676
308	IF(NC(NP) .EQ. 0) NC(NP)=1	SPECS	677
	NODE(NODNO)=NP	SPECS	678
	NO=NC(NP)	SPECS	679
	IF(ZTYP(NP) .NE. "WAVC,") GO TO 311	SPECS	680
	IF(FLAG(1) .EQ. "WIND") GO TO 315	SPECS	681
	READ(5,1010) WV,DM,F	SPECS	682
	IF (F.EQ.0.) GO TO 317	SPECS	683
	GO TO 316	SPECS	684
315	READ(5,1010) DM,F	SPECS	685
	IF (F.EQ.0.) GO TO 317	SPECS	686
	WV = WNDVEL	SPECS	687
316	CALL WAVSIM(NP,WV,DM,F)	SPECS	688
	ZTYP(NP)="WAVE,"	SPECS	689
	GO TO 318	SPECS	690
317	HB(NP) = 0.	SPECS	691
	ALPHAC(NP) = 0.	SPECS	692
	DO 319 I=1,NO	SPECS	693
319	K(NP,I) = 9999.	SPECS	694
	ZTYP(NP) = "SURF,"	SPECS	695
	GO TO 321	SPECS	696
318	WRITE(6,2077) NOONO,WV,DM,F,A(NP,1),K(NP,1),w(NP,1)	SPECS	697
	NP=NP+1	SPECS	698
	GO TO 307	SPECS	699

311	CONTINUE	SPECS	700
	IF (ZTYP(NP) .EQ. "WAVE,") READ(5,1010)((A(NP,I),K(NP,I),W(NP,I)),	SPECS	701
	SI#1,NO)	SPECS	702
	IF (ZTYP(NP) .EQ. "WAVE,") WRITE(6,2080) NODNO, ((A(NP,I),	SPECS	703
	SK(NP,I), W(NP,I)),I=1,NO)	SPECS	704
	IF (ZTYP(NP) .EQ. "SURF,") READ(5,1020) HB(NP),ALPHAC(NP),(K(NP,I),	SPECS	705
	SI#1,NO)	SPECS	706
321	IF (ZTYP(NP) .EQ. "SURF,") WRITE(6,2090) NODNO, HB(NP),	SPECS	707
	SALPHAC(NP), (K(NP,I),I=1,NO)	SPECS	708
	NP=NP+1	SPECS	709
	GO TO 307	SPECS	710
312	IF (KKK.EQ.1) CALL RWIND (WFREQ,LW)	SPECS	711
	AT = RAT+T*(KKK-1)	SPECS	712
	IF (IO.EQ.1 .AND. AMOD(AT,WFREQ) .LT. T) CALL RWAVE (AT,WFREQ)	SPECS	713
309	RETURN	SPECS	714
310	FORMAT (16A5/16A5)	SPECS	715
320	FORMAT (1H1)	SPECS	716
325	FORMAT (2I5)	SPECS	717
330	FORMAT (10X,16A5/10X,16A5)	SPECS	718
335	FORMAT (I5,3E10.3)	SPECS	719
336	FORMAT (6E10.3)	SPECS	720
341	FORMAT (12I5)	SPECS	721
348	FORMAT (5E10.3)	SPECS	722
349	FORMAT (2E10.3,T11,A10)	SPECS	723
350	FORMAT (7E10.3)	SPECS	724
351	FORMAT (6E10.3,I5)	SPECS	725
352	FORMAT (//,T31," SUSPENDED SEDIMENT SCOUR DEPOSITION",/,	SPECS	726
	\$T12"SUBSTANCE ELEMENT SEDIMENT FALL CRIT. SHEAR CRIT. SHEA	SPECS	727
	\$R ERODABILITY",/,T15"ID",T23,"NUMBER DIAMETER VELOCITY STRES	SPECS	728
	\$S STRESS COEFFICIENT",/)	SPECS	729
353	FORMAT (T15,I2,T24,I4, T31,1PE10.3,T42,E10.3,T53,E10.3,T65,E10.3,T	SPECS	730
	\$77,E10.3)	SPECS	731
355	FORMAT (T15,I2,T25,"ALL" T31,1PE10.3,T42,E10.3,T53,E10.3,T65,E10.3,T	SPECS	732
	\$77,E10.3)	SPECS	733
370	FORMAT (10X,"ERROR = TOO MANY NODES OR TOO MANY ELEMENTS")	SPECS	734
380	FORMAT (10X,I5,T19,1PE10.3,1X,E10.3,1X,3(1X,E10.3))	SPECS	735
381	FORMAT (10X,6E10.3,I5//)	FETFIX1	25
382	FORMAT (10X,5E10.3//)	SPECS	737
383	FORMAT (//,10X,"ALMBDA AKP(1) AKP(2) AKP(3) BETA1	FETFIX1	26
	I AREA1 NODBET"/)	FETFIX1	27
384	FORMAT (//,11X,"RHOSED(1) RHOSED(2) RHOSED(3) RHOAT POROSITY V	SPECS	739
	\$ISCOSITY",/)	SPECS	740
385	FORMAT (//,10X,"KD VALUES FOR SUBSTANCE",I2,/))	SPECS	741
386	FORMAT (10X,1P6E12.4)	SPECS	742
390	FORMAT (//,10X,"CONNECTIVITY TABLE",//10X,"ELEMENT",T20,"NODES (IN	SPECS	743
	\$CC# SEQUENCE)",/)	SPECS	744
400	FORMAT (//,10X,"NCOL = (BAND WIDTH + 1)/2 = ",I5)	SPECS	745
410	FORMAT (10X,"DATA ERROR = BAND WIDTH TOO LARGE",//)	SPECS	746
420	FORMAT (//,10X,"INITIAL CONDITION FOR EACH NODE",/)	SPECS	747
430	FORMAT (3E10.3)	SPECS	748
435	FORMAT (2I5,E10.3)	SPECS	749
440	FORMAT (//,10X,"VELOCITY AND DEPTH INPUT DATA",/)	SPECS	750
450	FORMAT (10X,I3,1X,1P3E11.4,1X,I3,1X,1P3E11.4,1X,I3,1X,1P3E11.4)	SPECS	751
460	FORMAT (//,10X,"BOUNDARY CONDITIONS",/)	SPECS	752
470	FORMAT (10X,I5,5X,1PE15.5)	SPECS	753
480	FORMAT (10X,"NODE",5X,"VX",9X,"VY",9X,"H",6X,"NODE",5X,"VX",9X,	SPECS	754
	I "VY",9X,"H",6X,"NODE",5X,"VX",9X,"VY",9X,"H")	SPECS	755
490	FORMAT (10X," NODE CONCENTRATION",/)	SPECS	756
500	FORMAT (I5,2E10.2,T1,A5)	SPECS	757
503	FORMAT (5E10.3)	SPECS	758
505	FORMAT (E10.3)	SPECS	759
509	FORMAT (I5)	SPECS	760

510	FORMAT (6E10.3)	SPECS	761
515	FORMAT(10X,"NONUNIFORM INITIAL CONDITIONS",11X,IS,/))	SPECS	762
520	FORMAT (//)	SPECS	763
530	FORMAT(10X,6(1S,1PE15.7))	SPECS	764
531	FORMAT(14X,"NBED",5X,"BDIV",5X,"XYSO",5X,"BED",5X,"BD50",/)	SPECS	765
532	FORMAT(9X,"M",1X,"NBED",5X,"BDIV",5X,"XYSO",5X,"BED",5X,"BD50",/)	SPECS	766
533	FORMAT(7X,"J",5X,"GBA",7X,"GBB",7X,"GBC",7X,"GBD",7X,"GBE",7X, 1"GBF",7X,"GBG",/)	SPECS	767
535	FORMAT(5X,2IS,4E10.3,//)	SPECS	769
536	FORMAT(5X,IS,7E10.3)	SPECS	770
537	FORMAT(10X,IS,4E10.3,//)	SPECS	771
538	FORMAT(10X,7E10.3,//)	SPECS	772
539	FORMAT(11X,"GBA(1,1) GBB(1,1) GBC(1,1) GBD(1,1) GBE(1,1) GBF(1,1) GBG(1,1)",/)	SPECS	773
540	FORMAT (10X,"DATA ERROR = NODE TABLE")	SPECS	775
550	FORMAT (10X,"TOTAL NUMBER OF CONSTITUENTS",12X,IS,/,10X,"NUMBER OF 1TRIANGULAR ELEMENTS",13X,I4,/,10X,"NUMBER OF NODES",26X,I4,/,10X," 2NUMBER OF TIME PLANES",20X,I4,/,10X,"SIMULATION TIME STEP",24X,1PE 310.4/10X,"PRINT FREQUENCY",25X,IS)	SPECS	776
560	FORMAT (7IS)	SPECS	777
570	FORMAT (10X,IS,2X,6IS)	SPECS	778
580	FORMAT(//,10X,"ELEMENT",T26,"X",T37,"Y",7X,"DECAY", 1 5X,"SOURCE",5X,"ELEMENT",/,10X,"NUMBER",T21,"DISPERSION DISPERS SSION TERM TERM THICKNESS",/)	SPECS	780
590	FORMAT(10X," NODE BOUNDARY FLUX",/)	SPECS	781
600	FORMAT (2(1S,E10.2))	SPECS	782
610	FORMAT(10X,"NODE",5X,"X-COORD",6X,"Y-COORD",/)	SPECS	783
611	FORMAT(10X,I3,3X,1PE11.4,2X,E11.4)	SPECS	784
705	FORMAT(10X,"NUMBER OF AREA SOURCES",18X,IS)	SPECS	785
706	FORMAT(/,13X,"CONSTITUENT",2X,"ELEMENT",3X,"SOURCE",/,15X, 1 "NUMBER",5X,"NUMBER",3X,"STRENGTH")	SPECS	786
707	FORMAT(15X,I4,6X,IS,5X,1PE8.2)	SPECS	787
708	FORMAT(/,10X,"NUMBER OF POINT SOURCES",17X,IS)	SPECS	788
709	FORMAT(/,13X,"CONSTITUENT",3X,"NODE",5X,"SOURCE",/,15X, 1 "NUMBER",5X,"NUMBER",3X,"STRENGTH")	SPECS	789
1000	FORMAT(1S,AS,IS)	SPECS	790
1010	FORMAT(3E10.4)	SPECS	791
1020	FORMAT(2F10.0/(5F10.0))	SPECS	792
1030	FORMAT(16A5)	SPECS	793
2000	FORMAT(1X,"SPECIFIED BOUNDARY CONDITION NODE",IS," IS OUT OF SEQUE SENCE.")	SPECS	794
2010	FORMAT(1X,"DERIVATIVE BOUNDARY CONDITION NODE",IS," IS OUT OF SEQU SENCE.")	SPECS	795
2020	FORMAT(1H0,9X"STYP ="F5.2)	SPECS	796
2030	FORMAT(1H+,T23"EXPLICIT SOLUTION SELECTED")	SPECS	797
2040	FORMAT(1H+,T23"CRANK-NICHOLSON SOLUTION SELECTED")	SPECS	798
2050	FORMAT(1H+,T23"IMPLICIT SOLUTION SELECTED")	SPECS	799
2060	FORMAT(1X,"STYP NOT SPECIFIED WITHIN THE RANGE 0. TO 1. -PROGRAM \$TERMINATED")	SPECS	800
2070	FORMAT(10X"CONSTITUENT"1S,10X"FLAG FOR STANDARD FORM"1S)	SPECS	801
2075	FORMAT(//11X,"HAVE DATA INPUT"//11X"NODE"19" WAVE ZONE"/)	SPECS	802
2077	FORMAT(10X,IS,T22,"WAVE",T28,"WIND VEL.",E12.5,T51,"MEAN DEPTH", 1\$E12.5,T75,"EFFECTIVE FETCH LENGTH=",E12.5,/,T31," A=",E12.5,T46," 1\$K=",E12.5,T61," H=",E12.5,/))	SPECS	803
2080	FORMAT(10X,IS,T22" WAVE" T31" A="E12.5,T46" K="E12.5,T61" H=" 1\$E12.5, (/T34,E12.5,T33" T48"="E12.5,T63"="E12.5))	SPECS	804
2090	FORMAT(10X,IS,T22" SURF" T31" H="E12.5,T46" ALPHAC="F8.1,T61" K=" 1\$E12.5, (/T64,E12.5,T63"="))	SPECS	805
1025	FORMAT(12(A5,1X))	SPECS	806
6000	FORMAT(10X,"THE FOLLOWING OPTIONS HAVE BEEN ACTIVATED=",/)	SPECS	807
6015	FORMAT(15X,"NONUNIFORM COEFFICIENTS(DX, DY, ALFA, BETA, HS)")	SPECS	808
6020	FORMAT(15X,"NONUNIFORM VELOCITY AND/OR DEPTH.")	SPECS	809

6025	FORMAT(15X,"TIME VARYING BOUNDARY CONDITIONS.")	SPECS	823
6030	FORMAT(15X,"TIME VARYING HYDRAULICS.")	SPECS	824
6035	FORMAT(15X,"TIME VARYING INPUT DATA.")	SPECS	825
6040	FORMAT(15X,"UNIFORM SUSPENDED SEDIMENT DIAMETER.")	SPECS	826
6045	FORMAT(15X,"NONUNIFORM SUSPENDED SEDIMENT DIAMETER.")	SPECS	827
6050	FORMAT(15X,"UNIFORM BED SEDIMENT DIAMETER.")	SPECS	828
6055	FORMAT(15X,"NONUNIFORM BED SEDIMENT DIAMETER.")	SPECS	829
6060	FORMAT(15X,"UNIFORM BED CONFIGURATION.")	SPECS	830
6065	FORMAT(15X,"NONUNIFORM BED CONFIGURATION.")	SPECS	831
6070	FORMAT(15X,"UNIFORM BED CONCENTRATION.")	SPECS	832
6075	FORMAT(15X,"NONUNIFORM BED CONCENTRATION.")	SPECS	833
6100	FORMAT(10X,"THE FOLLOWING OPTION HAS BEEN SELECTED FOR SUBSTANCE "	SPECS	834
	I2,"."/)	SPECS	835
6110	FORMAT(15X,"NONUNIFORM INITIAL CONDITION.",//)	SPECS	836
6120	FORMAT(15X,"UNIFORM INITIAL CONDITION.")	SPECS	837
6125	FORMAT(15X,"SPATIALLY CONSTANT WIND")	SPECS	838
6126	FORMAT(F10.0)	SPECS	839
6127	FORMAT(18X,"OF MAGNITUDE",F5.1," M/SEC")	SPECS	840
6130	FORMAT(15X,"SPATIALLY VARYING WIND")	SPECS	841
	END	SPECS	842
	SUBROUTINE SURFTR(ID,H,UEXT,QT)	SURFTR	2
C		SURFTR	3
C**	THIS SUBROUTINE CALCULATES THE LITTORAL (LONGSHORE) TRANSPORT	SURFTR	4
C**	OF SEDIMENT IN THE SURF ZONE.	SURFTR	5
C**	BASED ON WORK DONE BY KOMAR	SURFTR	6
C		SURFTR	7
C	CHARACTER*5 ZTYP	FETFIX1	28
	LOGICAL WAVE	SURFTR	9
C		SURFTR	10
	COMMON/WAVE/ NODE(240), NC(135), A(135,10), K(135,10),	SURFTR	11
	H(135,10), HB(135), ALPHAC(135), WAVE, N, D	SURFTR	12
	, VIS, DPTH(135)	SURFTR	13
	COMMON /ZTYPE/ ZTYP(135)	SURFTR	14
	COMMON/BLK12/ ACOF(6), U(6), V(6), DD(2), AKJ(9,100), ALHBA,	SURFTR	15
	RHOWAT, AKP(3)	SURFTR	16
	COMMON/BLK15/ ILAYR(100,3), XYSO(100), BDIV(100), NBEJ(100),	SURFTR	17
	BED(100), RHOSED(3), XNT(100,3), RSAVE1(100),	SURFTR	18
	RSV2(100), RSV3(100)	SURFTR	19
	REAL K, IL	SURFTR	20
C		SURFTR	21
C**	SLOPE IS THE SLOPE OF THE BEACH.	SURFTR	22
C		SURFTR	23
	SLOPE = .012	SURFTR	24
	PI=ACOS(-1.)	SURFTR	25
	UEXT=UEXT/3600./24.	SURFTR	26
	ΔPRIME=.6	SURFTR	27
	G=9.8	SURFTR	28
	SPWSED=RHOSED(ID)	SURFTR	29
	SPWAT=RHOWAT	SURFTR	30
	RHOS=SPWSED/G	SURFTR	31
	RHOW=SPWAT/G	SURFTR	32
	ALPHA = ABS(ALPHAC(N))*PI/180.0	SURFTR	33
	C=SQRT(G*H)	SURFTR	34
	EB=RHOW*G*HB(N)**2/8.	SURFTR	35
	UM=SQRT(2.*EB/(RHOW*H))	SURFTR	36
	ECNB=0.	SURFTR	37
	T1=RHOW*G*HB(N)**2*C/16.	SURFTR	38
	NO=NC(N)	SURFTR	39
	DO 10 I=1,NO	SURFTR	40
C	WRITE (6,99) NO,N,I,K(N,I),H	SURFTR	41
C99	FORMAT (" NO, N, I, K(N,I), H = ",3I5,1P2E12.4)	SURFTR	42
	ECNB=T1*(1.+(2.*K(N,I)*H/SINH(2.*K(N,I)*H)))+ECNB	SURFTR	43

```

10 CONTINUE
C
C** CALCULATE THE IMMERSED WEIGHT LITTORAL TRANSPORT RATE, IL, IN
C** KG(FORCE)/DAY.
C
UBARL=2.7*UM*SIN(ALPHAB)*COS(ALPHAB) + UEXT
COSALB = COS(ALPHAB)
SINALB = SIN(ALPHAB)
C
WRITE (6,2020) SINALB,COSALB
C2020 FORMAT (" ?????? SIN(ALPHAB), COS(ALPHAB) = ",1P2E12,5)
IL=.28*ECNB*UBARL/UM*3600.*24.
C
C** NOW CALCULATE QT IN KG(FORCE)/DAY , BASED ON THE SEDIMENT
C** DRY WEIGHT.
C
QT=IL*(RHOS/(RHOS-RHOW))
C
C** CALCULATE QT PER UNIT WIDTH OF SURF ZONE.
C
HW = DPTHS(N) / SLOPE *(H/DPTHS(N))
QT = QT / HW
C
WRITE (6,1030) RHOS,RHOW,ALPHAB,EB,UM,T1
C1030 FORMAT ("***** RHOS,RHOW,ALPHAB,EB,UM,T1=",
C
1 /3X,6E12.4)
C
WRITE (6,1035) ECNB,UBARL,IL,HW,QT
C1035 FORMAT ("***** ECNB,UBARL,IL,HW,QT=",
C
1 /3X,5E12.4)
C
WRITE (6,1010) UEXT
C1010 FORMAT (" XXXXX UEXT=UEXY/3600./24.",F12.4)
UEXT=UEXT*3600.*24.
C
WRITE (6,1020) UEXT
C1020 FORMAT (" XXXXX UEXT=UEXY*3600.*24.",F12.4)
RETURN
END
SUBROUTINE TRANSP(NE,NROW,NCOL,T,ID,IND,INO,K,NPRNT,NTP)
C
C THIS ROUTINE CONSTRUCTS THE COEFFICIENT MATICES AND LOAD VECTOR
C FOR THE SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS*
C (P) (DY/DT) = (S) (Y) + (R)
C
COMMON /BLK2/P(240,86),F(240)
LEVEL 2,P,F
COMMON /BLK3/S(240,86)
LEVEL 2,S
COMMON /BLK4/ R(240), RPAST(240,7), NODBET, BETA1, AREA1
LEVEL 2,R,RPAST,NODBET,BETA1,AREA1
COMMON /BLK6/VX(240),VY(240),H(240),STRESS(3)
COMMON /BLK7/ NOQ(240,6),X(240),Y(240)
COMMON /BLK10/PEL(6,6),SEL(6,6),REL(6)
COMMON /BLK11/D50(3,100),BD50(100),SR(3,100),SD(3,100)
COMMON /BLK12/ACOF(6),U(6),V(6),D(2),AKJ(9,100),ALMBDA,RHOMAT,
1 AKP(3)
COMMON /BLK13/WS(3,100),CRSTRS(3,100),CDSYRS(3,100),ERODA(3,100)
COMMON /BLK14/GBA(100,10),GBB(100,10),GBC(100,10),GRD(100,10),
1 GBE(100,10),GBF(100,10),GBG(100,10),POR
COMMON /BLK15/ILAYR(100,3),XYSO(100),BOIV(100),NBED(100),BED(100),
1 RHOSD(3),XNT(100,3),RSAV1(100),RSAV2(100),RSAV3(100)
COMMON/OPTION/ CLAYX, SANDX
C
LOGICAL CLAYX, SANDX
C
SURFTR 44
SURFTR 45
SURFTR 46
SURFTR 47
SURFTR 48
SURFTR 49
SURFTR 50
SURFTR 51
SURFTR 52
SURFTR 53
SURFTR 54
SURFTR 55
SURFTR 56
SURFTR 57
SURFTR 58
SURFTR 59
SURFTR 60
SURFTR 61
SURFTR 62
SURFTR 63
SURFTR 64
SURFTR 65
SURFTR 66
SURFTR 67
SURFTR 68
SURFTR 69
SURFTR 70
SURFTR 71
SURFTR 72
SURFTR 73
SURFTR 74
SURFTR 75
SURFTR 76
SURFTR 77
SURFTR 78
TRANSP 2
TRANSP 3
TRANSP 4
TRANSP 5
TRANSP 6
TRANSP 7
TRANSP 8
FETFIX4 24
TRANSP 9
FETFIX4 25
TRANSP 10
FETFIX4 26
TRANSP 11
TRANSP 12
TRANSP 13
TRANSP 14
TRANSP 15
TRANSP 16
TRANSP 17
TRANSP 18
TRANSP 19
TRANSP 20
TRANSP 21
TRANSP 22
TRANSP 23
TRANSP 24
TRANSP 25

```


C		TRANSP	26
	NBND=2*NCOL-1	TRANSP	27
	DO 100 I=1,NROW	TRANSP	28
	R(I)=0.	TRANSP	29
	DO 100 J=1,NBND	TRANSP	30
	P(I,J)=0.	TRANSP	31
	S(I,J)=0.	TRANSP	32
100	CONTINUE	TRANSP	33
C		TRANSP	34
	DO 103 M=1,NE	TRANSP	35
	AREA=0.0	TRANSP	36
	CALL ACOEFS(M,AREA)	TRANSP	37
	CALL PMATRIX(M,AREA)	TRANSP	38
	IF (INO.LE.0) GO TO 111	TRANSP	39
	IF (ID.LE.3) CALL SHEAR(M)	TRANSP	40
	GO TO (104,105,106,107,108,108,108) ID	TRANSP	41
104	IF(SANDX) CALL SAND(M, ID, T, AREA)	TRANSP	42
	IF(CLAYX) CALL CLAY(M, ID, T)	TRANSP	43
	CALL SEDIME(M, AREA, ID)	TRANSP	44
	GO TO 110	TRANSP	45
105	CALL SILT(M, ID, T)	TRANSP	46
	CALL SEDIME(M, AREA, ID)	TRANSP	47
	GO TO 110	TRANSP	48
106	IF(SANDX) CALL SILT(M, ID, T)	TRANSP	49
	IF(CLAYX) CALL SAND(M, ID, T, AREA)	TRANSP	50
	CALL SEDIME(M, AREA, ID)	TRANSP	51
	GO TO 110	TRANSP	52
107	CALL DISOLV(M, AREA, ID, IDN, T)	TRANSP	53
	GO TO 111	TRANSP	54
108	CALL PARTIC(M, AREA, ID, IDN, T)	TRANSP	55
	IF (ID.EQ.IDN) CALL BEDHIS(M, T)	TRANSP	56
	GO TO 111	TRANSP	57
110	CONTINUE	TRANSP	58
C		TRANSP	59
C**	WRITE THE FOLLOWING INFORMATION FOR THE LAST 50 ODD	TRANSP	60
C**	NUMBERED TIME PLANES.	TRANSP	61
C		TRANSP	62
C	IF(M.EQ.1) WRITE(6,200)	TRANSP	63
C	IF(M.EQ.1) WRITE(6,205) K	TRANSP	64
C	WRITE(6,211) ID,M,STRESS(1),STRESS(2),STRESS(3),SR(ID,M),	TRANSP	65
C	1 SD(ID,M),ILAYR(M,ID)	TRANSP	66
111	CALL SMATRIX (M,AREA,INO)	TRANSP	67
	IF(INO.LE.0.OR.KBC.GT.0) CALL RMATRIX(M,AREA,ID,INO)	TRANSP	68
	CALL MATADD(M,NCOL,ID)	TRANSP	69
103	CONTINUE	TRANSP	70
C		TRANSP	71
C		TRANSP	72
C	IF(ID.GT.1) GO TO 406	TRANSP	73
C	WRITE(6,403)	TRANSP	74
C 403	FORMAT(/,10X," SD MATRIX ",/)	TRANSP	75
C	WRITE(6,404) (SD(I,I),I=1,NE)	TRANSP	76
C 404	FORMAT(10(1PE12.3))	TRANSP	77
C	WRITE(6,405)	TRANSP	78
C 405	FORMAT(/,10X," R SAVI MATRIX ",/)	TRANSP	79
C	WRITE(6,404) (RSV1(I),I=1,NE)	TRANSP	80
C 406	CONTINUE	TRANSP	81
	RETURN	TRANSP	82
200	FORMAT(/,10X,"ID ELEMENT",3X,"STRESS(1)",4X,"STRESS(2)",4X,	TRANSP	83
1	"STRESS(3)",4X,"SR(ID,M)",4X,"SD(ID,M)",4X,"ILAYR(M,ID)"	TRANSP	84
205	FORMAT(/,1X,I34,"FOR TIME SEGMENT NUMBER ",I4,/)	TRANSP	85
211	FORMAT(10X,I2,3X,I3,3X,1PE11.4,2X,1PE11.4,2X,1PE11.4,2X,1PE11.4,	TRANSP	86
1	2X,1PE11.4,6X,I4)	TRANSP	87

END	TRANSP	88
SUBROUTINE WAVSAN(IO,H,UEXT,QS)	WAVSAN	2
C	WAVSAN	3
C** THIS SUBROUTINE CALCULATES THE SEDIMENT TRANSPORT	WAVSAN	4
C** DUE TO WAVES. THE SOLUTION SCHEME IS BASED ON THE	WAVSAN	5
C** REPORT #SEDIMENT TRANSPORT IN RANDOM WAVES# BY S.S. LIANG AND	WAVSAN	6
C** #SIANG WANG, UNIVERSITY OF DELAWARE, DECEMBER, 1973.	WAVSAN	7
C	WAVSAN	8
CHARACTER*5 ZTYP	FETFIX1	29
LOGICAL WAVE	WAVSAN	10
COMMON/WAVE/ N0DE(240), NC(135), A(135,10), K(135,10),	WAVSAN	11
\$ W(135,10), HB(135), ALPHAC(135), WAVE, N, D	WAVSAN	12
\$, VIS, DPTH(135)	WAVSAN	13
COMMON /ZTYPE/ ZTYP(135)	WAVSAN	14
COMMON/BLK12/ ACOF(6), U(6), V(6), DD(2), AKJ(9,100), ALMBOA,	WAVSAN	15
\$ RHOHAT, AKP(3)	WAVSAN	16
COMMON/BLK15/ ILAYR(100,3), XYSO(100), BOIV(100), NBED(100),	WAVSAN	17
\$ BED(100), RHOSED(3), XNT(100,3), RSAV1(100),	WAVSAN	18
\$ RSAV2(100), RSAV3(100)	WAVSAN	19
DIMENSION UZ(10), BETA(10), E(10), ANGLE(10), FCTR1(10),	WAVSAN	20
\$ FCTR2(10), FCTR3(10), R(10), FCTR4(10)	WAVSAN	21
REAL K,NU	WAVSAN	22
C	WAVSAN	23
C** CHECK TO SEE IF THE WAVE INFLUENCE WILL BE FELT	WAVSAN	24
C** BY THE BOTTOM.	WAVSAN	25
C	WAVSAN	26
PI=ACOS(-1.)	WAVSAN	27
HBAR=0.	WAVSAN	28
NO=NC(N)	WAVSAN	29
DO 5 I=1,NO	WAVSAN	30
5 HBAR=2.*PI/K(N,I)+HBAR	WAVSAN	31
HBAR=HBAR/FLOAT(NO)	WAVSAN	32
IF(H.GT.,HBAR/2.0) RETURN	WAVSAN	33
C	WAVSAN	34
C** THE FOLLOWING SECTION INITIALIZES CONSTANTS	WAVSAN	35
C	WAVSAN	36
UEXT=UEXT/3600./24.	WAVSAN	37
VIS=VIS/3600./24.	WAVSAN	38
SIGMA=5.15	WAVSAN	39
G=9.8	WAVSAN	40
AZ=587.	WAVSAN	41
OMEGA=1.	WAVSAN	42
BSTAR=4.	WAVSAN	43
ETAZ=1./1.5	WAVSAN	44
SPWSED=RHOSED(ID)	WAVSAN	45
SPHWAT=RHOHAT	WAVSAN	46
RHOS=SPWSED/G	WAVSAN	47
RHOH=SPHWAT/G	WAVSAN	48
NU=VIS	WAVSAN	49
B=18.*NU/((RHOS/RHOH+0.5)*D**2)	WAVSAN	50
Y=.35 * D	WAVSAN	51
GAMMA=3./(2.*RHOS/RHOH+1.)	WAVSAN	52
HV=(RHOS/RHOH-1.)*G*D**2/(18.*NU)	WAVSAN	53
USQ=0.	WAVSAN	54
C	WAVSAN	55
C** THE FOLLOWING SECTION INITIALIZES VARIABLES AND	WAVSAN	56
C** CALCULATES U SQUARED.	WAVSAN	57
C	WAVSAN	58
DO 10 I=1,NO	WAVSAN	59
FCTR1(I)=0.	WAVSAN	60
FCTR2(I)=0.	WAVSAN	61
F1=0.	WAVSAN	62

	EPSILON=ATAN(W(N,I)/H)	NAVSAN	63
	BETA(I)=SQRT(W(N,I)/(2.*NU))	NAVSAN	64
	E(I)=133.*SINH(K(N,I)*H)/(A(N,I)*BETA(I)*D)	NAVSAN	65
	ANGLE(I)=0.6*BETA(I)*D	NAVSAN	66
C		NAVSAN	67
C**	CHECK IF EXPONENT IS TOO SMALL.	NAVSAN	68
C		NAVSAN	69
	IF((-2.*D*E(I)).GT. -675.) FCTR1(I)=EXP(-2.*D*E(I))	NAVSAN	70
	IF((-4.*D*E(I)).GT. -675.) FCTR2(I)=EXP(-4.*D*E(I))	NAVSAN	71
	FCTR3(I)=E(I)**2+.09*BETA(I)**2	NAVSAN	72
	FCTR4(I)=E(I)**2-.09*BETA(I)**2	NAVSAN	73
C		NAVSAN	74
C**	FOR SHALLOW WATER CASE ONLY, THE FOLLOWING	NAVSAN	75
C**	EXPRESSION FOR R(I) IS TRUE.	NAVSAN	76
C		NAVSAN	77
	R(I)=WV*H/(GAMMA*SIGMA*A(N,I)*W(N,I))	NAVSAN	78
C		NAVSAN	79
C**	CHECK IF EXPONENT IS TOO SMALL.	NAVSAN	80
C		NAVSAN	81
	IF((-E(I)*Y).GT. -675.) F1=0.5*EXP(-E(I)*Y)	NAVSAN	82
	F2=0.3*BETA(I)*Y	NAVSAN	83
	UZ(I)=A(N,I)*W(N,I)/SINH(K(N,I)*H)	NAVSAN	84
C		NAVSAN	85
C**	CALCULATE U SQUARED	NAVSAN	86
C		NAVSAN	87
	USQ=UZ(I)**2*(1.-2.*F1*COS(F2)+F1**2)+USQ	NAVSAN	88
10	CONTINUE	NAVSAN	89
C		NAVSAN	90
C**	NOW CALCULATE P AND C ZERO	NAVSAN	91
C		NAVSAN	92
	PSI=(RHOS-RH0W)*G*D/(RH0W*USQ)	NAVSAN	93
	ARG=BCSTAR*PSI=1./ETAZ	NAVSAN	94
	CALC ERFC(ARG,P)	NAVSAN	95
	P=.5*P	NAVSAN	96
	GAMMAS=RHOS/RH0W	NAVSAN	97
	CZBAR=AZ*P*GAMMAS	NAVSAN	98
C		NAVSAN	99
C**	NOW CALCULATE THE BEDLOAD TRANSPORT, QB	NAVSAN	100
C**	QBE IS THE TRANSPORT DUE TO THE EXTERNAL VELOCITY, U.	NAVSAN	101
C		NAVSAN	102
	QB=0.	NAVSAN	103
	T1=0.	NAVSAN	104
	T2=0.	NAVSAN	105
	T3=0.	NAVSAN	106
	T4=0.	NAVSAN	107
	DO 20 I=1,NG	NAVSAN	108
	T1A=.5*K(N,I)*UZ(I)**2*FCTR1(I)	NAVSAN	109
	T1A=T1A/(2.*NU*FCTR3(I)**2)	NAVSAN	110
	T1B=2.*D*FCTR4(I)*SIN(ANGLE(I))	NAVSAN	111
	T1C=1.2*E(I)*BETA(I)*D*COS(ANGLE(I))	NAVSAN	112
	T1D=1./FCTR3(I)*(2.55*E(I)**2*BETA(I)-.0945*BETA(I)**3)*COS(ANGLE	NAVSAN	113
	S(I))	NAVSAN	114
	T1E=1./FCTR3(I)*(-.855*E(I)*BETA(I)**2+2.5*E(I)**3)*SIN(ANGLE(I))	NAVSAN	115
C		NAVSAN	116
	T1=T1A*(T1B+T1C+T1D+T1E)+T1	NAVSAN	117
C		NAVSAN	118
	T2A=0.5*K(N,I)*UZ(I)**2/(2.*NU*FCTR3(I))	NAVSAN	119
	T2B=-.0375*BETA(I)/E(I)**2*(FCTR2(I)-1.)	NAVSAN	120
	T2C=.15*BETA(I)*D/E(I)	NAVSAN	121
	T2D=2.4*BETA(I)*E(I)*D/FCTR3(I)	NAVSAN	122
	T2E=1./FCTR3(I)**2	NAVSAN	123
	T2E=T2E*(2.55*BETA(I)*E(I)**2-.0945*BETA(I)**3)	NAVSAN	124

C	T2=T2A*(T2B-T2C+T2D-T2E)+T2	WAVSAN	125
C		WAVSAN	126
C	T3A=K(N,I)*UZ(I)**2/(2.*K(N,I))	WAVSAN	127
C	T3B=D*FCTR1(I)*COS(ANGLE(I))	WAVSAN	128
C	T3C=0.5*E(I)/FCTR3(I)	WAVSAN	129
C	T3C=T3C*FCTR1(I)*COS(ANGLE(I))	WAVSAN	130
C	T3D=0.15*BETA(I)/FCTR3(I)*FCTR1(I)*SIN(ANGLE(I))	WAVSAN	131
C	T3E=.25*FCTR2(I)/(2.*E(I))	WAVSAN	132
C	T3F=2.*D	WAVSAN	133
C	T3G=.5*E(I)/FCTR3(I)	WAVSAN	134
C	T3H=.25/(2.*E(I))	WAVSAN	135
C		WAVSAN	136
C	T3= T3A*(T3B+T3C-T3D-T3E+T3F-T3G+T3H)+T3	WAVSAN	137
C		WAVSAN	138
C	T4A=.5*K(N,I)*UZ(I)**2/(2.*K(N,I)*FCTR3(I))	WAVSAN	139
C	T4B=0.5*E(I)*FCTR1(I)*COS(ANGLE(I))	WAVSAN	140
C	T4C=0.5*0.3*BETA(I)*FCTR1(I)*SIN(ANGLE(I))	WAVSAN	141
C	T4D=0.5/(2.*E(I))*FCTR4(I)*FCTR2(I)	WAVSAN	142
C	T4E=0.5*E(I)	WAVSAN	143
C	T4F=0.5/(2.*E(I))*FCTR4(I)	WAVSAN	144
C		WAVSAN	145
C	T4= T4A*(T4B+T4C-T4D-T4E+T4F)+T4	WAVSAN	146
C		WAVSAN	147
C	20 CONTINUE	WAVSAN	148
C		WAVSAN	149
C	QB=OMEGA*CZBAR*UEXT/(1.+1./7.)*(2.*D)**(1.+1./7.)*H**(-1./7.)	WAVSAN	150
C	QB=OMEGA*CZBAR*(T1+T2+T3+T4)+QB	WAVSAN	151
C		WAVSAN	152
C		WAVSAN	153
C		WAVSAN	154
C**	CONVERT TO KG(FORCE)/DAY=M, 1PPH=1MG(FORCE)/LITER	WAVSAN	155
C		WAVSAN	156
C	QB=QB/1000.*3600.*24.	WAVSAN	157
C		WAVSAN	158
C**	THE FOLLOWING SECTION CALCULATES THE SUSPENDED	WAVSAN	159
C**	SEDIMENT TRANSPORT, QSUS.	WAVSAN	160
C**	QSUSE IS THE SUSPENDED SEDIMENT TRANSPORT DUE TO THE EXTERNAL	WAVSAN	161
C**	VELOCITY, UEXT.	WAVSAN	162
C		WAVSAN	163
C	T1=0.	WAVSAN	164
C	T2=0.	WAVSAN	165
C	T3=0.	WAVSAN	166
C	DO 60 I=1,NO	WAVSAN	167
C	T1A=K(N,I)**2*K(N,I)/(4.*(SINH(K(N,I)*H))**2)	WAVSAN	168
C	T1B=3.*SINH(2.*K(N,I)*H)/(2.*K(N,I)*H)	WAVSAN	169
C	Y=H	WAVSAN	170
C	FLAG=1.	WAVSAN	171
C	T1C=0.	WAVSAN	172
C		WAVSAN	173
C**	CHECK TO SEE IF DENOMINATOR IN T1CA IS ZERO	WAVSAN	174
C		WAVSAN	175
C	21 IF ((R(I)+3.) .EQ. 0.) GO TO 25	WAVSAN	176
C	T1CA=(Y**(R(I)+3.))/(H**2*(R(I)+3.))	WAVSAN	177
C	GO TO 27	WAVSAN	178
C	25 T1CA=ALOG(Y)/(H**2)	WAVSAN	179
C		WAVSAN	180
C**	CHECK TO SEE IF DENOMINATOR IN T1CB IS ZERO	WAVSAN	181
C		WAVSAN	182
C	27 IF ((R(I)+2.) .EQ. 0.) GO TO 28	WAVSAN	183
C	T1CB=(2*Y**(R(I)+2.))/(H*(R(I)+2.))	WAVSAN	184
C	GO TO 29	WAVSAN	185
C	28 T1CB=2./H*ALOG(Y)	WAVSAN	186

29	T1C=FLAG*(T1CA-T1CB)+T1C	WAVSAN	187
	IF(FLAG .LE. 0.) GO TO 26	WAVSAN	188
	Y=2.*D	WAVSAN	189
	FLAG=-1.	WAVSAN	190
	GO TO 21	WAVSAN	191
C		WAVSAN	192
26	T1=T1A*T1B*T1C/((2.*D)**R(I))+T1	WAVSAN	193
C		WAVSAN	194
	T2A=UZ(I)**2*K(N,I)*.5/(2.*NU*FCTR3(I))	WAVSAN	195
	T2B=1.2*BETA(I)*E(I)/FCTR3(I)	WAVSAN	196
	T2C=0.075*BETA(I)/E(I)	WAVSAN	197
	Y=H	WAVSAN	198
	FLAG=1.	WAVSAN	199
	T3A=0.	WAVSAN	200
	T3B=0.	WAVSAN	201
	T3C=0.	WAVSAN	202
C		WAVSAN	203
C**	CHECK TO SEE IF DENOMINATOR OF T3A IS ZERO	WAVSAN	204
C		WAVSAN	205
41	IF((R(I)+3.) .EQ. 0.) GO TO 43	WAVSAN	206
	T3A=FLAG*1.5*Y**R(I)+3.)/(H**2*(R(I)+3.))+T3A	WAVSAN	207
	GO TO 44	WAVSAN	208
43	T3A=FLAG*1.5/(H**2)*ALOG(Y)+T3A	WAVSAN	209
C		WAVSAN	210
C**	CHECK TO SEE IF DENOMINATOR OF T3B IS ZERO	WAVSAN	211
C		WAVSAN	212
44	IF((R(I)+2.) .EQ. 0.) GO TO 45	WAVSAN	213
	T3B=FLAG*3.*Y**R(I)+2.)/(H*(R(I)+2.))+T3B	WAVSAN	214
	GO TO 46	WAVSAN	215
45	T3B=FLAG*3./H*ALOG(Y)+T3B	WAVSAN	216
C		WAVSAN	217
C**	CHECK TO SEE IF DENOMINATOR OF T3C IS ZERO	WAVSAN	218
C		WAVSAN	219
46	IF((R(I)+1.) .EQ. 0.) GO TO 47	WAVSAN	220
	T3C=FLAG*Y**R(I)+1.)/(R(I)+1.))+T3C	WAVSAN	221
	GO TO 48	WAVSAN	222
47	T3C=FLAG*ALOG(Y)+T3C	WAVSAN	223
48	IF (FLAG .LE. 0.) GO TO 42	WAVSAN	224
	Y=2.*D	WAVSAN	225
	FLAG=-1.	WAVSAN	226
	GO TO 41	WAVSAN	227
C		WAVSAN	228
42	T3=(T3A-T3B+T3C)/((2.*D)**R(I))	WAVSAN	229
C		WAVSAN	230
	T2=T2A*(T2B-T2C)*T3+T2	WAVSAN	231
C		WAVSAN	232
C**	THIS SECTION CALCULATES PART OF THE VALUES(Y4) REQUIRED TO	WAVSAN	233
C**	CALCULATE QSUSE.	WAVSAN	234
C		WAVSAN	235
	IF(UEXT .EQ. 0.) GO TO 60	WAVSAN	236
	Y=H	WAVSAN	237
	FLAG=1.	WAVSAN	238
	T4A=(2.*D)**(-R(I))	WAVSAN	239
	T4B=0.	WAVSAN	240
C		WAVSAN	241
C**	CHECK TO SEE IF DENOMINATOR OF T4B IS ZERO.	WAVSAN	242
C		WAVSAN	243
51	IF((1.+1./7.+R(I)) .EQ. 0.) GO TO 53	WAVSAN	244
	T4B=FLAG*Y**R(I)+1.+1./7.+R(I))/(1.+1./7.+R(I))+T4B	WAVSAN	245
	GO TO 55	WAVSAN	246
53	T4B=FLAG*ALOG(Y)+T4B	WAVSAN	247
55	IF(FLAG .LE. 0.) GO TO 57	WAVSAN	248

Y=2.*D	WAVSAN	249
FLAG=-1.	WAVSAN	250
GO TO S1	WAVSAN	251
57 T4=T4A*T4B+T4	WAVSAN	252
60 CONTINUE	WAVSAN	253
C	WAVSAN	254
QSUSE=CZBAR*UEXT/H**(1./7.)*T4	WAVSAN	255
QSUS=CZBAR*(T1+T2)+QSUSE	WAVSAN	256
C	WAVSAN	257
C** CONVERT TO KG(FORCE)/DAY=M	WAVSAN	258
C	WAVSAN	259
QSUS=QSUS/1000.*3600.*24.	WAVSAN	260
C	WAVSAN	261
C** NOW CALCULATE THE TOTAL SEDIMENT TRANSPORT	WAVSAN	262
C	WAVSAN	263
QS=QB+QSUS	WAVSAN	264
C	WAVSAN	265
WRITE (6,1500) QS, QB	WAVSAN	266
C1500 FORMAT (" \$\$\$S QS, QB = "1P2E12.4)	WAVSAN	267
UEXT=UEXT*3600.*24.	WAVSAN	268
VIS=VIS*3600.*24.	WAVSAN	269
RETURN	WAVSAN	270
END	WAVSIM	2
SUBROUTINE WAVSIM(NP, V, H, F)	WAVSIM	3
C	WAVSIM	4
LOGICAL WAVE	WAVSIM	5
COMMON/WAVE/ NODE(240), NC(135), A(135,10), K(135,10),	WAVSIM	6
W(135,10), HB(135), ALPHAC(135), WAVE, N, D, VIS	WAVSIM	7
S ,DPHHS(135)	WAVSIM	8
S COMMON /ZTYPE/ ZTYP(135)	WAVSIM	10
C CHARACTER*5 ZTYP	FETFIX1	30
C REAL K	WAVSIM	11
C I=1	WAVSIM	12
C	WAVSIM	13
C** CHECK FOR V EQUAL 0. THE WAVE GENERATOR WILL	WAVSIM	14
C** NOT ALLOW V TO BE IDENTICALLY 0.	WAVSIM	15
C	WAVSIM	16
IF(V .EQ. 0.) V=.01	WAVSIM	17
C	WAVSIM	18
C** CONVERT TO PROPER UNITS FOR HS AND TS EQ.	WAVSIM	19
C	WAVSIM	20
V=V*3.2808/(24.*3600.)	WAVSIM	21
H=H*3.2808	WAVSIM	22
F=F*3.2808	WAVSIM	23
G=32.2	WAVSIM	24
PI=ACOS(-1.)	WAVSIM	25
T1=G*H/(V**2)	WAVSIM	26
T2=G*F/(V**2)	WAVSIM	27
C	WAVSIM	28
C** CALCULATE THE SIGNIFICANT WAVE HEIGHT	WAVSIM	29
C	WAVSIM	30
HS=.283*V**2/G*TANH(.530*T1**75)*	WAVSIM	31
STANH((.0125*T2**42)/TANH(.530*T1**75))	WAVSIM	32
C	WAVSIM	33
C** USE THE RMS WAVE HEIGHT	WAVSIM	34
C	WAVSIM	35
HRMS=HS/1.416	WAVSIM	36
C	WAVSIM	37
C** CONVERT TO METRIC UNITS	WAVSIM	38
C	WAVSIM	39
HRMS=HRMS/3.2808	WAVSIM	40
A(NP, I)=HRMS/2.	WAVSIM	41
C	WAVSIM	

C** CALCULATE THE SIGNIFICANT WAVE PERIOD
C

TS=1.2*2.*PI*V/G*TANH(.833*T1**375)*
STANH((.077*T2**25)/TANH(.833*T1**375))
W(NP,I)=2.*PI/TS
G=9.8
V=V/3.2808*24.*3600.
H=H/3.2808
F=F/3.2808
K(NP,I)=SOL(H,G,W(NP,I),0.0001)
RETURN
END

HAVSIM 42
HAVSIM 43
HAVSIM 44
HAVSIM 45
HAVSIM 46
HAVSIM 47
HAVSIM 48
HAVSIM 49
HAVSIM 50
HAVSIM 51
HAVSIM 52
HAVSIM 53

DISTRIBUTION

No. of
Copies

No. of
Copies

OFFSITE

ONSITE

U.S. Nuclear Regulatory
Commission
Division of Technical
Information and
Document Control
7920 Norfolk Avenue
Bethesda, MD 20014

P. R. Reed
U.S. Nuclear Regulatory
Commission
Washington, DC 20555

R. B. Codell
U.S. Nuclear Regulatory
Commission
Washington, DC 20555

F. Swanberg
Office of Nuclear
Regulatory Research
U.S. Nuclear Regulatory
Commission
Washington, DC 20555

E. F. Conti
Office of Nuclear
Regulatory Research
U.S. Nuclear Regulatory
Commission
Washington, DC 20555

C. V. Alonso
U.S. Department of Agriculture
Sedimentation Laboratory
Southern Region
Agriculture Research Center
P.O. Box 1157
Oxford, MI 38655

50 Pacific Northwest Laboratory

D. B. Cearlock
D. W. Dragnich
P. C. Hays
E. A. Jenne
Y. Onishi (32)
W. L. Templeton
F. L. Thompson (4)
D. S. Trent
L. L. Wendell
Technical Information (5)
Publishing Coordination (2)

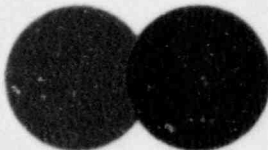
NRC FORM 335 U.S. NUCLEAR REGULATORY COMMISSION BIBLIOGRAPHIC DATA SHEET		1. REPORT NUMBER (Assigned by DDC) NUREG/CR-2424, Vol. 2 PNL-5088	
4. TITLE AND SUBTITLE (Add Volume No., if appropriate) Volume 2: Users Manual and Computer Program Listing for FETRA		2. (Leave blank)	
7. AUTHOR(S) Onishi F. L. Thompson		3. RECIPIENT'S ACCESSION NO. 5. DATE REPORT COMPLETED MONTH: April YEAR: 1984	
9. PERFORMING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code) Pacific Northwest Laboratory P.O. Box 999 Richland, WA 99352		DATE REPORT ISSUED MONTH: May YEAR: 1984	
12. SPONSORING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code) Division of Health, Siting and Waste Management Office of Nuclear Regulatory Research U.S. Nuclear Regulatory Commission Washington, DC		6. (Leave blank) 8. (Leave blank)	
13. TYPE OF REPORT Topical		10. PROJECT/TASK/WORK UNIT NO. 11. FIN NO. B2271	
15. SUPPLEMENTARY NOTES		14. (Leave blank)	
16. ABSTRACT (200 words or less) FETRA is a finite element model for simulating the sediment and contaminant transport to surface water. The model was applied to a test site in the Irish Sea and modified to account for wave mechanisms that affect sediment suspension. Volume 2 of this report presents a very brief users guide for FETRA and a computer program listing of the model.			
17. KEY WORDS AND DOCUMENT ANALYSIS Computer Code for Ocean/Marine Environment Sediment/Radionuclide Transport Model FETRA		17a. DESCRIPTORS	
17b. IDENTIFIERS (OPEN-ENDED TERMS)			
18. AVAILABILITY STATEMENT Unlimited		19. SECURITY CLASS (This report) Unclassified	21. NO. OF PAGES 5
		20. SECURITY CLASS (This paper)	22. PRICE

UNITED STATES
NUCLEAR REGULATORY COMMISSION
WASHINGTON, D.C. 20555

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, \$300

FOURTH CLASS MAIL
POSTAGE & FEES PAID
USNRC
WASH. D. C.
PERMIT No. 662

NI1389-00003 2
USNRC
JO ANN RESNER
W 501



NUREG/CR-2424, Vol. 2

MATHEMATICAL SIMULATION OF SEDIMENT AND RADIONUCLIDE TRANSPORT
IN COASTAL WATERS

MAY 1984