COMPUTER PROGRAMS FOR SEDIMENT TRANSPORT

Documentation and Listing

by

Khalid Mahmood

and

V. Miguel Ponce

Colorado State University
Engineering Research Center
Fort Collins, Colorado
80523

August 1975

CER75-76KM-VMP2

COMPUTER PROGRAMS FOR SEDIMENT TRANSPORT

TABLE OF CONTENTS

PREFACE

This manual presents FORTRAN programs for the computation of sediment transport by the following methods: 1) Einstein's Bed-Load Function, 2) Mahmood's Bed Material Transport Function, 3) Colby's Bed Material Load Method, 4) Meyer-Peter and Muller Bed-Load Equation, 5) Modified Einstein Procedure. The objective of the manual is to make these programs available for use as canned programs. For a particular program and specified input, appropriate quantities concerning the sediment load are obtained as the output. Input and output formats are illustrated in each case by examples.

The theoretical basis for the sediment transport methods covered herein varies widely, and so does their scope and range of applicability. Salient features of the methods are given as brief introductions to each chapter. It is assumed that the user has a working familiarity with these methods, their scope and limitations. References to original publications containing the theoretical developments and computational steps as used in the programs are provided in each case.

The computer programs included herein were developed for use in various research studies. The programming approach is therefore one of simplicity and ease of modification rather than of economy in compilation or processing time. The programs are written in FORTRAN IV language and have been extensively tested on the CDC 6400 computer at Colorado State University using SCOPE 3.3.

ACKNOWLEDGEMENTS

The programs included in this manual were developed over a period of time by the following persons: Einstein's Bed-Load Function, K. Mahmood and S. A. Rana; Mahmood's Bed-Material Transport Function, K. Mahmood; Colby's Bed Material Load Method, K. Mahmood; Meyer-Peter and Muller, V. M. Ponce; Modified Einstein Procedure, V. M. Ponce and T. Masood. The development of some of these programs and the present compilation is part of a continuing research effort in alluvial river mechanics sponsored by the National Science Foundation Grants No: ENG72-00274 A01 and OIP75-15976, with K. Mahmood as the Principal Investigator. Linda Koshio assisted in the preparation of the final copy of this report.

I. EINSTIN

(Einstein's Bed-Load Function)

1.0 Introduction

Program EINSTIN computes the bed material load and its size distribution in alluvial channels, based on Einstein's bed-load function [1]*. The original method, as developed by Einstein, encompasses both the resistance and transport function. The former yields a hydraulic solution that is used to evaluate grain-associated shear parameters. To be consistent with the scope of other methods presented herein, Program EINSTIN assumes that the resistance to flow has been evaluated separately in a previous step. Therefore, input to EINSTIN consists of mean velocity, depth of flow and the energy gradient.

The evaluation of integrals is made by Simpson's rule, using a variable discretization interval. The bed material is assumed to be lognormally-distributed [2], with known values of parameters D_{50}, the median size, and σ, the gradation coefficient. The bed material is divided into ten fractions, and the limit and mean sizes for each fraction are calculated. The fractions are of equal probability of occurrence; therefore, the sizes do not correspond to standard sieve sizes. To calculate the fractions corresponding to sieve sizes, the lognormal distribution can be used.

1.1 Input-Output Description and Examples

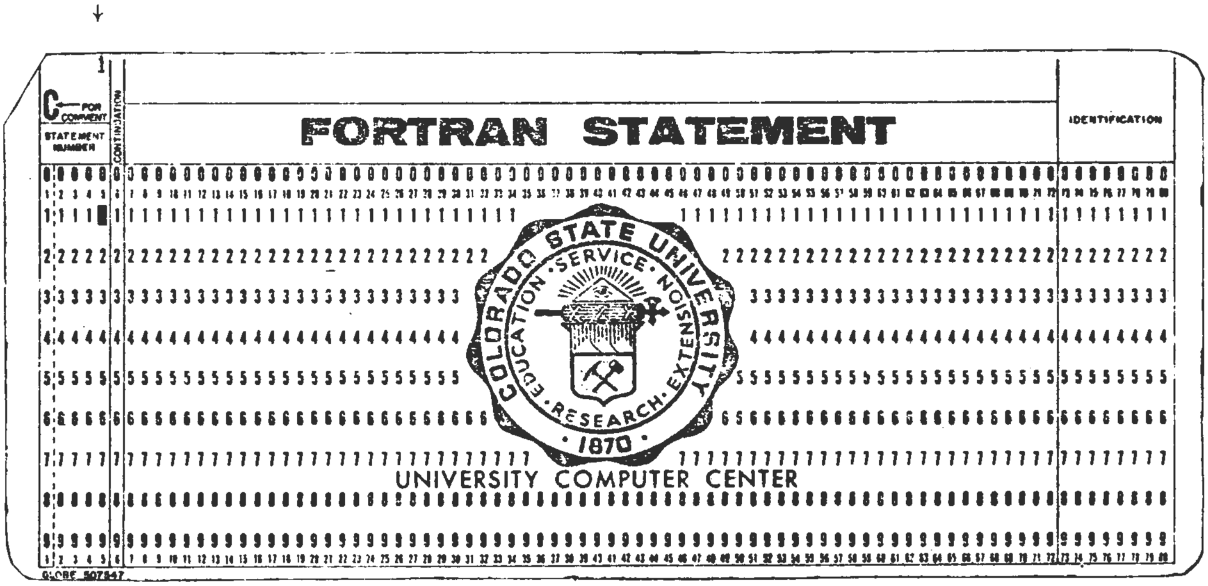
The first card in the input logical record should contain the value of NDATA, in Format I5. NDATA is the number of sets of input data to be fed to the computer at a time. A set of input data consists

---

*Numbers within brackets refer to the list of references.

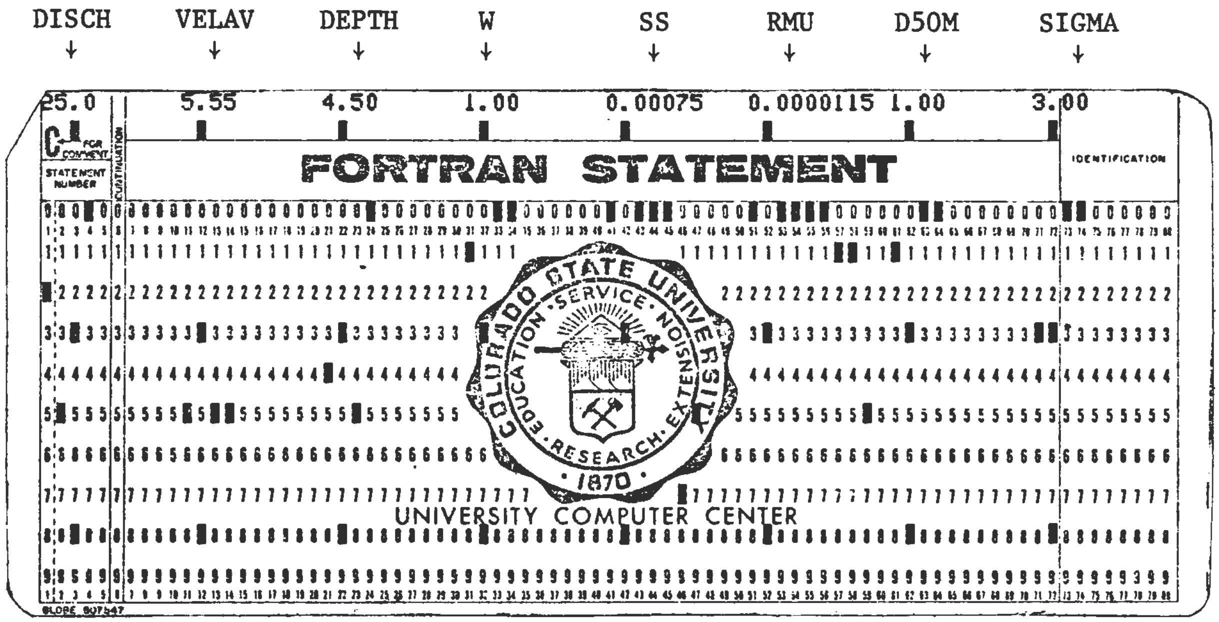
of a group of variables necessary to specify a problem, as detailed below.

NDATA



The first card in input is followed by the sets of input data, to be punched in format 8F10.0. A set of input data consists of the following variables relating to the flow in a cross-section.

VARIABLE	FORTRAN NAME	UNITS
1) Water Discharge	DISCH	cubic ft. per sec.
2) Average Velocity	VELAV	ft. per sec.
3) Hydraulic Depth	DEPTH	ft.
4) Water Surface Width	W	ft.
5) Energy Gradient	SS	ft. per ft.
6) Kinematic Viscosity	RMU	sq. ft. per sec.
7) Median Bed Material Size	D50M	millimeters
8) Gradation Coefficient	SIGMA	no units



Output consists of the input variables and the calculated quantities in five columns, as follows:

1) Fraction Number
2) Geometric Mean Size, in mm.
3) Bed Load, in Tons/day
4) Suspended Bed Material Load, in Tons/day
5) Total Bed Material Load, in Tons/day

A sample output follows.

COMPUTATION OF TOTAL BED MATERIAL
LOAD BY THE EINSTEIN BED-LOAD FUNCTION

WATER DISCHARGE	25.00	C.F.S.
AVERAGE VELOCITY	5.55	FT./SEC.
HYDRAULIC DEPTH	4.50	FT.
WATER SURFACE WIDTH	1.00	FT.
ENERGY GRADIENT	.0007500	FT./FT.
KINEMATIC VISCOSITY	.0000115	SQ.FT./SEC.
MEDIAN BED MATERIAL SIZE	1.00	MM.
GRADATION COEFFICIENT	3.00	

FRACTION NO.	GEO MEAN SIZE (MM)	BED LOAD (TONS/DAY)	SUS BED MAT LOAD (TONS/DAY)	BED MAT LOAD (TONS/DAY)
1	.19999	.00055	.00767	.00822
2	.31036	.01303	.03579	.04882
3	.47116	.14905	.13625	.28531
4	.65151	.70281	.44072	1.14353
5	.86929	1.84360	.86016	2.70377
6	1.15036	2.82378	.71100	3.53478
7	1.53489	3.27829	.15823	3.43652
8	2.12240	3.31282	0.00000	3.31282
9	3.22206	2.44739	0.00000	2.44739
10	5.00015	1.17852	0.00000	1.17852

TOTAL BED LOAD	15.7499	TONS/DAY
TOTAL SUSPENDED BED MATERIAL LOAD	2.3498	TONS/DAY
TOTAL BED MATERIAL LOAD	18.0997	TONS/DAY

1.2 Fortran Names for Input and Output Variables

INPUT

Water Discharge	DISCH
Average Velocity	VELAV
Hydraulic Depth	DEPTH
Water Surface Width	W
Energy Gradient	SS
Kinematic Viscosity	RMU
Median Bed Material Size	D50
Gradation Coefficient	SIGMA

OUTPUT

Fraction Number	I
Geometric Mean Size	X(I)
Bed Load	BD(I)
Suspended Bed Material Load	SD(I)
Total Bed Material Load	TSD(I)

II. STRANS

(Mahmood's Bed Material Transport Function)

2.0 Introduction

Program STRANS computes the total bed-material load in sand-bed channels for specified average velocity, hydraulic depth, energy gradient, kinematic viscosity, median bed-material size and gradation coefficient. The program is based on Chapter IX of reference [3]. Briefly, this transport function uses a two-layer model of flow in sand-bed channels: an inner layer where the shear stress is a constant, and an outer layer where it varies linearly. The rest of the phenomenological structure of this method is the same as that of Einstein's bed-load function.

Program STRANS assumes that the resistance problem has been separately solved so that the velocity, depth and energy gradient are available. The end product of this method is the amount of bed-material transport as well as its size distribution. In general, when 5 or 10 size fractions are used the smallest size fraction may correspond to the wash load size and should be exluded from the bed-material load. The cutoff size for this limit is left to the needs and judgment of the user. Analysis of flume data [3] has shown that the size distribution of sand size sediment load is more closely approximated by this method than by Einstein's bed-load function.

Program STRANS reads in the relevant data for digitized curves and functions, as well as the standard normal distribution for analyzing the size distribution of the transport based on the lognormal distribution. The hydraulic and sediment variables are also read as part of the input data. The main analysis is carries out in subroutine TPORT.

The transport function is designed for five size fractions. Program STRANS provides a choice of 1, 5 or 10 size fractions, depending on the value of NN, fed as input. These fractions correspond to equal probability of occurrence and their limit sizes do not correspond to standard sieve sizes.

The bed-material size and the bed-load in most sand-bed channels are lognormally distributed. With this distribution, two parameters, the median size D_{50} and the gradient coefficient σ are sufficient to describe the size distributions and to calculate the mean sizes of various size fractions. The size distribution bed-material fractions as well as in the transport is assumed lognormal.

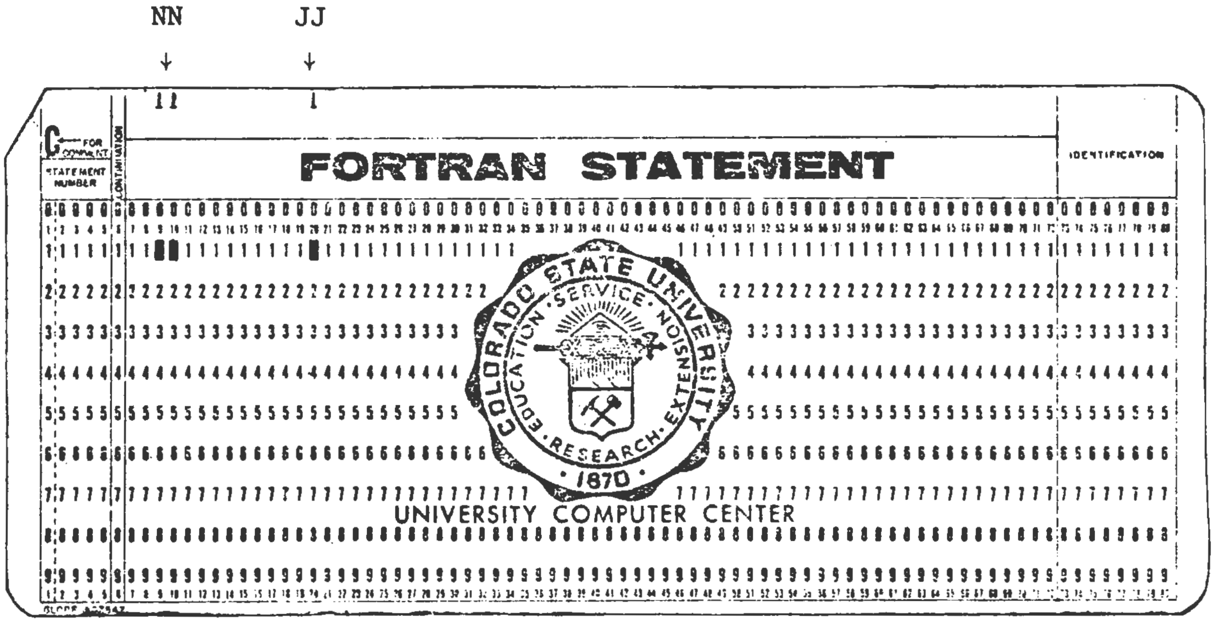
2.1 Input-Output Description and Examples

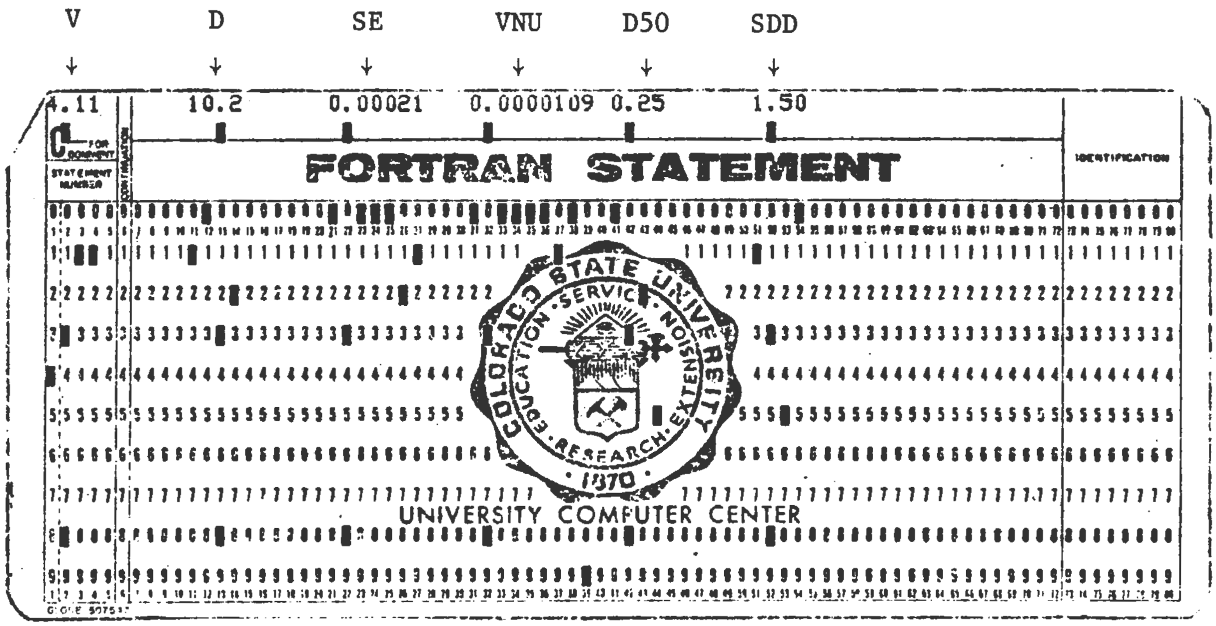
Input consists of the following, in the order shown.

1) Integers NN and JJ, to be read in format 2I10. NN is an input indicator. If NN=2, the number of size fractions is 1. If NN=6, the number of size fractions is 5. If NN=11, the number of size fractions is 10. JJ is an output indicator. If JJ=1, intermediate results are printed out. If JJ=2, intermediate results are omitted from the output.

2) Input variables V, D, SE, VNU, D50, SDD, to be read in format 6F10.0. V is the average velocity in ft. per sec., D is the hydraulic depth in ft., SE is the energy gradient in ft. per ft., VNU is kinematic viscosity in sq. ft. per sec., D_{50} is the median bed material size in mm., SDD is the gradation coefficient sigma.

Following is an example of the input logical record.





A sample output follows.

COMPUTATION OF TOTAL BED MATERIAL DISCHARGE BY MAHMOODS TRANSPORT FUNCTION

AVERAGE VELOCITY	4.11	FT./SEC.
HYDRAULIC DEPTH	10.20	FT.
WATER SURFACE WIDTH	1.00	FT.
ENERGY GRADIENT	.0002100	FT./FT.
KINEMATIC VISCOSITY	.0000109	SQ.FT./SEC.
MEDIAN BED MATERIAL SIZE	.25	MM.
GRADATION COEFFICIENT	1.50	

FRACTION-WISE VALUES OF COMPUTATIONAL PARAMETERS ARE AS FOLLOWS

F NO.	SIZE-FT	ETA=DM /D	ROUSE NO.	PARAM. A1	PARAM. B1	PARAM. C1	PARAM. D1	INTEGRAL I	INTEGRAL J	DEL. TAU
1	.171E-02	.334E-03	.200E+01	.417E-02	-.233E-02	.718E+01	.151E-06	.273E+01	-.397E+01	.958E+00
2	.126E-02	.247E-03	.158E+01	.522E-02	-.281E-02	.718E+01	.261E-05	.173E+01	-.235E+01	.966E+00
3	.108E-02	.212E-03	.138E+01	.648E-02	-.331E-02	.718E+01	.111E-04	.142E+01	-.185E+01	.969E+00
4	.960E-03	.188E-03	.122E+01	.819E-02	-.395E-02	.718E+01	.338E-04	.125E+01	-.156E+01	.971E+00
5	.863E-03	.169E-03	.109E+01	.107E-01	-.481E-02	.718E+01	.900E-04	.112E+01	-.136E+01	.972E+00
6	.779E-03	.153E-03	.973E+00	.145E-01	-.606E-02	.718E+01	.227E-03	.103E+01	-.120E+01	.973E+00
7	.701E-03	.137E-03	.855E+00	.209E-01	-.802E-02	.718E+01	.574E-03	.956E+00	-.107E+01	.974E+00
8	.622E-03	.122E-03	.730E+00	.329E-01	-.115E-01	.718E+01	.156E-02	.892E+00	-.951E+00	.975E+00
9	.533E-03	.105E-03	.586E+00	.610E-01	-.191E-01	.718E+01	.511E-02	.838E+00	-.838E+00	.976E+00
10	.394E-03	.773E-04	.360E+00	.191E+00	-.518E-01	.718E+01	.351E-01	.797E+00	-.702E+00	.977E+00

FRACTION-WISE ANALYSIS IS AS FOLLOWS

F NO.	SIZE-MM	FALL VEL	ROUSE NO.	CRIT.SHEAR	WEIGH.FACT	DEL.SUSP.	DEL. BED	FRAC.IN GS	FRAC.IN GB	FRAC.IN GT
1	.520E+00	.210E+00	.200E+01	.557E-02	.958E+00	.893E-03	.736E-03	.266E-02	.217E+00	.480E-02
2	.385E+00	.166E+00	.158E+01	.460E-02	.966E+00	.121E-02	.512E-03	.360E-02	.151E+00	.507E-02
3	.330E+00	.144E+00	.138E+01	.418E-02	.969E+00	.165E-02	.424E-03	.491E-02	.125E+00	.611E-02
4	.293E+00	.129E+00	.122E+01	.387E-02	.971E+00	.234E-02	.366E-03	.696E-02	.108E+00	.797E-02
5	.263E+00	.115E+00	.109E+01	.369E-02	.972E+00	.349E-02	.321E-03	.104E-01	.947E-01	.112E-01
6	.238E+00	.102E+00	.973E+00	.359E-02	.973E+00	.556E-02	.282E-03	.166E-01	.832E-01	.172E-01
7	.214E+00	.898E-01	.855E+00	.350E-02	.974E+00	.966E-02	.247E-03	.288E-01	.728E-01	.292E-01
8	.190E+00	.767E-01	.730E+00	.340E-02	.975E+00	.190E-01	.212E-03	.567E-01	.625E-01	.567E-01
9	.163E+00	.616E-01	.586E+00	.327E-02	.976E+00	.471E-01	.174E-03	.140E+00	.513E-01	.139E+00
10	.120E+00	.378E-01	.360E+00	.303E-02	.977E+00	.245E+00	.117E-03	.729E+00	.346E-01	.722E+00

F NO.	SIZE-FT	PERCENT FINER THAN	SIZE-MM	PPM IN FRAC	PPM IN FINER THAN
1	.171E-02	100.00	.520	.918E+02	.191E+05
2	.126E-02	99.52	.385	.970E+02	.190E+05
3	.108E-02	99.01	.330	.117E+03	.189E+05
4	.960E-03	98.40	.293	.152E+03	.188E+05
5	.863E-03	97.60	.263	.215E+03	.187E+05
6	.779E-03	96.48	.238	.329E+03	.184E+05
7	.701E-03	94.76	.214	.559E+03	.181E+05
8	.622E-03	91.83	.190	.108E+04	.176E+05
9	.533E-03	86.16	.163	.266E+04	.165E+05
10	.394E-03	72.23	.120	.138E+05	.138E+05

FOR B.M. TRANSP. D84= .1541 MM, D50= .0842 MM D16= .0460 MM AND SIGMA= 1.8311

2.2 Fortran Names for Input and Output Variables

INPUT

Input Indicator	NN
Output Indicator	JJ
Average Velocity	V
Hydraulic Depth	D
Energy Gradient	SE
Kinematic Viscosity	VNU
Median Bed Material Size	D50
Gradation Coefficient	SDD

OUTPUT

Geometric Mean Size in a fraction in ft.	DM
Geometric Mean Size in a fraction in mm.	DMM
Dimensionless Distance from Mean Bed	ETA1
Rouse Number	H
Parameters of Computation (see reference [3])	A1, B1, C1, D1
Integral I and J (see reference [3])	XI1, XI2
Percent finer than specified size fraction by weight	PFG
Bed Material Load in a specified size fraction by parts per million	PPMF
Cumulative Bed Material Concentration (in ppm) finer than a specified size fraction	PPMFT

III. COLBY

(Colby's Bed Material Load Method)

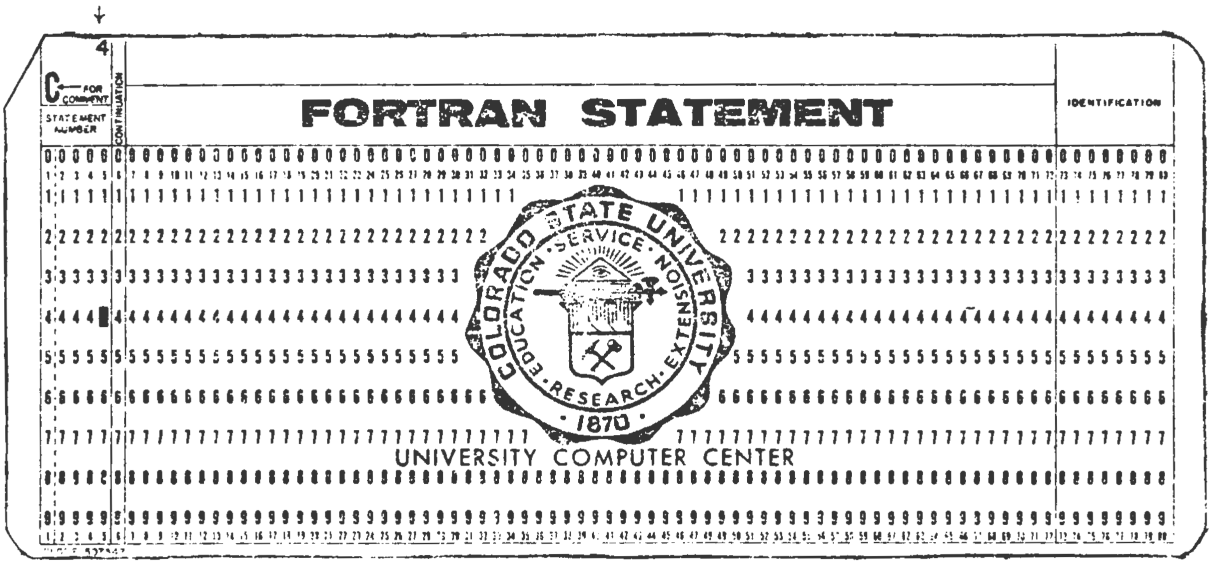
3.0 Introduction

Program COLBY computes bed-material load by Colby's Method [4]. Data input consists of average velocity (ft. per sec.), hydraulic depth (ft.), water surface width (ft.), temperature (°F.), median bed material size (mm.) and fine material concentration (ppm). A remark included as part of the output indicates whether the computations were carried out in a normal fashion, or if one or more variables were out of the value range specified in this method. If velocity, depth or bed-material size are out of range, the program fails to give any results. If temperature or fine material concentration are out of range, the program extrapolates and gives a result, albeit of limited value.

3.1 Input-Output Description and Examples

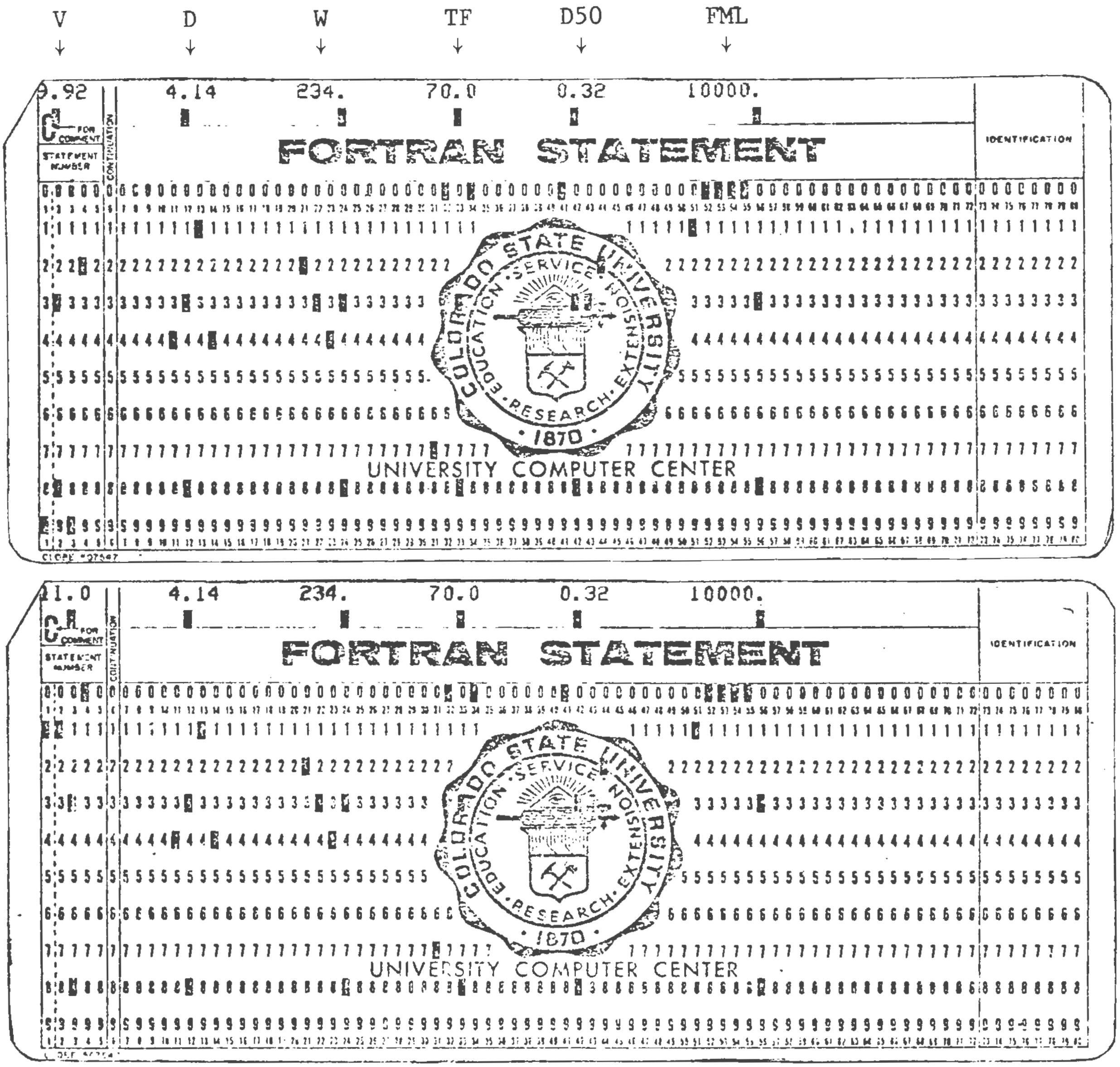
The first card in the input logical record should contain the value of NDATA, in format I5. NDATA is the number of sets of input data to be fed to the computer at a time. A set of input data consists of a group of variables necessary to specify a problem, as detailed below.

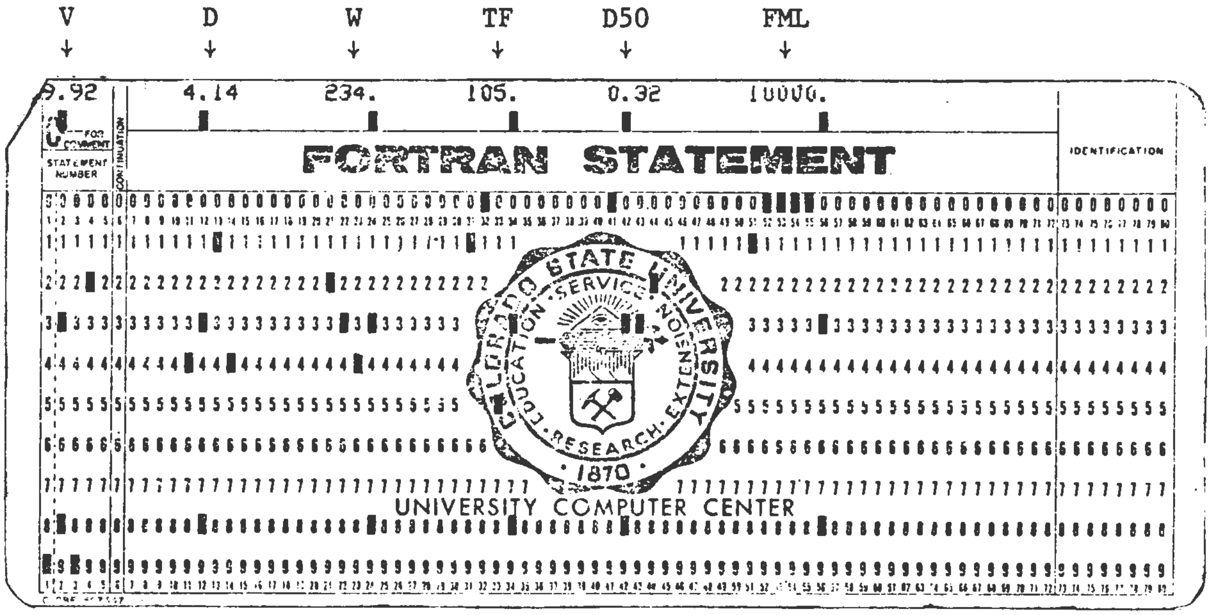
NDATA

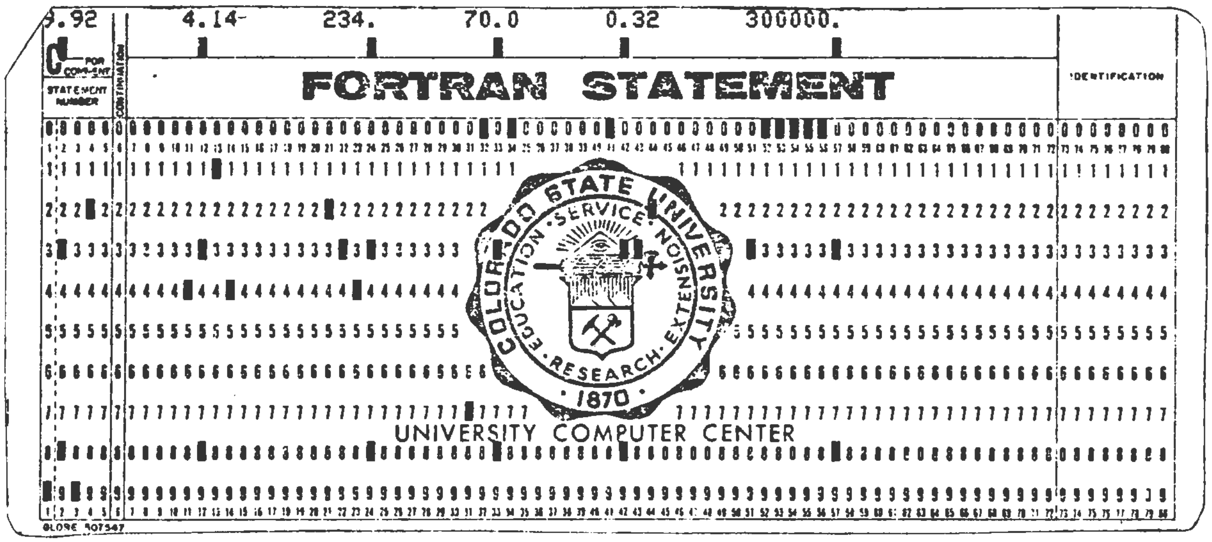


The first card in input is followed by the sets of input data, to be punched in format 6F10.0. A set of input data consists of the following variables, relating to a channel cross-section.

	VARIABLES	FORTRAN NAME	UNITS
1)	Average Velocity	V	ft. per sec.
2)	Hydraulic Depth	D	ft.
3)	Water Surface Width	W	ft.
4)	Temperature	TF	°F
5)	Median Bed Material Size	D50	mm.
6)	Fine Material Concentration	FML	ppm







Output consists of the total bed material transport in Tons/day, and a remark on how the computations were carried out. If REMARK=OK, the computations were carried out successfully. If REMARK=OOR, velocity, depth or bed material size is out of range. If REMARK=TOOR, temperature is out of range. If REMARK=FOOR, fine material concentration is out of range.

VARIABLE	RANGE
Average Velocity	1-10 ft. per sec.
Hydraulic Depth	1-100 ft.
Temperature	32-100 ^{o}F.
Median Bed Material Size	0.1-0.8 mm.
Fine Material Concentration	0-200000 ppm.

A sample output follows.

```
     COMPUTATION OF TOTAL BED MATERIAL
     TRANSPORT BY COLBYS METHOD


SET      1
AVERAGE VELOCITY                           9.92 FT./SEC.
HYDRAULIC DEPTH                            4.14 FT.
WATER SURFACE WIDTH                      234.00 FT.
TEMPERATURE                               70.00 DEG.FAHREN.
MEDIAN BED MATERIAL SIZE                    .32 MM.
FINE MATERIAL CONCENTRATION            10000.00 PPM.

BED MATERIAL TRANSPORT =    76173.08304 TONS/DAY
REMARK = OK


SET      2
AVERAGE VELOCITY                          11.00 FT./SEC.
HYDRAULIC DEPTH                            4.14 FT.
WATER SURFACE WIDTH                      234.00 FT.
TEMPERATURE                               70.00 DEG.FAHREN.
MEDIAN BED MATERIAL SIZE                    .32 MM.
FINE MATERIAL CONCENTRATION            10000.00 PPM.

COMPUTATIONS COULD NOT BE CARRIED OUT
DUE TO DATA OUT OF RANGE
REMARK= OOR


SET      3
AVERAGE VELOCITY                           9.92 FT./SEC.
HYDRAULIC DEPTH                            4.14 FT.
WATER SURFACE WIDTH                      234.00 FT.
TEMPERATURE                              105.00 DEG.FAHREN.
MEDIAN BED MATERIAL SIZE                    .32 MM.
FINE MATERIAL CONCENTRATION            10000.00 PPM.

BED MATERIAL TRANSPORT =    59231.54605 TONS/DAY
REMARK = TOOR


SET      4
AVERAGE VELOCITY                           9.92 FT./SEC.
HYDRAULIC DEPTH                            4.14 FT.
WATER SURFACE WIDTH                      234.00 FT.
TEMPERATURE                               70.00 DEG.FAHREN.
MEDIAN BED MATERIAL SIZE                    .32 MM.
FINE MATERIAL CONCENTRATION           300000.00 PPM.

BED MATERIAL TRANSPORT =   810518.47909 TONS/DAY
REMARK = FOOR
```

3.2 Fortran Names for Input and Output Variables

VARIABLE	FORTRAN NAME
Average Velocity	V
Hydraulic Depth	D
Water Surface Width	W
Temperature	TF
Median Bed Material Size	D50
Fine Material Concentration	FML
Bed Material Transport	GT

IV. MEYER

(Meyer-Peter and Muller Bed-Load Equation)

4.0 Introduction

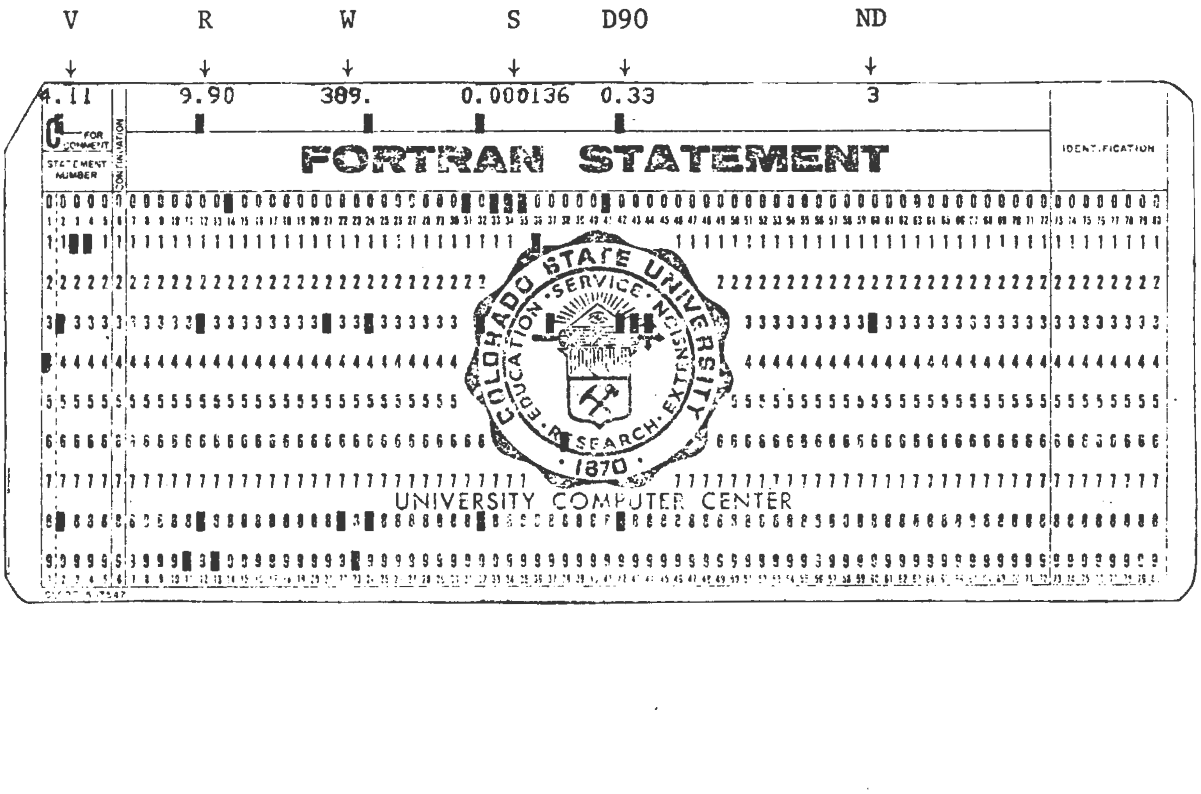
Program MEYER is based on reference [5]. It calculates bed-load transport in Tons/day by the Meyer-Peter and Muller formula. The required Input data are the average velocity, hydraulic radius, water surface width, energy gradient and D_{90} for the bed material.

4.1 Input-Output Description and Examples

Input consists of the following:

1) Variables V, R, W, S, D90, ND, to be read in format 5F10.0, I10.

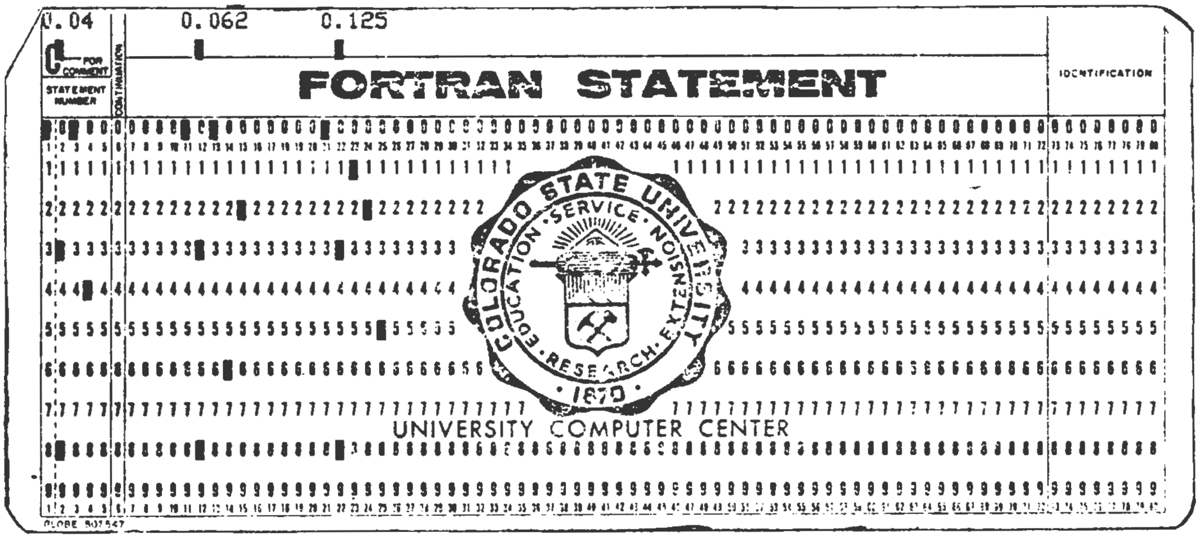
VARIABLE	FORTRAN NAME	UNITS
Average Velocity	V	ft. per sec.
Hydraulic Radius	R	ft.
Water Surface Width	W	ft.
Energy Gradient	S	ft. per ft.
D_{90}	D90	mm.
Number of Fractions in Bed Material	ND	



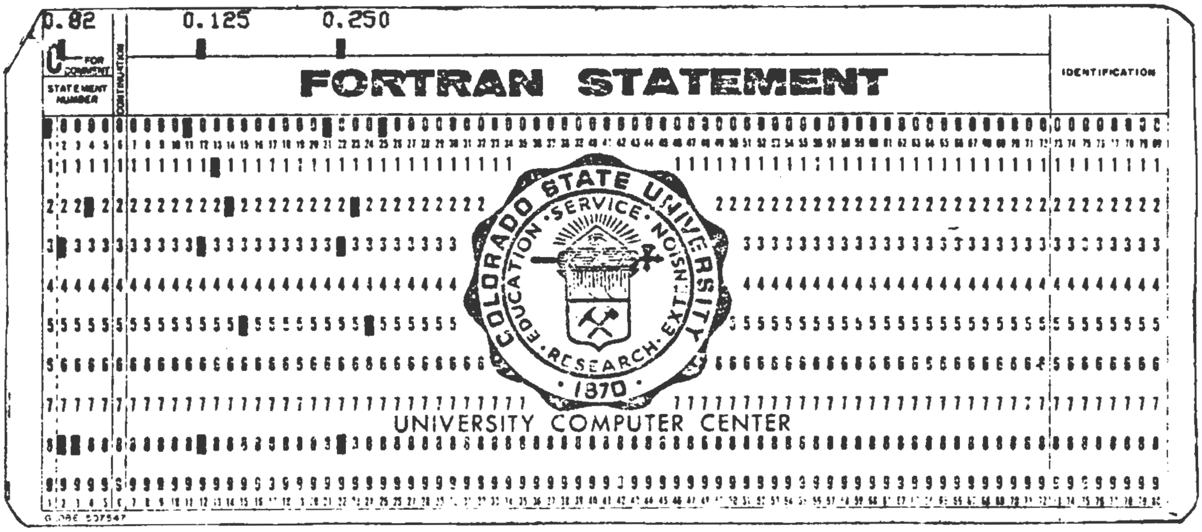
2) Arrays FB(ND), DRL(ND), DRU(ND), to be read in format 3F10.0.

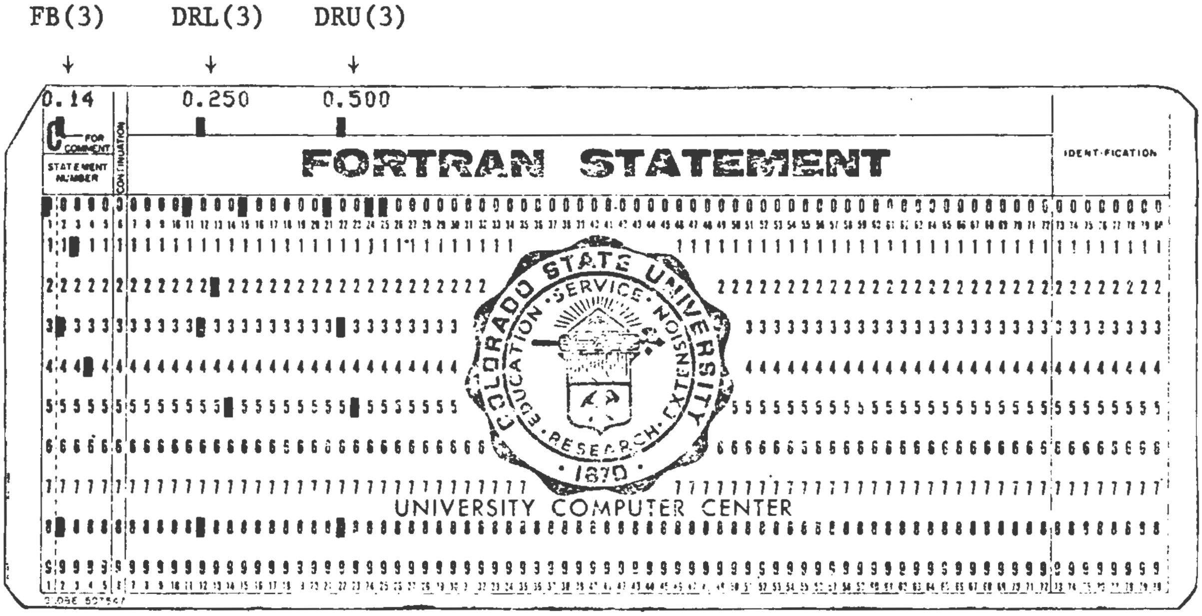
VARIABLE	FORTRAN NAME	UNITS
Fraction of Bed Material in Size Fraction	FB(J)	
Lower Limit of Size Fraction	DRL(J)	mm.
Upper Limit of Size Fraction	DRU(J)	mm.

FB(1) DRL(1) DRU(1)



FB(2) DRL(2) DRU(2)




FB(3)
DRL(3)
DRU(3)
0.14
0.250
0.500
FORTRAN STATEMENT
STATEMENT NUMBER
CONTINUATION
IDENTIFICATION
COLORADO STATE UNIVERSITY
EDUCATION · SERVICE · RESEARCH · EXTENSION
1870
UNIVERSITY COMPUTER CENTER

A sample output follows.

```
        COMPUTATION OF BED LOAD TRANSPORT
        BY MEYER-PETER AND MULLER FORMULA(1948)

    AVERAGE VELOCITY                      4.11 FT./SEC.
    HYDRAULIC RADIUS                      9.90 FT.
    WATER SURFACE WIDTH                 389.00 FT.
    ENERGY GRADIENT                   .0001360 FT./FT.
    DIAMETER 90 PERCENT FINER             .330 MM.

        J       FB(J)     DRL(J)     DRU(J)

        1       .04       .0620      .1250
        2       .82       .1250      .2500
        3       .14       .2500      .5000

    TOTAL BED LOAD TRANSPORT=        733.682 TONS/DAY
```

4.2 Fortran Names for Input and Output Variables

VARIABLE	FORTRAN NAME
Average Velocity	V
Hydraulic Radius	R
Water Surface Width	W
Energy Gradient	S
Diameter for 90% Finer	D90
No of Fractions in Bed Material	ND
Fraction of Bed Material in Size Fraction	FB(J)
Lower Limit of Size Fraction, in mm.	DRL(J)
Upper Limit of Size Fraction, in mm.	DRU(J)
Bed-Load Transport	GS

V. MODEINS

(Modified Einstein Procedure)

5.0 Introduction

Program MODEINS computes the total sediment load and its size distribution in sandbed channels. The procedure used is the Modified Einstein Procedure (MEP) developed by the U.S. Geological Survey [6] and the U.S. Bureau of Reclamation [7,8]. Essentially, the MEP is based on the direct measurement of hydraulic quantities, bed-material size and the suspended load (except within a small distance near the bed) in an alluvial channel. The procedure extrapolates the sediment discharge in the unmeasured zone, thus calculating the total sediment load. The MEP has the same phenomenological structure as Einstein's bed-load function [1] with some modifications in the empirical components.

The sediment load computation by the MEP is more accurate than by other computational methods, mainly because the MEP is based on the direct measurement of the hydraulic and sediment transport quantities. This is especially true in sandbed channels where a large proportion of the total sediment load is transported in the sampled zone and is actually measured. The MEP is only applicable where the basic hydraulic and sedimentation parameters have been measure in the field.

Program MODEINS basically follows the computational procedure outlined in reference [7]. Major deviations consist of the following: 1) the integral functions are evaluated by numerical integration using Simpson's rule with a variable discretization interval, and 2) the extrapolation of the Rouse number for fractions other than the reference size is based on reference [8].

5.1 Input and Output Description and Examples

MODEINS can be set up to read and analyze as many runs as needed. Also with each series of runs analyzed at one time, the program provides an option to use either the 1:2 ratio sieve sizes of reference [7] or any other series specified by the user. The output can be limited to the sedimentation quantities related to total load, or extended to print additional hydraulic parameters and intermediate computational values. The details of input and output controls are given in the following.

The first card in the input logical record should contain the value of NDATA, in format I5. NDATA is the number of sets of input data to be analyzed at one time. A set of input data consists of a group of variables relating to one observation as detailed below. Note that an observation may relate to the sediment load computation in the whole of the cross-section or the load in a segment or on a vertical as the case may be.

The first card is to be followed by individual sets of input data, each one consisting of the following, in the order shown.

1) GENERAL DATA: 13 variables to be punched in format 8F10.0.

VARIABLES	FORTRAN NAME	UNITS
Water Discharge	DISCH	cubic ft. per sec.
Average Velocity	UAVE	ft. per sec.
Hydraulic Depth	DEPTH	ft.
Water Surface Width	W	ft.
Area of Cross-Section	AREA	sq. ft.
Temperature	TEMP	^{o}F.
Kinematic Viscosity	XNU	sq. ft. per sec.
65 Percent Finer Diameter for Bed Material	D65	ft.
35 Percent Finer Diameter for Bed Material	D35	ft.
Average Concentration	CONC	ppm.
Sampled Suspended Load	QSM	Tons per day
Portion of Depth Not Sampled	DN	ft.
Average Depth of Sampling	DS	ft.

2) Integer selectors JIN and JOUT, to be punched in format 2I1.

JIN selects the number and range in the computational size fractions. ND is the number of size fractions. If JIN=1, the size fractions used in reference [7] will be used. The first two size fractions will be used and the third deleted, resulting in ND=10. If JIN=2, the size fractions used in reference [7] will be used. In this case the first two size fractions will be deleted and the third one used insted, resulting in ND=9. If JIN=3, the user has the option of specifying the number and range of computational size fractions, up to 9 fractions. If this option is chosen, ND should be read in the card immediately following, in format I1.

JOUT selects the type of output desired. If JOUT=1, output will consist of the general data, check on convergence of Z Prime, and the final results in 20 columns, as follows.

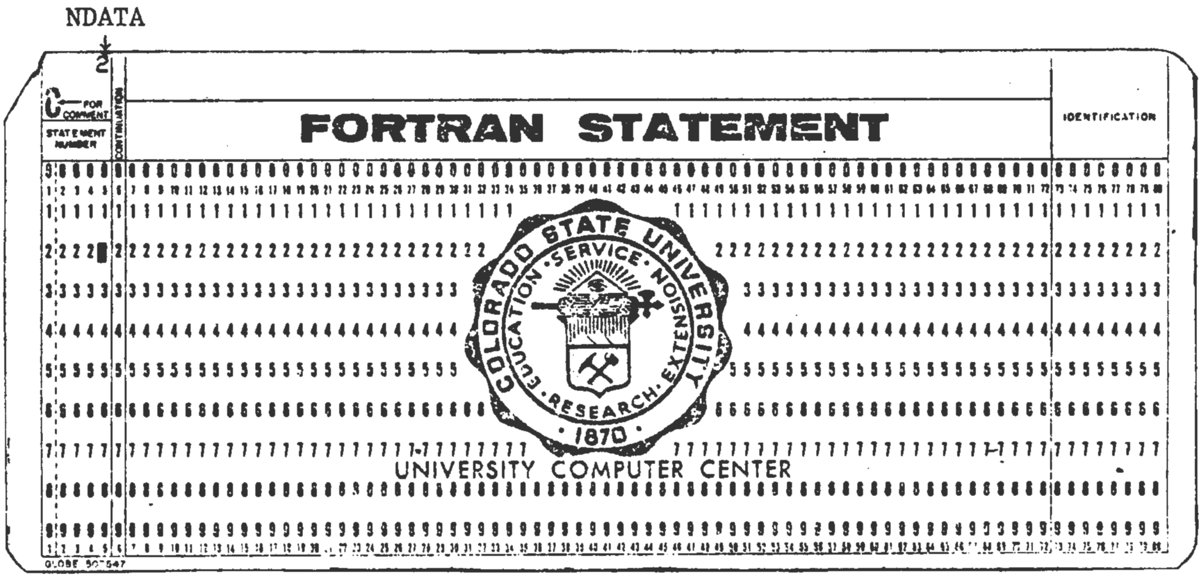
1) Geometric mean diameter, in ft.
2) PSI
3) PHI Shear
4) Percentage of Bed Material in Size Fraction
5) Bed Load Transport, in Tons/day
6) Percentage of Suspended Load in Size Fraction
7) Sampled Transport in Size Fraction
8) Multipliers
9) A Prime Values
10) A Double Prime Values
11) Geometric Mean Diameter, in ft.
12) J-One Prime
13) J-Two Prime
14) J-One Double Prime
15) J-Two Double Prime
16) Product of J's
17) I-One Double Prime
18) I-Two Double Prime
19) Product of I's
20) Computed Load, in Tons/day

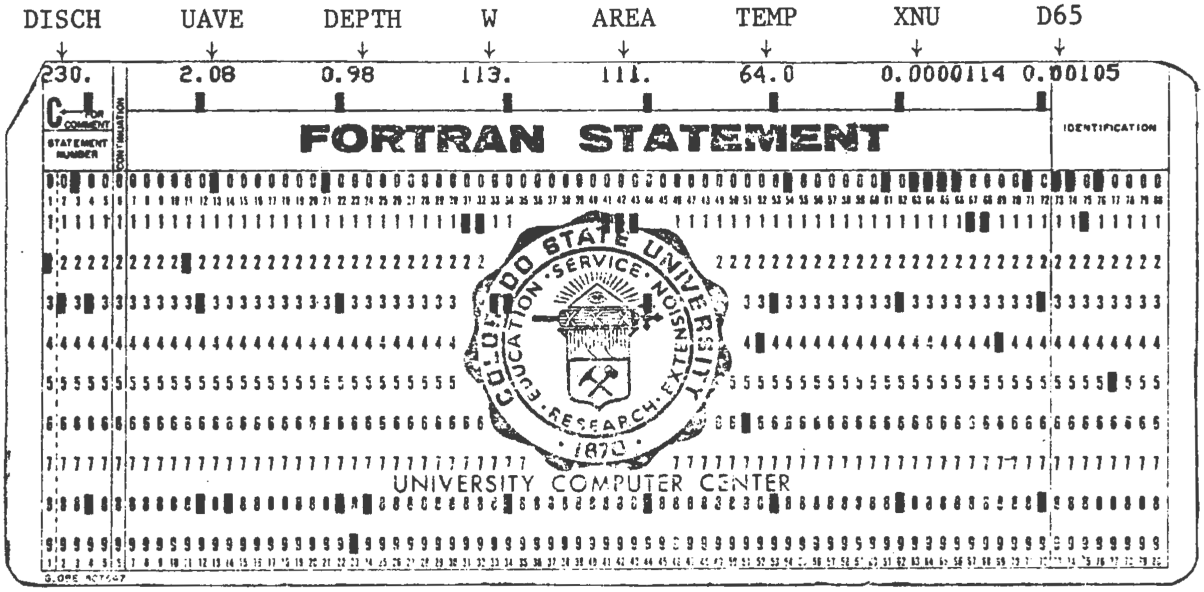
If JOUT=2 is selected, most of the 20 columns will be omitted in the printout, and instead only columns 1, 4, 5, 6 and 20 will be printed. Additionally, DRL(J) and DRU(J), lower and upper limits of the size fraction range, in mm., will be printed to the left of the five columns previously mentioned.

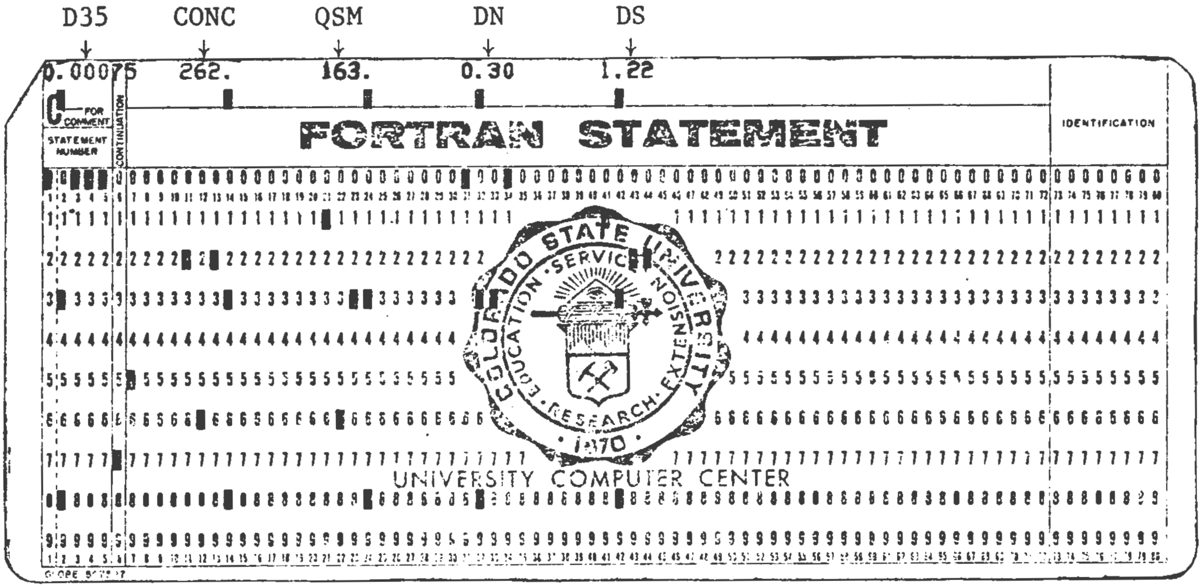
3) Data Arrays:

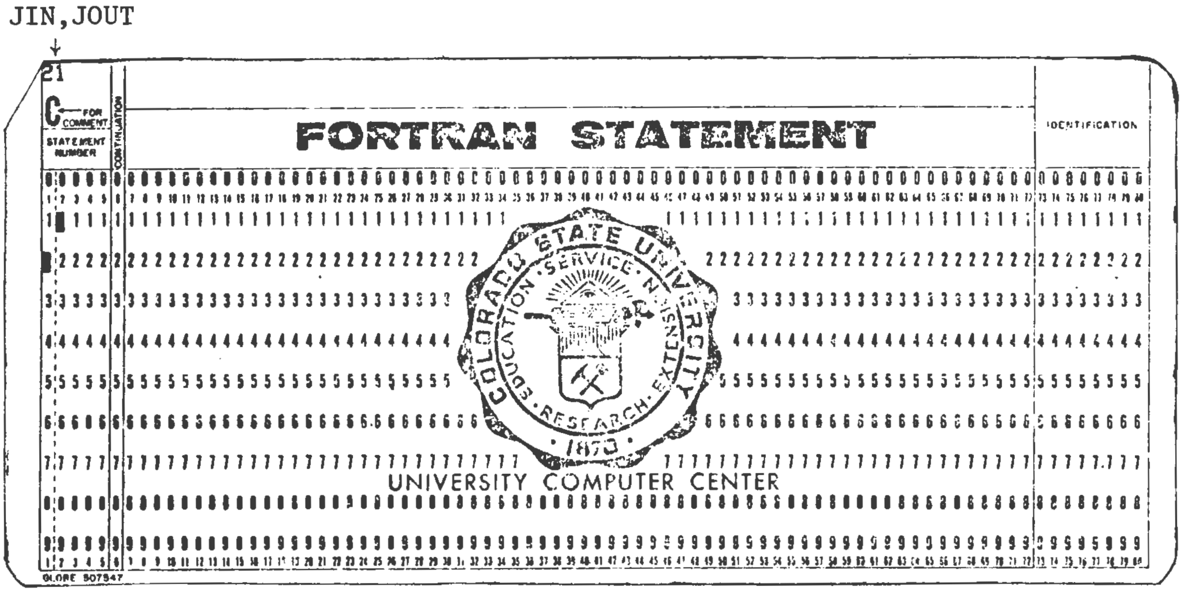
If JIN=1, the fractions of bed material in various size ranges FB(10), and fractions of suspended load in various size ranges FS(10) should be punched in format 2F10.0. If JIN=2, FB(9) and FS(9) should be punched in format 2F10.0. If JIN=3, the range of computational size fractions should be specified in addition to the percentages FB and FS. If this option is chosen, DRL(ND), DRU(ND), FB(ND) and FS(ND) should be punched in format 4F10.0. DRL(J) and DRU(J) are lower and upper limits of the size fraction range in mm., respectively. Note that size fractions should be punched in the order of increasing size.

A sequence of two runs is illustrated in the following arrangement of data cards. Different integer selection JIN and JOUT have been used in these runs for illustration. The corresponding output follows the data card sequence.

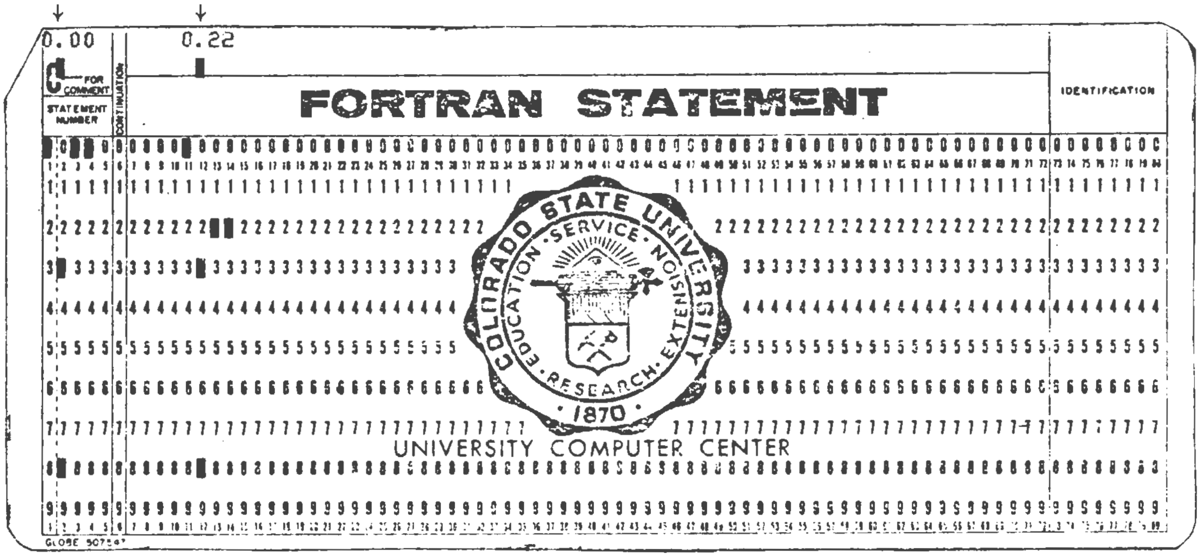



DISCH UAVE DEPTH W AREA TEMP XNU D65
230. 2.08 0.98 113. 111. 64.0 0.0000114 0.00105
FORTRAN STATEMENT
UNIVERSITY COMPUTER CENTER

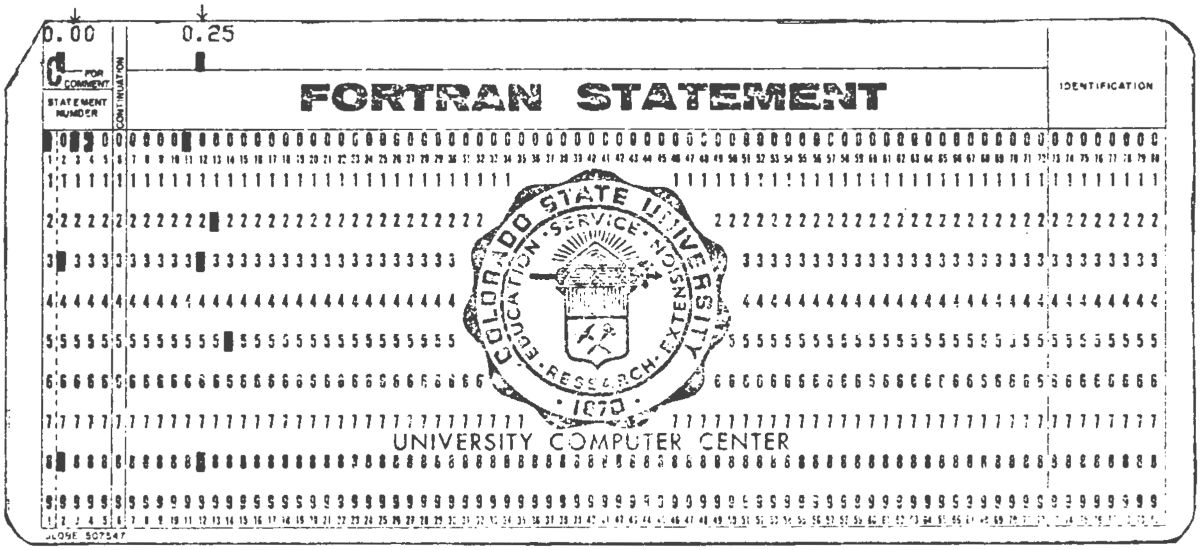

D35 CONC QSM DN DS
0.00075 262. 163. 0.30 1.22
FORTRAN STATEMENT
UNIVERSITY COMPUTER CENTER


JIN,JOUT
21
FORTRAN STATEMENT
UNIVERSITY COMPUTER CENTER

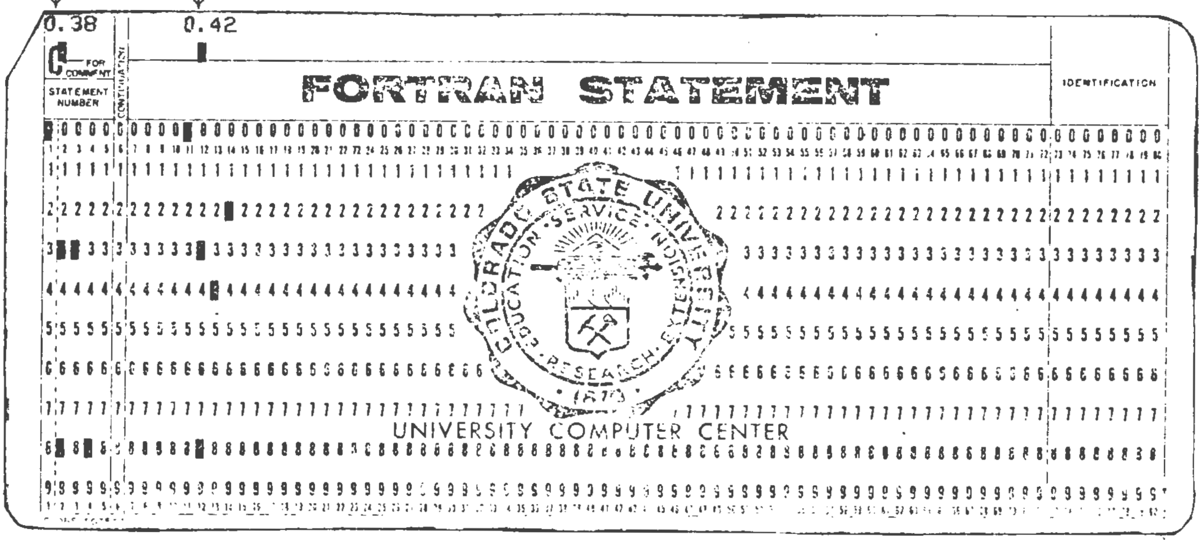
FB(1) FS(1)



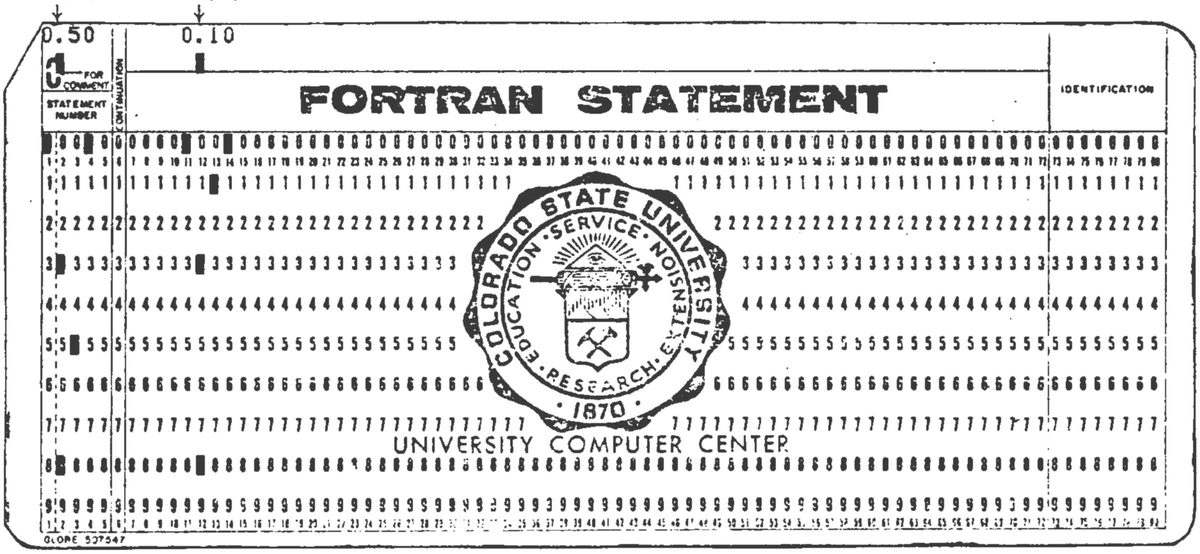
FB(2) FS(2)



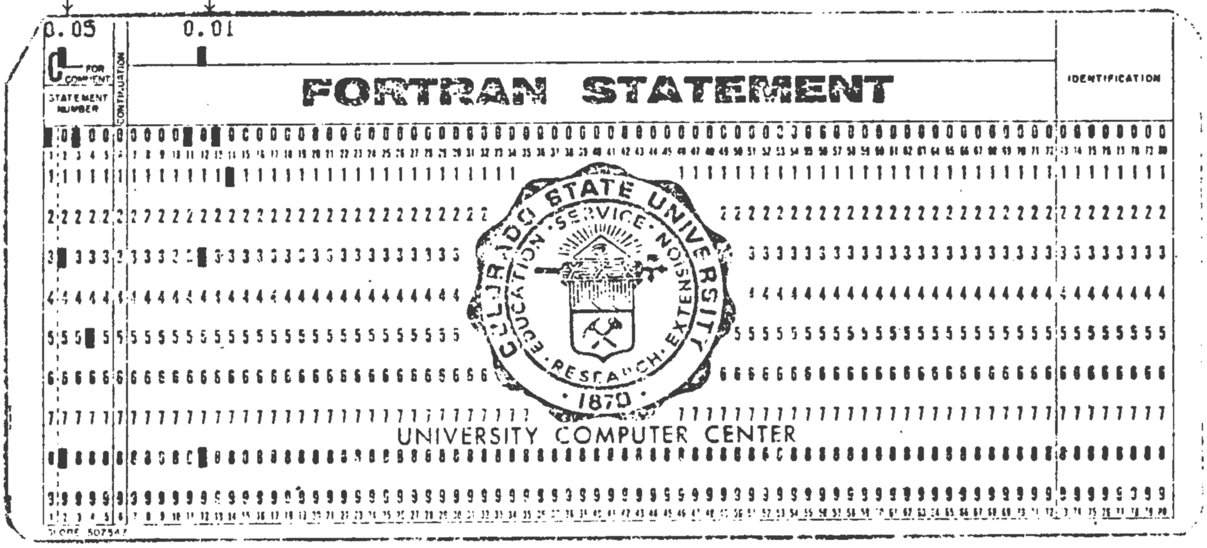
FB(3) FS(3)



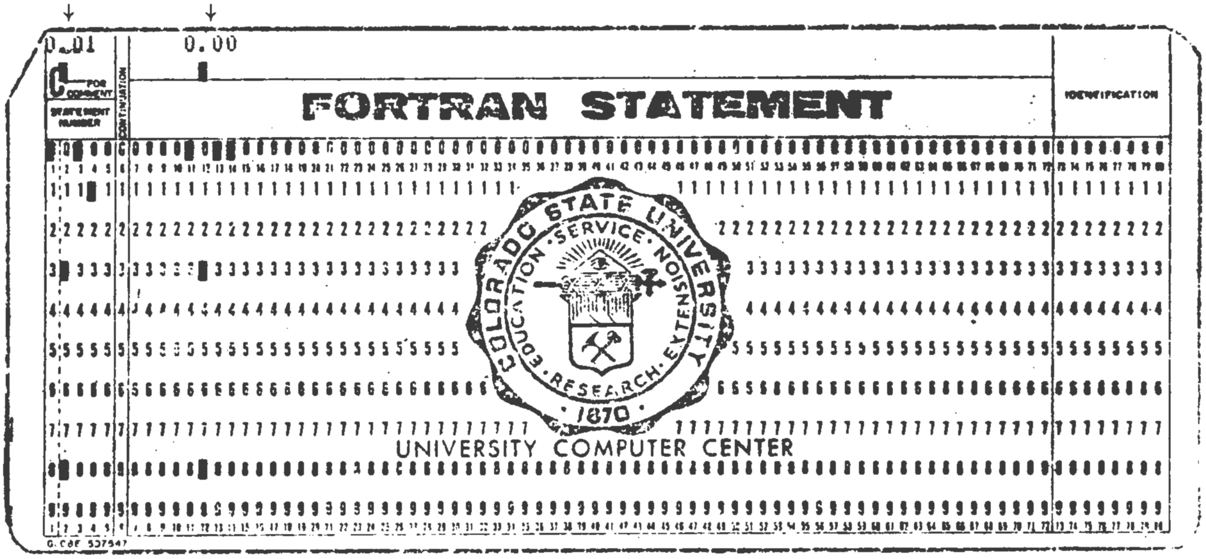
FB(4) FS(4)

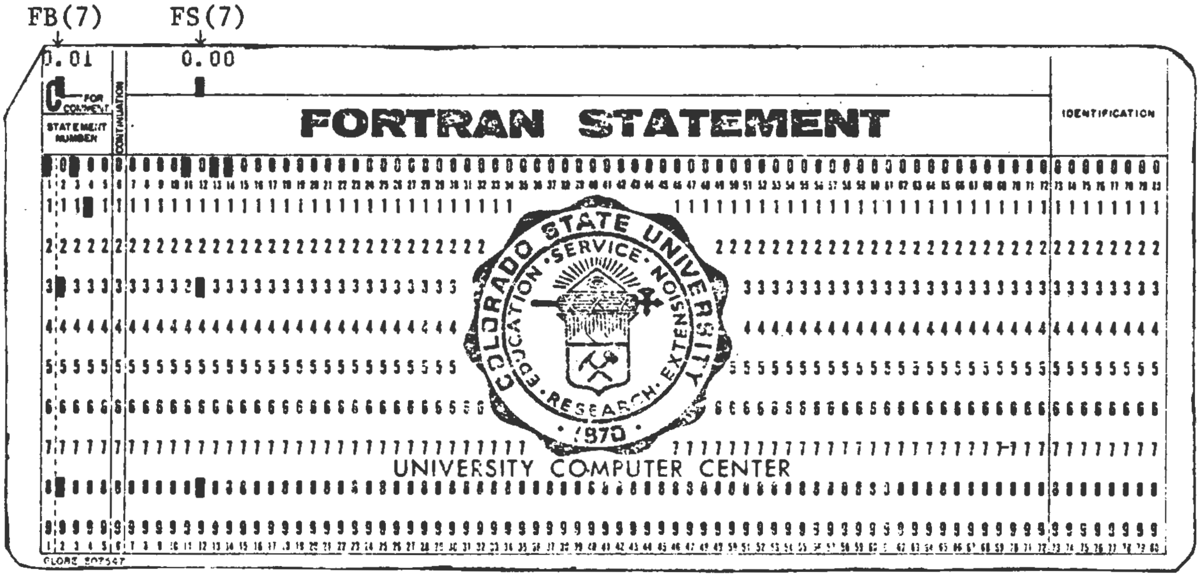


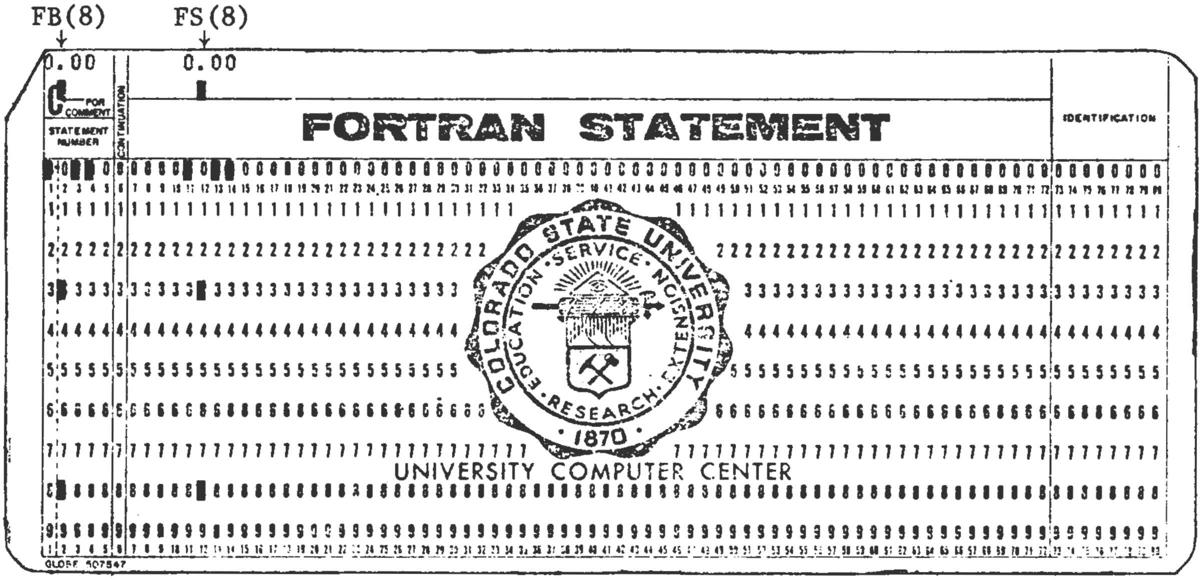
FB(5) FS(5)

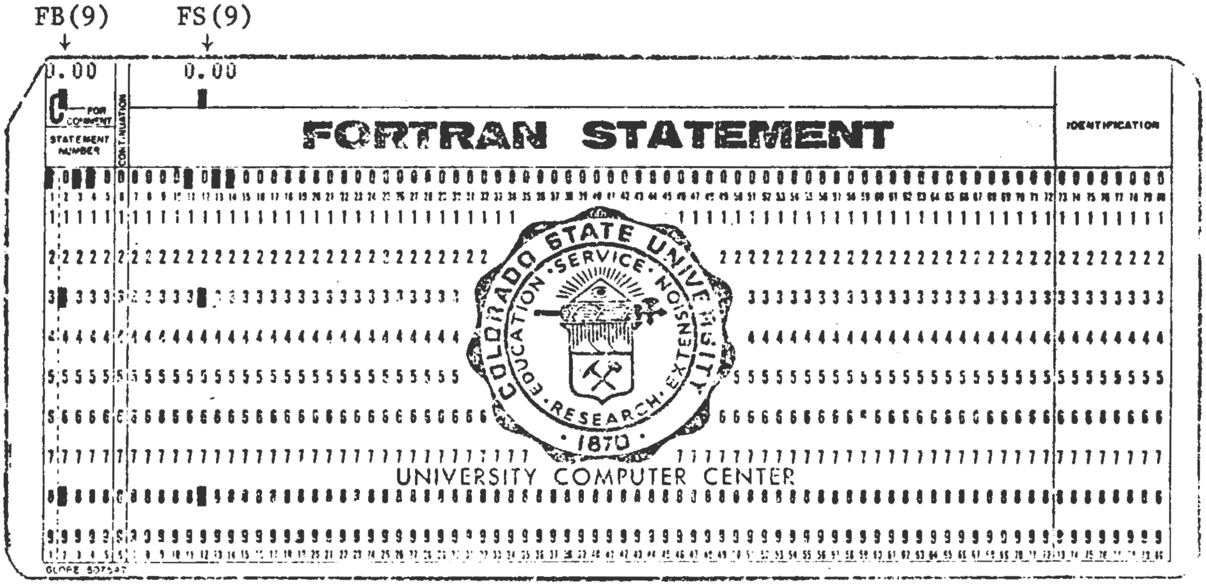


FB(6) FS(6)

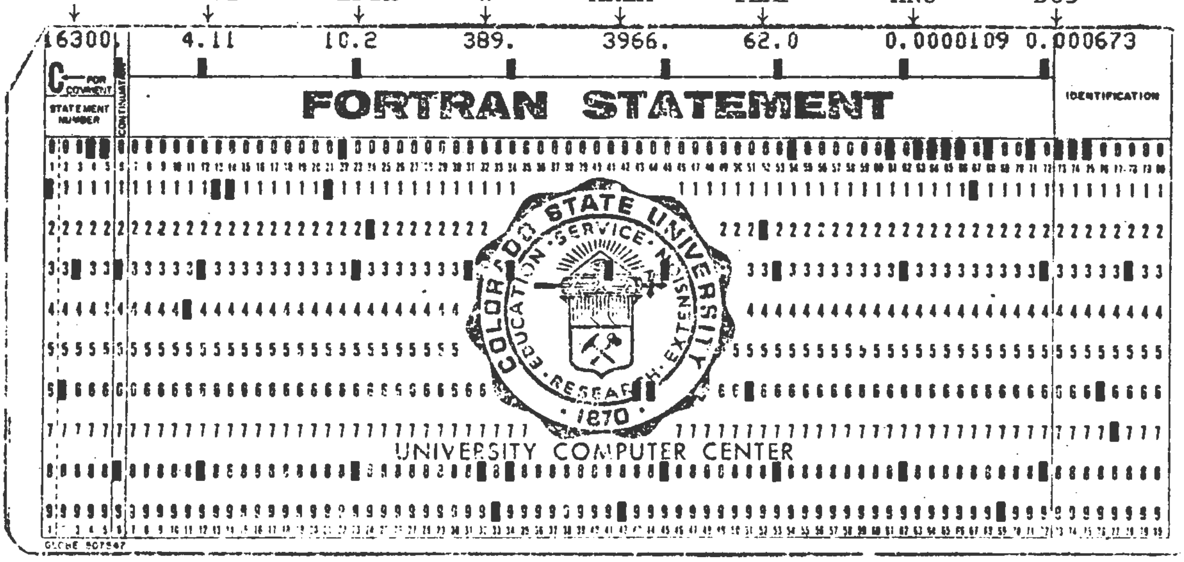



FB(7)
FS(7)
0.01
0.00
FORTRAN STATEMENT
IDENTIFICATION
COLORADO STATE UNIVERSITY
UNIVERSITY COMPUTER CENTER

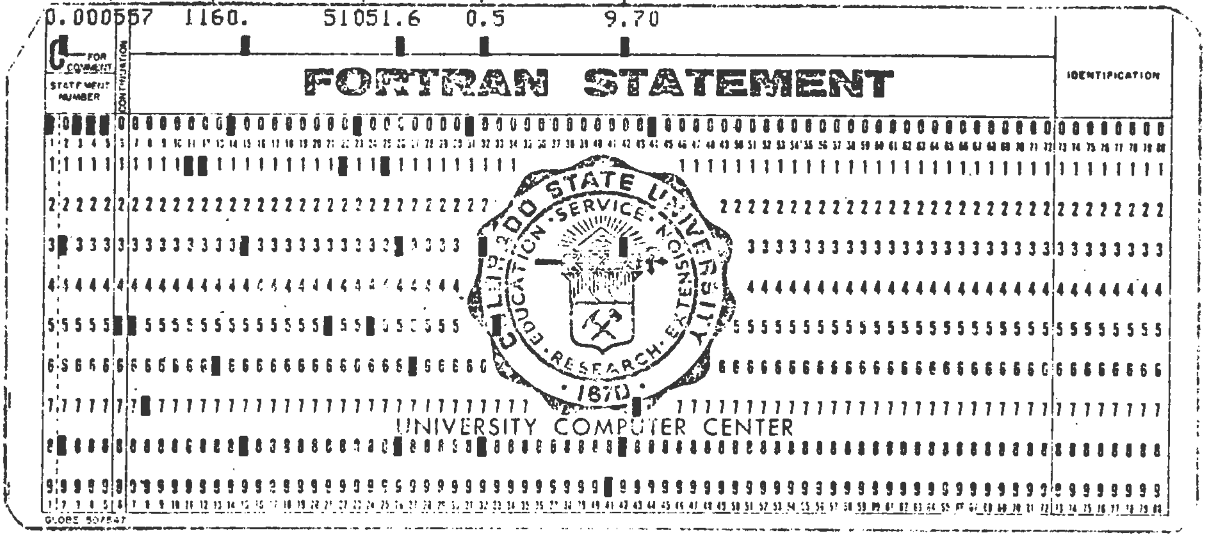

FB(8)
FS(8)
0.00
0.00
FORTRAN STATEMENT
IDENTIFICATION
COLORADO STATE UNIVERSITY
UNIVERSITY COMPUTER CENTER


FB(9)
FS(9)
0.00
0.00
FORTRAN STATEMENT
IDENTIFICATION
COLORADO STATE UNIVERSITY
UNIVERSITY COMPUTER CENTER

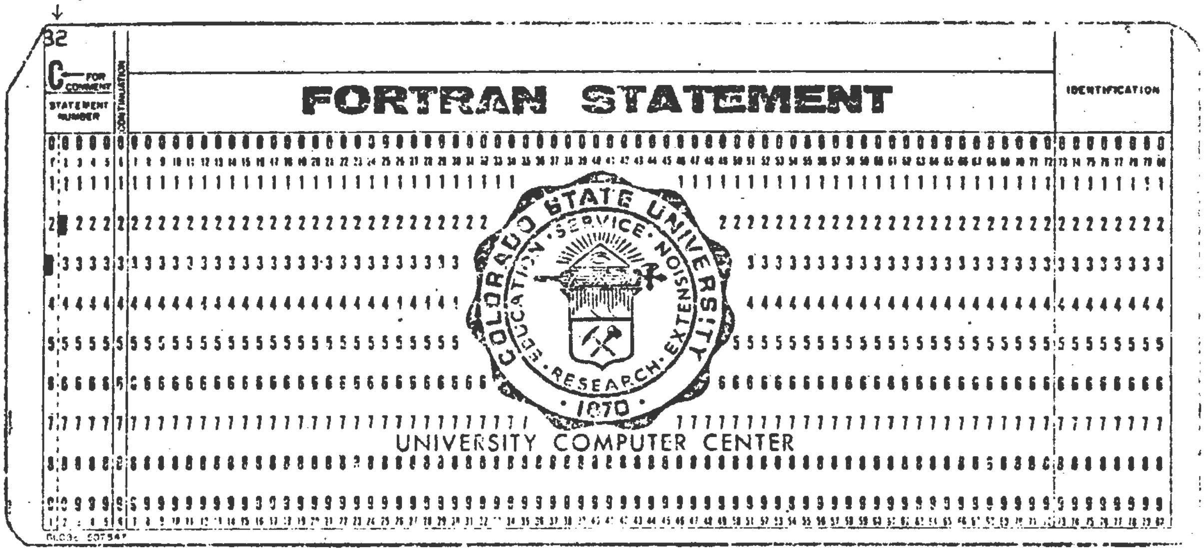
DISCH UAVE DEPTH W AREA TEMP XNU D65

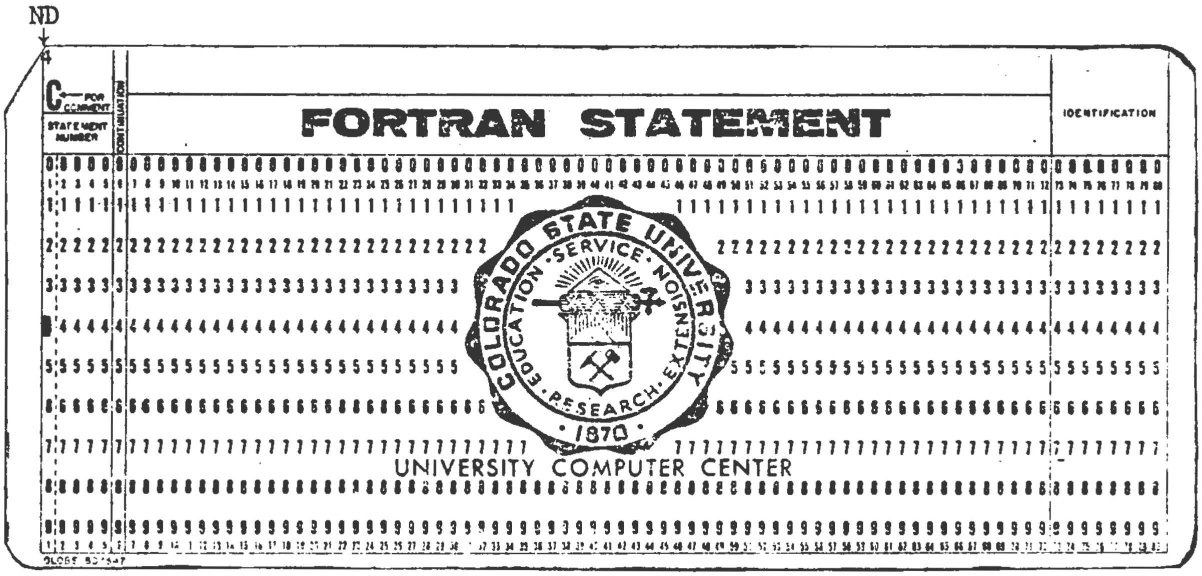


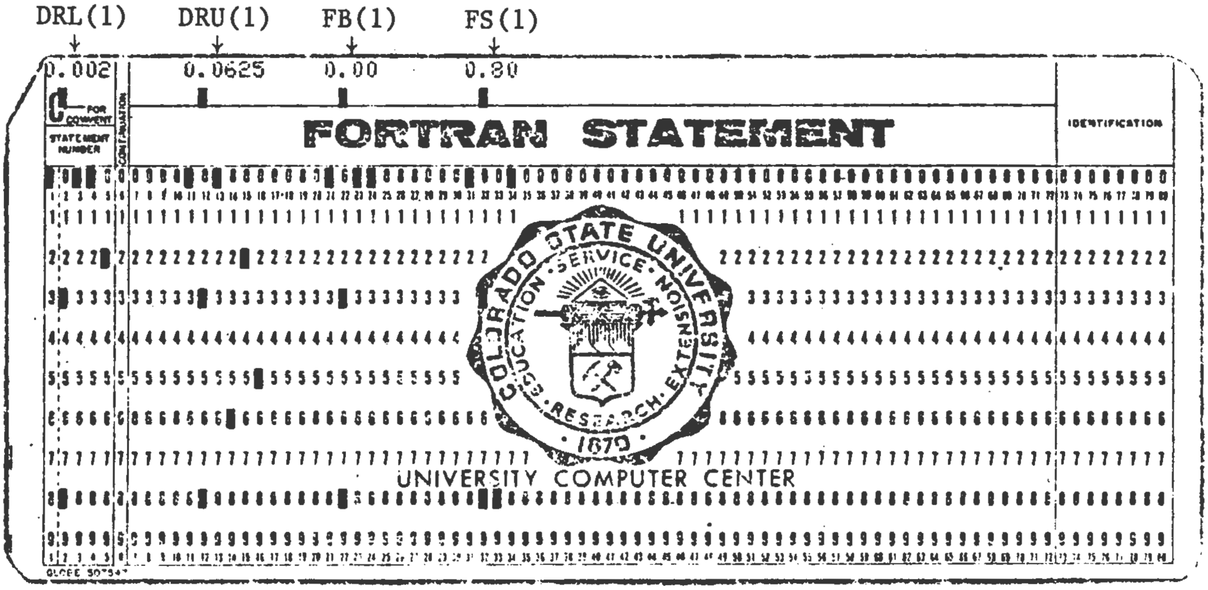
D35 CONC QSM DN DS

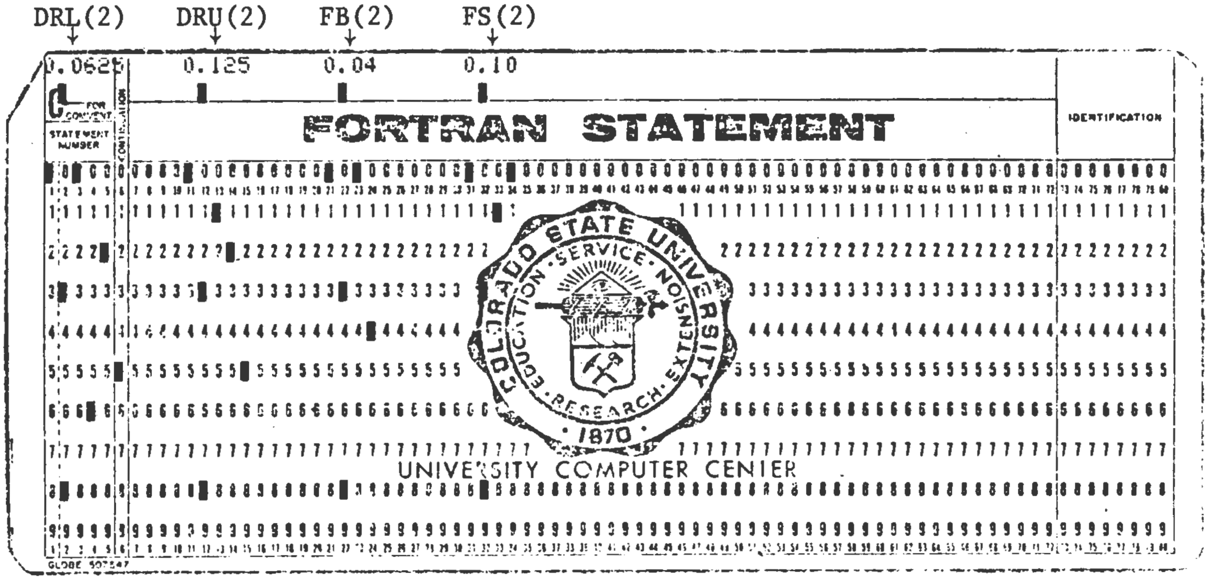


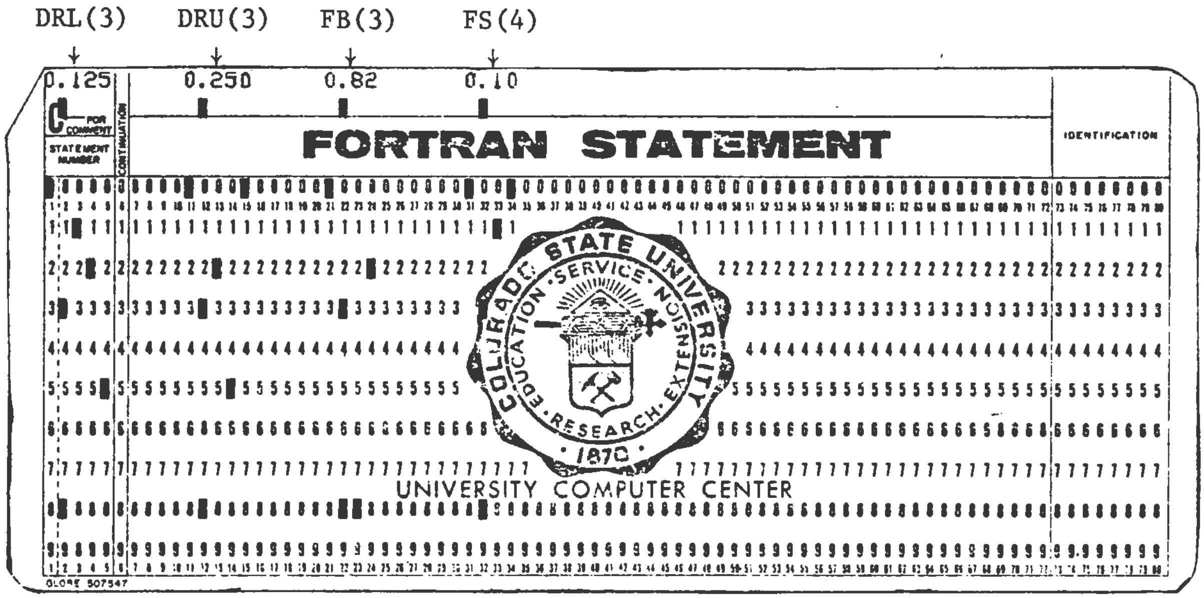
JIN,JOUT

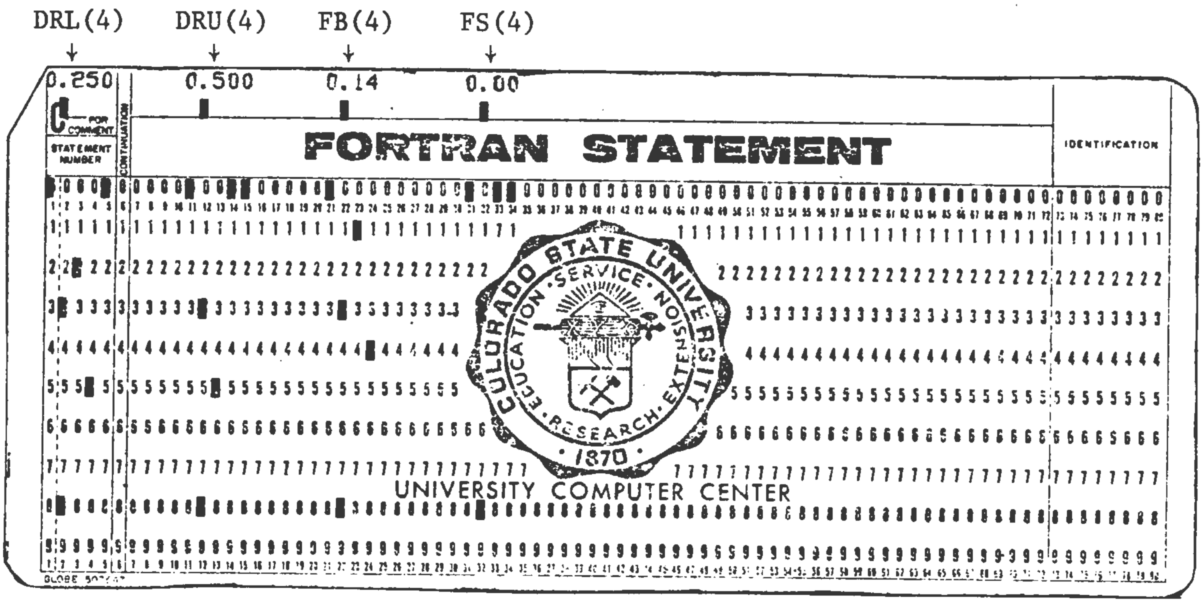



ND
4
FORTRAN STATEMENT
IDENTIFICATION
UNIVERSITY COMPUTER CENTER


DRL(1)
DRU(1)
FB(1)
FS(1)
0.002
0.0625
0.00
0.80
FORTRAN STATEMENT
IDENTIFICATION
UNIVERSITY COMPUTER CENTER


DRL(2)
DRU(2)
FB(2)
FS(2)
0.0625
0.125
0.04
0.10
FORTRAN STATEMENT
IDENTIFICATION
UNIVERSITY COMPUTER CENTER


DRL(3)
DRU(3)
FB(3)
FS(4)
0.125
0.250
0.82
0.10
FORTRAN STATEMENT
IDENTIFICATION
STATEMENT NUMBER
CONTINUATION
COLORADO STATE UNIVERSITY
SERVICE
EDUCATION
RESEARCH
EXTENSION
1870
UNIVERSITY COMPUTER CENTER


DRL(4)
DRU(4)
FB(4)
FS(4)
0.250
0.500
0.14
0.00
FORTRAN STATEMENT
IDENTIFICATION
STATEMENT NUMBER
CONTINUATION
COLORADO STATE UNIVERSITY
SERVICE
EDUCATION
RESEARCH
EXTENSION
1870
UNIVERSITY COMPUTER CENTER

COMPUTATION OF TOTAL SEDIMENT LOAD BY THE MODIFIED EINSTEIN PROCEDURE

DATA INPUT

SET 1	
WATER DISCHARGE	230.00 C.F.S.
AVERAGE VELOCITY	2.08 FT./SEC.
HYDRAULIC DEPTH	.98 FT.
WATER SURFACE WIDTH	113.00 FT.
AREA	111.00 SQ.FT.
TEMPERATURE	64.00 DEG.FAHREN.
KINEMATIC VISCOSITY	.0000114 SQ.FT./SEC.
D65	.001050 FT.
D35	.000750 FT.
AVERAGE CONCENTRATION	262.00 PPM.
SAMPLED SUSPENDED LOAD	163.0000 TONS/DAY
PORTION OF DEPTH NOT SAMPLED	.30 FT.
AVERAGE DEPTH AT SAMPLING	1.22 FT.

CONVERGENCE OF SUBROUTINE ZPCOM IS CHECKED BY PRINTING OUT VALUES INVOLVED

ITER.	ZTRY	RQSP	CRQSP	DCRQ
1	.80537	6.79554	4.98206	-1.81348
2	.75273	6.79554	7.08930	.29376
3	.75911	6.79554	6.79193	-.00362

CONVERGENCE OF SUBROUTINE ZPCOM IS CHECKED BY PRINTING OUT VALUES INVOLVED

ITER.	ZTRY	RQSP	CRQSP	DCRQ
1	1.19938	.43475	.43236	-.00240

CONVERGENCE OF SUBROUTINE ZPCOM IS CHECKED BY PRINTING OUT VALUES INVOLVED

ITER.	ZTRY	RQSP	CRQSP	DCRQ
1	1.26422	.27654	.35921	.08267
2	1.31128	.27654	.28384	.00730
3	1.31656	.27654	.27645	-.00009

ARRAYS ZP AND VS BEFORE LEAST SQUARE FIT

J	ZP(J)	VS(J)
3	.759113	.067624
4	1.199380	.152040
5	1.316557	.258550

ARRAYS ZP AND VS AFTER LEAST SQUARE FIT

J	ZP(J)	VS(J)
1	.084272	.000348
2	.476377	.020833
3	.784160	.067624
4	1.104958	.152040
5	1.383415	.258550
6	1.647641	.390728
7	1.927071	.565719
8	2.239438	.806727
9	2.596524	1.144244

J	D(J)	PSI(J)	PHISH(J)	FB(J)	XIBQB(J)	FS(J)	QSP(J)	XMULT(J)	ZP(J)	APP(J)
1	.000037	5.489	.51128	0.000	0.000	.220	28.292	0.000	.084	.000075
2	.000290	5.489	.51128	0.000	0.000	.250	32.150	0.000	.476	.000592
3	.000580	5.489	.51128	.380	7.948	.420	54.012	0.000	.784	.001184
4	.001160	5.489	.51128	.500	29.580	.100	12.860	0.000	1.105	.002367
5	.002320	6.791	.28418	.050	4.650	.010	1.286	0.000	1.383	.004734
6	.004640	13.582	.02402	.010	.222	0.000	0.000	0.000	1.648	.009469
7	.009280	27.163	.00010	.010	.003	0.000	0.000	0.000	1.927	.018938
8	.018559	54.327	.00000	0.000	0.000	0.000	0.000	0.000	2.239	.037876
9	.037118	108.654	.00000	0.000	0.000	0.000	0.000	0.000	2.597	.075752

J	D(J)	COL16(J)	COL17(J)	COL18(J)	COL19(J)	COL20(J)	COL21(J)	COL22(J)	COL23(J)	COMP.LOAD
1	.000037	.721	-.415	1.001	-1.161	1.307	0.000	0.000	0.000	36.9895
2	.000290	.637	-.460	1.443	-3.082	1.944	0.000	0.000	0.000	62.4968
3	.000580	0.000	0.000	0.000	0.000	0.000	2.633	-8.027	21.089	167.6220
4	.001160	0.000	0.000	0.000	0.000	0.000	.832	-3.066	6.823	201.8157
5	.002320	0.000	0.000	0.000	0.000	0.000	.434	-1.637	3.995	18.5788
6	.004640	0.000	0.000	0.000	0.000	0.000	.282	-1.002	3.009	.6691
7	.009280	0.000	0.000	0.000	0.000	0.000	.202	-.636	2.519	.0065
8	.018559	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.0000
9	.037118	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.0000

TOTAL BED LOAD 42.4035 TONS/DAY
TOTAL SUSPENDED BED MATERIAL LOAD 445.7749 TONS/DAY
TOTAL BED MATERIAL LOAD 488.1784 TONS/DAY

COMPUTATION OF TOTAL SEDIMENT LOAD BY THE MODIFIED EINSTEIN PROCEDURE

DATA INPUT

SET	2	
WATER DISCHARGE	16300.00	C.F.S.
AVERAGE VELOCITY	4.11	FT./SEC.
HYDRAULIC DEPTH	10.20	FT.
WATER SURFACE WIDTH	389.00	FT.
AREA	3966.00	SQ.FT.
TEMPERATURE	62.00	DEG.FAHREN.
KINEMATIC VISCOSITY	.0000109	SQ.FT./SEC.
D65	.000673	FT.
D35	.000557	FT.
AVERAGE CONCENTRATION	1160.00	PPM.
SAMPLED SUSPENDED LOAD	51051.6000	TONS/DAY
PORTION OF DEPTH NOT SAMPLED	.50	FT.
AVERAGE DEPTH AT SAMPLING	9.70	FT.

DRL(J)	DRU(J)	D(J)	FB(J)	XIBQB(J)	FS(J)	FQL(J)
.002000	.062500	.000037	0.000000	0.000000	.800000	40696.436
.062500	.125000	.000290	.040000	7.651876	.100000	8809.269
.125000	.250000	.000580	.820000	443.676883	.100000	11014.426
.250000	.500000	.001160	.140000	214.252539	0.000000	748.035

TOTAL BED LOAD	665.5813	TONS/DAY
TOTAL SUSPENDED BED MATERIAL LOAD	60602.5843	TONS/DAY
TOTAL BED MATERIAL LOAD	61268.1656	TONS/DAY

5.2 Fortran Names for Input and Output Variables

INPUT

Water Discharge	DISCH
Average Velocity	UAVE
Hydraulic Depth	DEPTH
Water Surface Width	W
Area	AREA
Temperature	TEMP
Kinematic Viscosity	XNU
65 Percent Finer Diameter for Bed-Material	D65
35 Percent Finer Diameter for Bed-Material	D35
Average Concentration	CONC
Sampled Suspended Load	QSM
Portion of Depth Not Sampled	DN
Average Depth of Sampling	DS

OUTPUT

Geometric Mean Diameter, in ft.	D(J)
PSI	PSI(J)
PHI Shear	PHISH(J)
Percentage of Bed-Material in Size Fraction	FB(J)
Bed-Load Transport, in Tons/day	XIBQB(J)
Percentage of Suspended Load in Size Fraction	FS(J)
Sampled Transport in Size Fraction	QSP(J)
Multipliers	XMULT(J)
Z Prime Values	ZP(J)
A Double Prime Values	APP(J)
Geometric Mean Diameter, in ft.	D(J)
J-One Prime	COL16(J)
J-Two Prime	COL17(J)
J-One Double Prime	COL18(J)
J-Two Double Prime	COL19(J)
Product of J's	COL20(J)
I-One Double Prime	COL21(J)
I-Two Double Prime	COL22(J)
Product of I's	COL23(J)
Computed Load, in Tons/day	FQL(J)
Trial Z	ZTRY
Real Q_s'	RQSP
Computed Q_s'	CRQSP
Difference of Real and Computed Q_s'	DCRQ
Settling Velocity	VS(J)
Total Bed Load	TBL
Total Suspended Bed Material Load	TSL
Total Bed Material Load	TQL

REFERENCES

1. Einstein, H.A., "The Bed-Load Function for Sediment Transportation in Open Channel Flows", Technical Bulletin 1026, September 1950, U.S. Department of Agriculture, Soil Conservation Service.

2. Mahmood, K., "Lognormal Size Distribution of Particulate Matter", Journal of Sedimentary Petrology, Vol. 43, No. 4, 1973.

3. Mahmood, K., "Flow in Sand-Bed Channels", CUSUSWASH Water Management Technical Publication No. 11, 1971, Colorado State University, Fort Collins.

4. Colby, B.R., "Discharge of Sands and Mean Velocity Relationships in Sand-Bed Streams", U.S. Geological Survey Professional Paper 462-A, 1964.

5. Meyer-Peter, E. and Muller, R., "Formulas for Bed Load Transport", International Association for Hydraulic Research, Second Meeting, Stockholm, 1948.

6. Colby, B.R. and Hembree, C.H., "Computations of Total Sediment Discharge, Niobrara River near Cody, Nebraska", U.S. Geological Survey Water Supply Paper 1357, 1955.

7. U.S. Bureau of Reclamation Publication, "Step Method for Computing Total Sediment Load by the Modified Einstein Procedure", July 1955 (Revised).

8. U.S. Bureau of Reclamation Publication, "Computation of Z for Use in the Modified Einstein Procedure", June 1966.

APPENDIX:

PROGRAM LISTINGS

EINSTIN

LISTING

```
      PROGRAM EINSTIN (INPUT,OUTPUT)
C
C
C     DEVELOPED          COLORADO STATE UNIVERSITY ENGINEERING RESEARCH
C                        CENTER, FORT COLLINS, COLORADO 80523.
C     PURPOSE            COMPUTATION OF BED MATERIAL LOAD BY EINSTEINS
C                        METHOD.
C     REFERENCE          EINSTEIN, H.A., THE BED-LOAD FUNCTION FOR
C                        SEDIMENT TRANSPORTATION IN OPEN CHANNEL FLOW,
C                        TECHNICAL BULLETIN 1026, SEPTEMBER 1950,
C                        UNITED STATES DEPARTMENT OF AGRICULTURE
C     CORE USAGE         CDC 6400 SCOPE 3.3 SYSTEM DEFAULT VALUE,
C                        43000 OCTAL.
C     COMPILATION TIME   APPROXIMATELY 6 SEC.
C     CENTRAL PROCESSOR
C     TIME FOR ONE
C     SET OF DATA        LESS THAN 1.0 SEC.
C
C
C     INPUT AND OUTPUT DESCRIPTION
C
C     THE FIRST CARD IN THE INPUT LOGICAL RECORD SHOULD CONTAIN THE
C     VALUE OF NDATA, IN FORMAT I5. NDATA IS THE NUMBER OF SETS OF INPUT
C     DATA TO BE FED TO THE COMPUTER AT A TIME. A SET OF INPUT DATA
C     CONSISTS OF A GROUP OF VARIABLES NECESSARY TO SPECIFY A PROBLEM,
C     AS DETAILED BELOW.
C
C     THE FIRST CARD IN INPUT IS FOLLOWED BY THE SETS OF INPUT DATA,
C     TO BE PUNCHED IN FORMAT 8F10.0
C     A SET OF INPUT DATA CONSISTS OF THE FOLLOWING VARIABLES,
C                                      FORTRAN NAME         UNITS
C     1) WATER DISCHARGE               DISCH                C.F.S.
C     2) AVERAGE VELOCITY              VELAV                FT./SEC.
C     3) HYDRAULIC DEPTH               DEPTH                FT.
C     4) WATER SURFACE WIDTH           W                    FT.
C     5) ENERGY GRADIENT               SS                   FT./FT.
C     6) KINEMATIC VISCOSITY           RMU                  SQ.FT./SEC.
C     7) MEDIAN BED MATERIAL SIZE      D50M                 MM.
C     8) GRADATION COEFFICIENT         SIGMA                NO UNITS
C
C
C     OUTPUT CONSISTS OF FIVE COLUMNS, AS FOLLOWS
C     1) FRACTION NUMBER
C     2) GEOMETRIC MEAN SIZE, IN MM.
C     3) BED LOAD, IN TONS/DAY
C     4) SUSPENDED BED MATERIAL LOAD, IN TONS/DAY
C     5) TOTAL BED MATERIAL LOAD, IN TONS/DAY
C
C
      DIMENSION X(10), D(10), TSD(10), BD(10), SD(10)
      COMMON /CEF/ CJ0(2),CJ1(2),CJ2(2),CJ3(2),CJ(2),C1(2),C2(2),C3(2),C
     14(2),M
      READ 115, NDATA
      DO 109 L=1,NDATA
        PRINT 116
        PRINT 117
        READ 110, DISCH,VELAV,DEPTH,W,SS,RMU,D50M,SIGMA
        PRINT 111, DISCH,VELAV,DEPTH,W,SS,RMU,D50M,SIGMA
        C=1./304.8
```

```
      G=32.2
      M=0
      DISCH=DISCH/W
C
C *** ASSUMPTION OF LOG-NORMAL SIZE DISTRIBUTION OF BED MATERIALS
C
      DF95=D50M*C*(SIGMA)**1.645
      DF90=D50M*C*(SIGMA)**1.285
      DF80=D50M*C*(SIGMA)**0.845
      DF70=D50M*C*(SIGMA)**0.525
      DF60=D50M*C*(SIGMA)**0.255
      DF50=D50M*C
      DF40=D50M*C/(SIGMA)**0.255
      DF30=D50M*C/(SIGMA)**0.525
      DF20=D50M*C/(SIGMA)**0.845
      DF10=D50M*C/(SIGMA)**1.285
      DF5=D50M*C/(SIGMA)**1.645
      D(1)=SQRT(DF5*DF10)
      D(2)=SQRT(DF10*DF20)
      D(3)=SQRT(DF20*DF30)
      D(4)=SQRT(DF30*DF40)
      D(5)=SQRT(DF40*DF50)
      D(6)=SQRT(DF50*DF60)
      D(7)=SQRT(DF60*DF70)
      D(8)=SQRT(DF70*DF80)
      D(9)=SQRT(DF80*DF90)
      D(10)=SQRT(DF90*DF95)
      D65=D50M*C*SIGMA**0.385
      D35=D50M*C/SIGMA**0.385
      DO 101 I=1,10
  101 X(I)=D(I)*304.8
      RT=DEPTH
      RBP=(VELAV*D65**0.1667/(7.66*SQRT(G*SS)))**1.5
      RBPP=RT-RBP
      SVP=SQRT(G*RBP*SS)
      DELTA=11.6*RMU/SVP
      X4=D65/DELTA
      CALL FIG4 (X4,Y4)
C
C *** COMPUTATION OF SEDIMENT DISCHARGE
C
      DELT=D65/Y4
      RATIO=DELT/DELTA
      IF (RATIO.LT.1.80) GO TO 102
      CAPX=0.77*DELT
      GO TO 103
  102 CAPX=1.39*DELTA
  103 X8=D65/DELTA
      CALL FIG8 (X8,Y8)
      BETAX=ALOG10(10.6*CAPX/DELT)
      PP=2.304*ALOG10(30.2*Y4*RT/D65)
      DO 105 I=1,10
         X7=D(I)/CAPX
         CALL FIG7 (X7,Y7)
         PSIS=Y7*Y8*(1.025/BETAX)**2*(1.68*D(I)/(RBP*SS))
         X10=PSIS
         CALL FIG10 (X10,Y10)
         BD(I)=Y10*1215.00*(D(I)**1.5)*0.1
         A=2.0*D(I)/RT
```

```
         SETV=((36.064*(D(I)**3)+36.*RMU**2)**0.5-6.*RMU)/D(I)
         Z=SETV/(0.4*SVP)
         IF (Z.LT.5.5) GO TO 104
         TSD(I)=BD(I)
         GO TO 105
 104     XM=A
         CALL POLYNML (XM,Z,XI1,XI2,XJ1,XJ2)
         TSD(I)=BD(I)*(PP*XI1+XI2+1.0)
         IF (TSD(I).LT.BD(I)) TSD(I)=BD(I)
 105     CONTINUE
      SDT=0.
      SDB=0.
      SDS=0.
      DO 106 I=1,10
         SDT=SDT+TSD(I)
         SDB=SDB+BD(I)
         SD(I)=TSD(I)-BD(I)
         SDS=SDS+SD(I)
 106  CONTINUE
      DO 107 I=1,10
         SD(I)=SD(I)*43.2*W
         BD(I)=BD(I)*43.2*W
         TSD(I)=TSD(I)*43.2*W
 107  CONTINUE
      SDS=SDS*43.2*W
      SDB=SDB*43.2*W
      SDT=SDT*43.2*W
      PRINT 112
      DO 108 I=1,10
 108  PRINT 113, I,X(I),BD(I),SD(I),TSD(I)
      PRINT 114, SDB,SDS,SDT
 109 CONTINUE
     CALL EXIT
C
 110 FORMAT (8F10.0)
 111 FORMAT (5X, 34HWATER DISCHARGE                    ,F12.2, 12H C.F.S
   1.      ,/5X, 34HAVERAGE VELOCITY                   ,F12.2, 12H FT./S
   2EC.    ,/5X, 34HHYDRAULIC DEPTH                    ,F12.2, 12H FT.
   3       ,/5X, 34HWATER SURFACE WIDTH                ,F12.2, 12H FT.
   4       ,/5X, 34HENERGY GRADIENT                    ,F12.7, 12H FT./F
   5T.     ,/5X, 34HKINEMATIC VISCOSITY                ,F12.7, 12H SQ.FT
   6./SEC.,/5X, 34HMEDIAN BED MATERIAL SIZE            ,F12.2, 13H MM.
   7       ,/5X, 34HGRADATION COEFFICIENT               ,F12.2,//)
 112 FORMAT (1X,8HFRACTION,5X,13HGEO MEAN SIZE,8X,8HBED LOAD,7X,16HSUS
   1BED MAT LOAD,5X,12HBED MAT LOAD/4X,3HNO.,11X,4H(MM),12X,10H(TONS/D
   2AY),9X,10H(TONS/DAY),9X,10H(TONS/DAY)/)
 113 FORMAT (2X,I3,4F19.5)
 114 FORMAT (///,5X, 34HTOTAL BED LOAD                      ,F16.4, 10H
   1TONS/DAY,/5X, 34HTOTAL SUSPENDED BED MATERIAL LOAD ,F16.4, 10H  TO
   2NS/DAY,/5X, 34HTOTAL BED MATERIAL LOAD              ,F16.4, 10H  TONS
   3/DAY)
 115 FORMAT (I5)
 116 FORMAT (1H1)
 117 FORMAT (10X, 33HCOMPUTATION OF TOTAL BED MATERIAL,/,10X, 38HLOAD B
   1Y THE EINSTEIN BED-LOAD FUNCTION,///)
C
     END
```

```
      SUBROUTINE FIG4 (X,Y)
C
C *** THIS SUBROUTINE APPROXIMATES EINSTEINS FIG 4 SERIES OF EQNS.
C
      IF (X.LE.0.40) GO TO 101
      GO TO 102
  101 Y=1.769*ALOG10(X/0.080)
      GO TO 117
  102 IF (X.GT.0.40.AND.X.LE.0.56) GO TO 103
      GO TO 104
  103 Y=1.495*ALOG10(X/0.059)
      GO TO 117
  104 IF (X.GT.0.56.AND.X.LE.0.76) GO TO 105
      GO TO 106
  105 Y=0.92*ALOG10(X/0.0145)
      GO TO 117
  106 IF (X.GT.0.76.AND.X.LE.0.96) GO TO 107
      GO TO 108
  107 Y=0.292*ALOG10(X/2.9E-06)
      GO TO 117
  108 IF (X.GT.0.96.AND.X.LE.1.35) GO TO 109
      GO TO 110
  109 Y=0.277*ALOG10(632000.0/X)
      GO TO 117
  110 IF (X.GT.1.35.AND.X.LE.3.00) GO TO 111
      GO TO 112
  111 Y=1.115*ALOG10(34.4/X)
      GO TO 117
  112 IF (X.GT.3.00.AND.X.LE.4.00) GO TO 113
      GO TO 114
  113 Y=0.725*ALOG10(128.0/X)
      GO TO 117
  114 IF (X.GT.4.00.AND.X.LE.6.70) GO TO 115
      GO TO 116
  115 Y=0.399*ALOG10(2160.0/X)
      GO TO 117
  116 IF (X.GT.6.70) Y=1.0
  117 RETURN
C
      END
```

```
      SUBROUTINE FIG5 (X,Y)
C
C *** THIS SUBROUTINE APPROXIMATES EINSTEINS FIG 5 BY A SERIES OF EQNS.
C
      IF (X.LE.1.0) GO TO 101
      GO TO 102
  101 Y=40.0*X**(-1.288)
      GO TO 109
  102 IF (X.GT.1.0.AND.X.LE.2.0) GO TO 103
      GO TO 104
  103 Y=40.0*X**(-0.982)
      GO TO 109
  104 IF (X.GT.2.0.AND.X.LE.4.0) GO TO 105
      GO TO 106
  105 Y=31.1*X**(-0.618)
      GO TO 109
  106 IF (X.GT.4.0.AND.X.LE.8.0) GO TO 107
      GO TO 108
  107 Y=26.0*X**(-0.486)
      GO TO 109
  108 IF (X.GT.8.0) Y=21.4*X**(-0.394)
  109 RETURN
C
      END
```

```
      SUBROUTINE FIG7 (X,Y)                                             EIN 2400
C                                                                       EIN 2410
C *** THIS SUBROUTINE APPROXIMATES EINSTEINS FIG 7 BY A SERIES OF EQNS. EIN 2420
C                                                                       EIN 2430
      IF (X.LE.0.20) Y=(X/1.090)**(-2.088)                              EIN 2440
      IF (X.GT.0.20.AND.X.LE.0.40) Y=(X/0.877)**(-2.402)                EIN 2450
      IF (X.GT.0.40.AND.X.LE.0.65) Y=(X/0.832)**(-2.582)                EIN 2460
      IF (X.GT.0.65.AND.X.LE.0.80) Y=(X/0.990)**(-1.515)                EIN 2470
      IF (X.GT.0.80.AND.X.LE.1.00) Y=(X/1.185)**(-0.826)                EIN 2480
      IF (X.GT.1.00.AND.X.LE.1.45) Y=(X/1.450)**(-0.375)                EIN 2490
      IF (X.GT.1.45) Y=1.0                                              EIN 2500
      RETURN                                                            EIN 2510
C                                                                       EIN 2520
      END                                                               EIN 2530
```

```
      SUBROUTINE FIG8 (X,Y)                                            EIN 2540
C                                                                      EIN 2550
C *** THIS SUBROUTINE ESTIMATES EINSTEINS FIG 8 BY A SERIES OF EQNS.    EIN 2560
C                                                                      EIN 2570
      IF (X.LT.0.66) Y=(X/1.005)**1.178                                EIN 2580
      IF (X.GT.0.66.AND.X.LE.0.84) Y=(X/1.104)**(0.957)                EIN 2590
      IF (X.GT.0.84.AND.X.LE.1.10) Y=(X/1.940)**(0.310)                EIN 2600
      IF (X.GT.1.10.AND.X.LE.1.30) Y=(X/0.475)**(-0.208)               EIN 2610
      IF (X.GT.1.30.AND.X.LE.2.20) Y=(X/0.930)**(-0.633)               EIN 2620
      IF (X.GT.2.20.AND.X.LE.3.10) Y=(X/0.278)**(-0.266)               EIN 2630
      IF (X.GT.3.10) Y=0.530                                           EIN 2640
      RETURN                                                           EIN 2650
C                                                                      EIN 2660
      END                                                              EIN 2670
```

```
      SUBROUTINE FIG10 (X,Y)                                            EIN 2680
C                                                                       EIN 2690
C *** THIS SUBROUTINE APPROXIMATES EINSTEINS FIG 10 BY A SERIES OF EQNS.EIN 2700
C                                                                       EIN 2710
      IF (X.LE.0.77) Y=(7.56/X)**1.01                                   EIN 2720
      IF (X.GT.0.77.AND.X.LE.2.12) Y=(5.35/X)**1.19                     EIN 2730
      IF (X.GT.2.12.AND.X.LE.4.10) Y=(4.10/X)**1.67                     EIN 2740
      IF (X.GT.4.10.AND.X.LE.6.10) Y=(4.10/X)**2.30                     EIN 2750
      IF (X.GT.6.10.AND.X.LE.11.0) Y=(4.60/X)**3.23                     EIN 2760
      IF (X.GT.11.0.AND.X.LE.16.7) Y=(5.66/X)**4.26                     EIN 2770
      IF (X.GT.16.7.AND.X.LE.22.5) Y=(9.28/X)**7.81                     EIN 2780
      IF (X.GT.22.5) Y=(13.10/X)**12.66                                 EIN 2790
      RETURN                                                            EIN 2800
C                                                                       EIN 2810
      END                                                               EIN 2820
```

```
      SUBROUTINE POLYNML (A,Z,XI1,XI2,XJ1,XJ2)
      COMMON /CEF/ CJ0(2),CJ1(2),CJ2(2),CJ3(2),CJ(2),C1(2),C2(2),C3(2),C
     14(2),M
C
C *** COMPUTATION OF THE POLYNOMIALS WHICH APPROXIMATE THE INTEGERS I1,
C        I2, J1 AND J2
C
      IS=0
      X1=0.
      X2=0.
      IF (A.GE.0.0050.OR.Z.GE.0.8) GO TO 101
      A1=A
      A=0.0050
      IS=1
  101 CONTINUE
      IF (M.EQ.0) CALL COEF (A)
      DO 102 I=1,2
  102 CJ(I)=10.**(C1(I)+C2(I)*Z+C3(I)*Z*Z+C4(I)*Z**3)
      FACT=0.216*A**(Z-1.)/(1.-A)**Z
      IF (IS.NE.1) GO TO 103
      CALL SIMPSON (A1,0.0050,Z,X1,X2)
      A=A1
  103 CONTINUE
      XJ1=X1+CJ(1)
      XJ2=X2-CJ(2)
      XI1=FACT*XJ1
      XI2=FACT*XJ2
      RETURN
C
      END
```

```
      SUBROUTINE COEF (A)
      COMMON /CEF/ CJ0(2),CJ1(2),CJ2(2),CJ3(2),CJ(2),C1(2),C2(2),C3(2),C
     14(2),M
C
C *** COMPUTATION OF THE COEFFICIENTS OF THE POLYNOMIALS WHICH
C         APPROXIMATE THE INTEGERS I1,I2,J1,AND J2
C
      AP=ALOG(A)
      CL1=1.-A
      CL2=A-1.-A*AP
      CJ0(1)=ALOG10(CL1)
      CJ1(1)=ALOG10(-CL1-AP)
      CJ2(1)=ALOG10(1./A+2.*AP-A)
      CJ3(1)=ALOG10(1.5+A-3./A+0.5/(A*A)-3.*AP)
      CJ0(2)=ALOG10(-CL2)
      CJ1(2)=ALOG10(-(-CL2-AP**2/2.))
      CJ2(2)=ALOG10(-(-2.+(1.+AP)/A+A*(1.-AP)+AP**2))
      CJ3(2)=ALOG10(-(3.75+(AP+0.5)/(2.*A*A)-3./A*(1.+AP)+A*(AP-1.)-1.5*
     1AP**2))
      DO 101 I=1,2
         C1(I)=CJ0(I)
         C2(I)=-1.83333*CJ0(I)+3.*CJ1(I)-1.5*CJ2(I)+0.3333*CJ3(I)
         C3(I)=(2.*CJ0(I)-5.*CJ1(I)+4.*CJ2(I)-CJ3(I))/2.
         C4(I)=(-CJ0(I)+3.*CJ1(I)-3.*CJ2(I)+CJ3(I))/6.
  101 CONTINUE
      RETURN
C
      END
```

```
      SUBROUTINE SIMPSON (XM,XC,Z,XJ1,XJ2)
C
C *** THIS SUBROUTINE EVALUATES J1 AND J2 BY SIMPSONS RULE
C
      DIMENSION YI1(51), YI2(51)
      SUMI=0.
      SUMJ=0.
      XB=50.0
      XI1=0.
      XI2=0.
      XM1=0.1
      INDI=0
      IF (XM1.GT.XC) XM1=XC/10.
  101 IF (XM1.LE.XM) INDI=1
      IF (INDI.EQ.1) XM1=XM
      DX1=(XM1-XC)/XB
      NXB=XB+1.1
      DO 102 I=1,NXB
         XI=I
         X=XC+(XI-1.)*DX1
         YI1(I)=((1.-X)/X)**Z
         YI2(I)=YI1(I)*ALOG(X)
  102 CONTINUE
      NXB1=NXB-2
      DO 103 I=1,NXB1,2
         SUMI=SUMI+(YI1(I)+4.*YI1(I+1)+YI1(I+2))
         SUMJ=SUMJ+(YI2(I)+4.*YI2(I+1)+YI2(I+2))
  103 CONTINUE
      XI1=XI1+SUMI*DX1/3.
      XI2=XI2+SUMJ*DX1/3.
      IF (INDI.EQ.1) GO TO 104
      XC=XM1
      XM1=XM1/10.
      SUMI=0.0
      SUMJ=0.0
      GO TO 101
  104 CONTINUE
      XJ1=-XI1
      XJ2=-XI2
      RETURN
C
      END
```

WRITE(6,1000) XJ1, XJ2

1000 FORMAT(2E10.4)

TO J

STRANS

LISTING

```
      PROGRAM STRANS (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C
C     DEVELOPED           COLORADO STATE UNIVERSITY ENGINEERING RESEARCH
C                         CENTER, FORT COLLINS, COLORADO 80523
C     PURPOSE             COMPUTATION OF BED MATERIAL DISCHARGE BY
C                         MAHMOODS TRANPORT FUNCTION
C     REFERENCE           MAHMOOD, K., FLOW IN SAND-BED CHANNELS,
C                         CUSUSWASH WATER MANAGEMENT TECHNICAL PUBLICATION
C                         REPORT NO. 11, 1971, COLORADO STATE UNIVERSITY
C                         FORT COLLINS, COLORADO
C     CORE USAGE          CDC 6400 SCOPE 3.3 SYSTEM DEFAULT VALUE,
C                         43000 OCTAL.
C     COMPILATION TIME    APPROXIMATELY 5 SEC.
C     CENTRAL PROCESSOR
C     TIME FOR ONE RUN    LESS THAN 1 SEC.
C
C     THIS PROGRAM WILL COMPUTE SEDIMENT TRANSPORT BY INDIVIDUAL
C     FRACTIONS. IT CAN ALSO CALCULATE THE SUSPENDED SEDIMENT CONCENTRA-
C     TION FOR THE INDIVIDUAL FRACTIONS.
C
C     INPUT AND OUTPUT DESCRIPTION
C     INPUT CONSISTS OF THE FOLLOWING, IN THE ORDER SHOWN
C     1) INTEGERS NN AND JJ, TO BE READ IN FORMAT 2I10
C        NN IS AN INPUT INDICATOR
C        IF NN=2, THE NUMBER OF SIZE FRACTIONS IS 1
C        IF NN=6, THE NUMBER OF SIZE FRACTIONS IS 5
C        IF NN=11, THE NUMBER OF SIZE FRACTIONS IS 10
C        JJ IS AN OUTPUT INDICATOR
C        IF JJ=1, INTERMEDIATE RESULTS ARE PRINTED OUT
C        IF JJ=2, INTERMEDIATE RESULTS ARE OMITTED FROM THE OUTPUT
C     2) INPUT VARIABLES V,D,SE,VNU,D50,SDD, TO BE READ IN FORMAT 6F10.0
C        THE TRANSPORT COMPUTATIONS ARE FOR A UNIT WIDTH
C        FOR NON-RECTANGULAR SECTIONS, USE HYDRAULIC DEPTH
C        V    IS AVERAGE VELOCITY IN FT./SEC.
C        D    IS HYDRAULIC DEPTH IN FT.
C        SE   IS ENERGY GRADIENT IN FT./FT.
C        VNU IS KINEMATIC VISCOSITY IN SQ.FT./SEC.
C        D50 IS MEDIAN BED MATERIAL SIZE IN MM.
C        SDD IS GRADATION COEFFICIENT
C
C
      DIMENSION XX(3,11), XX1(3,11), YY1(3,11), NPP(3)
      COMMON /SDDATA/ X(11),X1(11),Y1(11),PG,GX,NP
      COMMON /BDATA/ II,JJ,NN,D,V,SE,VNU,D50,SDD,GT,PPM,SV,AE,QPB,QPT,CR
     1EF
      COMMON /ZDATA/ Z(25),ZX1(25),ZX2(25),ZI,XI1,XI2
      COMMON /SNDATA/ FX(45),VX(45)
      DATA (FX(I),I=1,45)/49.99,49.98,49.97,49.96,49.95,49.90,49.85,49.8
     10,49.70,49.60,49.51,49.40,49.29,49.20,49.01,48.81,48.61,48.30,47.9
     28,47.50,46.99,46.41,45.73,44.95,44.06,43.06,41.92,40.66,39.25,37.7
     30,35.54,34.13,31.59,28.81,25.80,22.57,19.15,15.54,11.79,7.93,3.98,
     43.19,1.99,.40,.00/
      DATA (VX(I),I=1,45)/3.62,3.47,3.39,3.32,3.27,3.08,2.96,2.88,2.75,2
     1.65,2.58,2.51,2.45,2.41,2.33,2.26,2.20,2.12,2.05,1.96,1.88,1.80,1.
     272,1.64,1.56,1.48,1.40,1.32,1.24,1.16,1.06,1.00,.90,.80,.70,.60,.5
     30,.40,.30,.20,.10,.08,.05,.01,.00/
C
C     FX,VX ARE STND. NORMAL DISTN FX,X.
```

```
C
      NPP(1)=8
      NPP(2)=11
      NPP(3)=8
      DATA (XX(1,I),I=1,11)/+1.0,-1.0,9*0.0/
      DATA (XX(2,I),I=1,11)/1.65,.84,.253,-.253,-.84,-1.65,5*0.0/
      DATA (XX(3,I),I=1,11)/2.33,1.282,.842,.524,.253,.000,-.253,-.524,-
     10.842,-1.282,-2.330/
      DATA (XX1(1,I),I=1,8)/0.008,.000001,.05,.001,.1,.001,.2,.045/,(YY1
     1(1,I),I=1,8)/.70,.30,4.8,3.0,10.0,6.5,100.0,80.0/
      DATA (XX1(2,I),I=1,11)/2.00E-02,1.00E-05,4.200E-02,2.00E-04,6.70E-
     102,1.000E-03,1.09E-01,4.00E-03,2.13E-01,2.00E-02,4.00E-01/,(YY1(2,
     2I),I=1,11)/7.00E-02,8.20E-01,2.00E-01,3.30E+00,1.00E+00,1.00E+01,3
     3.20E+00,3.30E+01,1.00E+01,1.00E+02,3.40E+01/
      DATA (XX1(3,I),I=1,8)/.01,.000001,.034,.0001,.07,.001,.18,.01/,(YY
     11(3,I),I=1,8)/.40,.033,1.1,.20,7.50,2.0,100.0,50.0/
      DATA (Z=.001,.050,.100,.150,.200,.300,.400,.500,.600,.700,.800,1.0
     100,1.200,1.400,1.600,1.800,2.000,2.500,3.000,3.500,4.000,5.000,6.0
     200,7.000,10.000)
      DATA (ZX1=.84792E+00,.83006E+00,.81590E+00,.80545E+00,.79840E+00,.
     179349E+00,.79965E+00,.81600E+00,.84209E+00,.87786E+00,.92357E+00,.
     210471E+01,.12200E+01,.14539E+01,.17655E+01,.21779E+01,.27224E+01,.
     349794E+01,.95636E+01,.19018E+02,.38805E+02,.17061E+03,.78860E+03,.
     437708E+04,.46666E+06)
      DATA (ZX2=-.56570E+00,-.57938E+00,-.59482E+00,-.61182E+00,-.63042E
     1+00,-.67275E+00,-.72247E+00,-.78039E+00,-.84749E+00,-.92493E+00,-.
     210141E+01,-.12341E+01,-.15238E+01,-.19049E+01,-.24071E+01,-.30699E
     3+01,-.39472E+01,-.76175E+01,-.15188E+02,-.31034E+02,-.64623E+02,-.
     429226E+03,-.13761E+04,-.66653E+04,-.84368E+06)
C
C     J=1,2,3 FOR 1,5 AND 10 FRACTIONS.
C     X,XX RELATE TO NORMAL DEVIATES, XX1,X1,YY1,Y1 TO PG-GT CURVE
C
      READ (5,104) NN,JJ
      J=NN/4+1
      NP=NPP(J)
      DO 101 I=1,NN
  101 X(I)=XX(J,I)
      DO 102 I=1,NP
         X1(I)=XX1(J,I)
  102 Y1(I)=YY1(J,I)
C     NP= NO OF POINTS ON PG-GT CURVE
C
      READ (5,103) V,D,SE,VNU,D50,SDD
      WRITE (6,105)
      WRITE (6,106) V,D,SE,VNU,D50,SDD
      D50=D50/304.8
      VNU=VNU*100000.
      CALL TPORT
      STOP
C
  103 FORMAT (8F10.3)
  104 FORMAT (2I10)
  105 FORMAT (1H1,//9X, 74HCOMPUTATION OF TOTAL BED MATERIAL DISCHARGE B
     1Y MAHMOODS TRANSPORT FUNCTION,//)
  106 FORMAT (5X, 29HAVERAGE VELOCITY             ,F12.2, 12H FT./SEC.
     1 ,/5X, 29HHYDRAULIC DEPTH              ,F12.2, 12H FT.         ,/5X
     2, 29HWATER SURFACE WIDTH          ,8X, 15H1.00 FT.        ,/5X, 29H
     3ENERGY GRADIENT              ,F12.7, 12H FT./FT.     ,/5X, 29HKINEM
```

```
     4ATIC VISCOSITY          ,F12.7, 12H SQ.FT./SEC.,/5X, 29HMEDIAN BEDSTR 1190
     5 MATERIAL SIZE      ,F12.2, 11H MM.        ,/5X, 29HGRADATION COEFFISTR 1200
     6CIENT         ,F12.2,//)                                            STR 1210
C                                                                         STR 1220
      END                                                                 STR 1230
```

```
      SUBROUTINE TPORT
      COMMON /ADATA/ DM84,DM50,DM16,SIGMA
      COMMON /BDATA/ II,JJ,NN,D,V,SE,VNU,D50,SDD,GT,PPM,SV,AE,QPB,QPT,CR
     1EF
      COMMON /ZDATA/ Z(25),ZX1(25),ZX2(25),ZI,XI1,XI2
      COMMON /SDDATA/ X(11),X1(11),Y1(11),PG,GX,NP
      DIMENSION PIF(11), DMM(11), A(11), GIF(11)
      DIMENSION DM(10), FV(10), H(10), A1(10), B1(10), C1(10), D1(10), D
     1ELS(10), DELB(10), ETA1(10), DTAU(10), TAUC(10), DF(11)
      DIMENSION PPMFT(10), PPMF(10)
      COMMON GPS(10),GPB(10),GPT(10),U(10,60),C(10,60),ET(10,60),SUMQ(10
     1,60),SUMG(10,60),ETD(10,60),G(10,60),GB(10),QB(10),N(10),GL(10),QL
     22(10),GU(10),QU(10),DELG(10),DELQ(10),PFG(10),DPM(10)
      RVS(DM,VN)=(1./DM)*(SQRT(35.43*DM**3+3.6E-9*VN**2)-6.E-5*VN)
C
C     RVS(DIAMETER IN FT,KIN. VISCOSITY IN SQ.FT/SEC*1.E+5) IS RUBEYS FA
C     VELOCITY IN FT/SEC
C
      SFA(DR,ETA,Z)=(ALOG(33.35*DR)/(1.-Z))*((ETA/.15)**Z*.15-ETA)
      SFB(ETA,Z)=((-.2846*(ETA/.15)**Z-ETA*ALOG(ETA))/(1.-Z))-((.15*(ETA
     1/.15)**Z-ETA)/((1.-Z)**2))
      SFD(ETA,Z)=(ETA/.85)**Z
      SFC(DR,AE)=(ALOG(5.*DR))/AE+1.897
      SFA1(ETA,DR)=ETA*ALOG(33.35*DR)*ALOG(.15/ETA)
      SFB1(ETA)=ALOG(.15*ETA)*ALOG(.15/ETA)*ETA/2.
      SFPB(DB,AE,D84)=DB*ALOG(33.35*DB/D84)/AE
C
C     DM IS GEOMETRIC MEAN DIA OF A FRACTION. 1 IS LARGEST.
C     DR IS DEPTH D/D84.     Z IS ROUSE NO
C     ETA IS   2*DM(I)/D     AE IS U*0/U*0E
C
      II=1
      SV=SQRT(32.2*D*SE)
      TAU=62.4*D*SE
      D84=D50*SDD
      DR=D/D84
      VP=2.50*ALOG(12.27*DR)
      VF=V/SV
      SVE=SV*(VF-2.62)/(VP-2.62)
      IF (SVE.GE.SV) SVG=.99*SV
      IF (SVE.LE.0.) SVG=.01*SV
      AE=SV/SVE
      N1=NN-1
      XN1=N1
      W50=RVS(D50,VNU)
      FVRG=W50/SVE
      SH=SVE**2/(53.1*D50)
      IF (JJ.EQ.2) GO TO 101
      WRITE (6,119)
  101 CONTINUE
      DO 102 I=1,NN
         DF(I)=D50*SDD**X(I)
  102 CONTINUE
      DO 103 I=1,N1
  103 N(I)=I
      DO 109 I=1,N1
         DM(I)=SQRT(DF(I)*DF(I+1))
         DMM(I)=DM(I)*305.
         ETA1(I)=2.*DM(I)/D
```

```
         DI=DM(I)
         FV(I)=RVS(DI,VNU)
         H(I)=FV(I)/(0.40*SV)
C
C     COMPUTATION OF CRITICAL SHEAR AND WEIGHTING FACTOR
C
         IF (DM(I)-0.0009) 104,105,105
  104    TAUC(I)=0.0215*(DM(I)**.25)
         GO TO 108
  105    IF (DM(I)-0.0018) 106,107,107
  106    TAUC(I)=0.315*(DM(I)**.633)
         GO TO 108
  107    CONTINUE
         TAUC(I)=16.8*(DM(I)**1.262)
  108    DTAU(I)=1.-TAUC(I)/TAU
         IF (DTAU(I).LE.0.) DTAU(I)=0.
  109 CONTINUE
      QPB=0.
      QPS=0.
      DO 113 I=1,N1
         ZI=H(I)
         CALL ZPOLATE
         ETA=ETA1(I)
         DB=2.*DM(I)
C
C     ZPOLATE YIELDS XI1 AND XI2
C
         C1(I)=SFC(DR,AE)
         D1(I)=SFD(ETA,ZI)
         IF (ZI.EQ.1.) GO TO 110
         A1(I)=SFA(DR,ETA,ZI)
         B1(I)=SFB(ETA,ZI)
         GO TO 111
  110    A1(I)=SFA1(ETA,DR)
         B1(I)=SFB1(ETA)
  111    CONTINUE
C
C     COMPUTATION OF SUSPENDED AND BEDLOAD FRACTIONWISE DISCHARGE
C
         DELS(I)=D*((A1(I)+B1(I))/AE+XI1*C1(I)*D1(I)+XI2*D1(I))*DTAU(I)/
     1   XN1
         DELB(I)=SFPB(DB,AE,D84)*DTAU(I)/XN1
         QPS=QPS+DELS(I)
         QPB=QPB+DELB(I)
         IF (JJ.EQ.2) GO TO 112
         WRITE (6,121) I,DM(I),ETA1(I),H(I),A1(I),B1(I),C1(I),D1(I),XI1,
     1   XI2,DTAU(I)
  112    CONTINUE
  113 CONTINUE
      QPT=QPB+QPS
      WRITE (6,120)
      DO 114 I=1,N1
         GPS(I)=DELS(I)/QPS
         GPB(I)=DELB(I)/QPB
         GPT(I)=(DELS(I)+DELB(I))/QPT
         WRITE (6,121) I,DMM(I),FV(I),H(I),TAUC(I),DTAU(I),DELS(I),DELB(
     1   I),GPS(I),GPB(I),GPT(I)
  114 CONTINUE
      PG=414.*FVRG*SV*QPT*SH**.75
```

```
      CALL GPOLATE
      GT=GX
      PPM=GT*1.E6/(62.4*V*D)
      CREF=GT/(2.5*SV*QPT)
      SUMT=0.
      DO 115 I=1,N1
         J=N1-I+1
         SUMT=SUMT+GPT(J)*100.
         PFG(J)=SUMT
         PPMFT(J)=PPM*PFG(J)/100.
         PPMF(J)=PPM*GPT(J)
         GIF(J)=GT*GPT(J)
         PIF(J)=PPMF(J)
         A(J)=GPT(J)*100.
  115 CONTINUE
      WRITE (6,118)
      DO 116 I=1,N1
         WRITE (6,117) N(I),DM(I),PFG(I),DMM(I),PPMF(I),PPMFT(I)
  116 CONTINUE
      CALL LNORM (N1,DMM,PFG)
      WRITE (6,122) DM84,DM50,DM16,SIGMA
      RETURN
C
  117 FORMAT (5X,I3,3X,E10.3,8X,F6.2,9X,F7.3,3X,E10.3,6X,E10.3)
  118 FORMAT (/5X,5HF NO.,4X,7HSIZE-FT,3X,18HPERCENT FINER THAN,3X,7HSIZ
     1E-MM,3X,11HPPM IN FRAC,2X,17HPPM IN FINER THAN/)
  119 FORMAT (/6X, 63HFRACTION-WISE VALUES OF COMPUTATIONAL PARAMETERS A
     1RE AS FOLLOWS,/5X,5HF NO.,4X,7HSIZE-FT,3X,9HETA=DM /D,3X,9HROUSE N
     20.,3X,9HPARAM. A1,3X,9HPARAM. B1,3X,9HPARAM. C1,3X,9HPARAM. D1,3X,
     310HINTEGRAL I,2X,10HINTEGRAL J,3X,8HDEL. TAU/)
  120 FORMAT (/6X, 36HFRACTION-WISE ANALYSIS IS AS FOLLOWS,/5X,5HF NO.,4
     1X,7HSIZE-MM,4X,8HFALL VEL,3X,9HROUSE NO.,3X,10HCRIT.SHEAR,2X,10HWE
     2IGH.FACT,3X,9HDEL.SUSP.,4X,8HDEL. BED,3X,10HFRAC.IN GS,2X,10HFRAC.
     3IN GB,2X,10HFRAC.IN GT/)
  121 FORMAT (5X,I3,3X,10(E10.3,2X))
  122 FORMAT (/5X, 21HFOR B.M. TRANSP. D84=,F8.4, 11H  MM,  D50=,F8.4, 1
     10H MM   D16=,F8.4, 14H MM AND SIGMA=,F8.4)
C
      END
```

```
      SUBROUTINE GPOLATE
      COMMON /SDDATA/ X(11),X1(11),Y1(11),PG,GT,N
      IF (PG-X1(1)) 101,101,102
  101 GT=Y1(1)
      GO TO 107
  102 IF (PG-X1(N)) 104,103,103
  103 GT=Y1(N)
      GO TO 107
  104 CONTINUE
      DO 106 J=1,11
         IF ((PG.GT.X1(J)).AND.(PG.LE.X1(J+1))) GO TO 105
         GO TO 106
  105    I=J+1
C
C     THIS SUBROUTINE IS FOR LOG-LOG INTERPOLATION
C
         A=ALOG(Y1(I)/Y1(I-1))
         B=ALOG(X1(I)/X1(I-1))
         C=ALOG(PG/X1(I-1))
         GT=Y1(I-1)*(EXP(A*C/B))
         GO TO 107
  106 CONTINUE
  107 CONTINUE
      RETURN
C
      END
```

```
      SUBROUTINE ZPOLATE
      COMMON /ZDATA/ 7(25),XI1(25),XI2(25),Z1,XII,XIJ
      IF (Z1.GT.0.001) GO TO 101
      XII=0.85
      XIJ=-.85-.15*ALOG(0.15)
      GO TO 104
  101 CONTINUE
      DO 103 J=1,24
         IF ((Z1.GT.Z(J)).AND.(Z1.LE.Z(J+1))) GO TO 102
         GO TO 103
  102    I=J+1
C
C     IF SEMILOG PLOT IS LINEAR
C
         A=(Z1-Z(I-1))/(Z(I)-Z(I-1))
         B=XI1(I)/XI1(I-1)
         C=XI2(I)/XI2(I-1)
         XIJ=XI2(I-1)*(C**A)
         XII=XI1(I-1)*(B**A)
         GO TO 104
  103 CONTINUE
  104 CONTINUE
      RETURN
C
      END
```

```
      SUBROUTINE LNORM (N,X,P)
C
C     THIS WILL DETERMINE LOG NORMAL DISTRIBUTION  PARMETERS.
C     N=NO OF POINTS IN X ARRAY FOR WHICH P ARE CDF.STARTING WITH HIGHES
C     X IS FIRST CONVERTED TO NATURAL LOG.
C     IT WILL ALSO DETERMINE ANY OTHER  PERCENTILE SIZES FOR WHICH
C     NO IS NO OF SUCH POINTS,PO  ARE PERCENTILES AND XO ARE READ. SIZES
C     SET NO=3  UNLES XO ARE REQUIRED. PO(1)=84,PO(2)=50,PO(3)=16 ALWAYS
C     YO ARE DEVIATIONS OF PO  YO(1)=1.,YO(2)=0.,YO(3)=-1.
C     IF RANGE OF P .NOT. 15.LT.P.GT.85   ONLY P=50 - P85 USED
C     IF            .NOT. 50.LT.P.GT.85   IT WILL NOT ESTIMATE PARAMETER
C     BUT  WILL ESTIMATE XO  IF  NO.NE.0.
C
      COMMON /SNDATA/ FX(45),XX(45)
      COMMON /ADATA/ XO(3),SIGMA
      DIMENSION X(N), P(N), PO(10), YO(10), IND(15), Q(15), Z(15), Y(15)
C
C     FIRST ELIMINATE P.LT.0.01  AND  P.GT.99.99  AND DETERMINE DEVIATES
C
      NO=3
      PO(1)=84.
      PO(2)=50.
      PO(3)=16.
      YO(1)=1.
      YO(2)=0.
      YO(3)=-1.
      IN=0
      DO 104 I=1,N
         IF ((P(I).GE.99.99).OR.(P(I).LE.0.01)) GO TO 104
         IN=IN+1
         Q(IN)=P(I)-50.
         Z(IN)=X(I)
         QQ=ABS(Q(IN))
         DO 102 J=1,45,4
            IF (QQ.LT.FX(J)) GO TO 102
            J1=J-4
            J2=J
            DO 101 K=J1,J2
               IF (QQ.LT.FX(K)) GO TO 101
               J3=K-1
               J4=K
               Y(IN)=XX(J4)+(QQ-FX(J4))*(XX(J3)-XX(J4))/(FX(J3)-FX(J4))
               GO TO 103
  101       CONTINUE
  102    CONTINUE
  103    IF (Q(IN).LT.0.) Y(IN)=-Y(IN)
  104 CONTINUE
C
C     NOW TAKE LOG OF X
C
      DO 105 J=1,IN
  105 Z(J)=ALOG(Z(J))
      P1=Q(1)+50.
      P2=Q(IN)+50.
      DO 108 J=1,NO
         IF ((PO(J).LE.P1).AND.(PO(J).GE.P2)) GO TO 106
         GO TO 107
  106    IND(J)=1
         GO TO 108
```

```
  107    IND(J)=0
  108 CONTINUE
C
C     PERCENTILES ARE COMPUTED NEXT WHERE POSSIBLE
C
      DO 113 J=1,NO
         IF (IND(J).EQ.0) GO TO 113
         DO 109 I=1,IN
            IF (Y(I).LE.YO(J)) GO TO 110
  109    CONTINUE
         I=IN
  110    IF (I-1) 112,111,112
  111    XO(J)=EXP(Z(1))
         GO TO 113
  112    K=I-1
         XO(J)=Z(I)+(YO(J)-Y(I))*(Z(K)-Z(I))/(Y(K)-Y(I))
         XO(J)=EXP(XO(J))
  113 CONTINUE
      DO 115 J=1,NO
         IF (IND(J).EQ.0) GO TO 114
         GO TO 115
  114    CONTINUE
         IF (PO(J).GT.P1) XO(J)=EXP(Z(1)+(YO(J)-Y(1))*(Z(1)-Z(2))/(Y(1)-
     1   Y(2)))
         IF (PO(J).LT.P2) XO(J)=EXP(Z(IN)+(YO(J)-Y(IN))*(Z(IN-1)-Z(IN))/
     1   (Y(IN-1)-Y(IN)))
  115 CONTINUE
      SIGMA=0.5*(XO(1)/XO(2)+XO(2)/XO(3))
      RETURN
C
C
      END
```

COLBY

LISTING

```
      PROGRAM COLBY (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C
C     DEVELOPED          COLORADO STATE UNIVERSITY ENGINEERING RESEARCH
C                        CENTER, FORT COLLINS,COLORADO 80523
C     PURPOSE            COMPUTATION OF BED MATERIAL LOAD BY COLBYS
C                        METHOD
C     REFERENCE          COLBY,B.R., DISCHARGE OF SANDS AND MEAN VFLOCITY
C                        RELATIONSHIPS IN SAND-BED STREAMS, PROFESSIONAL
C                        PAPER 462-A, 1964, U.S. GEOLOGICAL SURVEY.
C     CORE USAGE         CDC 6400 SCOPE 3.3 SYSTEM DEFAULT VALUE,
C                        43000 OCTAL.
C     COMPILATION TIME   APPROXIMATELY 4 SEC.
C     CENTRAL PROCESSOR
C     TIME FOR ONE
C     SET OF DATA        LESS THAN 0.6 SEC.
C
C     INPUT AND OUTPUT DESCRIPTION
C
C     THE FIRST CARD IN THE INPUT LOGICAL RECORD SHOULD CONTAIN THE
C     VALUE OF NDATA, IN FORMAT I5. NDATA IS THE NUMBER OF SETS OF INPUT
C     DATA TO BE FED TO THE COMPUTER AT A TIME. A SET OF INPUT DATA
C     CONSISTS OF A GROUP OF VARIABLES NECESSARY TO SPECIFY A PROBLEM,
C     AS DETAILED BELOW.
C
C     THE FIRST CARD IN INPUT IS FOLLOWED BY THE SETS OF INPUT DATA,
C     TO BE PUNCHED IN FORMAT 6F10.0
C     A SET OF INPUT DATA CONSISTS OF THE FOLLOWING VARIABLES.
C     1) AVERAGE VELOCITY                V              F.P.S.
C     2) HYDRAULIC DEPTH                 D              FT.
C     3) WATER SURFACE WIDTH             W              FT.
C     4) TEMPERATURE                     TF             DEG.FAHREN.
C     5) MEDIAN BED MATERIAL SIZE        D50            MM.
C     6) FINE MATERIAL CONCENTRATION     FML            PPM.
C
C     OUTPUT CONSISTS OF THE TOTAL BED MATERIAL TRANSPORT IN TONS/DAY,
C     AND A REMARK ON HOW THE COMPUTATIONS WERE CARRIED OUT.
C     IF  REMARK= OK, THE COMPUTATIONS WERE CARRIED OUT SUCCESSFULLY.
C     IF  REMARK= OOR, VELOCITY, DEPTH OR BED MATERIAL SIZE IS OUT OF
C     RANGE.
C     IF  REMARK= TOOR, TEMPERATURE IS OUT OR RANGE.
C     IF  REMARK= FOOR, FINE MATERIAL CONCENTRATION IS OUT OF RANGE.
C     VARIABLE                           RANGE
C     AVERAGE VELOCITY                   1-10  F.P.S.
C     HYDRAULIC DEPTH                    1-100  FT.
C     WATER SURFACE WIDTH
C     TEMPERATURE                        32-100  DEG.FAHREN.
C     MEDIAN BED MATERIAL SIZE           0.1-0.8  MM.
C     FINE MATERIAL CONCENTRATION        0-200000 PPM.
C
C
      COMMON /CLBY/ G(4,8,6),F(5,10),T(7,4),P(11),DF(10),CF(5),DP(11),DG
     1(4),VG(8),D50G(6),TEMP(7)
      DIMENSION II(2), JJ(2), KK(2), XX(2), YY(2), ZZ(2), X(2,2), XA(2),
     1 XG(2), XT(2,2), XCT(2), XF(2,2)
      WRITE (6,159)
      READ (5,162) NDATA
      DO 157 L=1,NDATA
         READ (5,163) V,D,W,TF,D50,FML
```

```
      WRITE (6,160) L,V,D,W,TF,D50,FML
      DATA (((G(I,J,K),I=1,4),J=1,8),K=1,3)/1.0,0.30,0.06,0.0,3.0,3.3
     10,2.50,2.0,5.4,9.00,10.00,20.00,11.0,26.00,50.00,150.00,17.0,49
     2.00,130.00,500.0,29.0,101.00,400.00,1350.0,44.0,160.00,700.00,2
     3500.0,60.0,220.0,1000.0,4400.0,.38,0.06,0.00,0.00,1.60,1.20,.65
     4,.10,3.70,5.00,4.00,3.00,10.00,18.00,30.00,52.00,17.00,40.00,80
     5.00,160.00,36.00,95.00,230.00,650.00,60.00,150.00,415.00,1200.0
     6,81.00,215.00,620.00,1500.0,0.14,0.0,0.0,0.0,1.00,0.60,.15,0.0,
     73.30,3.00,1.7,0.5,11.00,15.00,17.0,14.0,20.00,35.00,49.0,70.0,4
     84.00,85.00,150.0,250.0,71.00,145.00,290.0,500.0,100.00,202.00,4
     900.0,700.0/
      DATA (((G(I,J,K),I=1,4),J=1,8),K=4,6)/0.0,0.0,0.00,0.00,0.70,0.
     130,0.06,0.00,2.90,2.30,1.00,0.06,11.50,13.00,12.00,7.00,22.00,3
     21.00,40.00,50.00,47.00,84.00,135.00,210.00,75.00,140.00,240.00,
     3410.00,106.00,190.00,350.00,630.00,0.00,0.0,0.00,0.0,0.44,0.06,
     40.00,0.0,2.80,1.80,0.60,0.0,12.00,12.50,10.00,4.5,24.00,30.00,3
     55.00,37.0,52.00,78.00,120.00,190.00,83.00,180.00,215.00,380.0,1
     620.00,190.00,305.00,550.00,0.0,0.0,0.0,0.0,0.3,0.0,0.0,0.0,2.9,
     71.4,0.3,0.0,14.0,11.0,7.7,3.0,27.0,29.0,30.0,30.0,57.0,75.0,110
     8.0,170.0,90.0,140.0,200.0,330.0,135.0,190.0,290.0,520.0/
      DATA ((F(I,J),I=2,5),J=1,10)/1.10,1.60,2.60,4.20,1.10,1.65,2.75
     1,4.90,1.10,1.70,3.00,5.50,1.12,1.90,3.60,7.00,1.17,2.05,4.30,8.
     270,1.20,2.30,5.50,11.20,1.22,2.75,8.00,22.00,1.25,3.00,9.60,29.
     300,1.30,3.50,12.00,43.00,1.40,4.90,22.00,120.0/
      DATA (F(1,I),I=1,10)/1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0/
      DATA ((T(I,J),I=1,7),J=1,4)/1.20,1.15,1.10,0.96,0.90,0.85,0.82,
     11.35,1.25,1.12,0.92,0.86,0.80,0.75,1.60,1.40,1.20,0.89,0.80,0.7
     22,0.66,2.00,1.65,1.30,0.85,0.72,0.63,0.55/
      DATA (DF(I),I=1,10)/0.10,0.20,0.30,0.60,1.00,2.00,6.00,10.00,20
     1.00,1.E2/,(CF(I),I=1,5)/9.0,1.E4,5.E4,1.E5,1.5E5/
      DATA (P(I),I=1,11)/0.60,0.90,1.0,1.0,0.83,0.60,0.40,0.25,0.15,0
     1.09,0.05/,(DP(I),I=1,11)/0.10,0.15,0.20,0.30,0.40,0.50,0.60,0.7
     20,0.80,0.90,1.00/,(DG(I),I=1,4)/0.10,1.0,10.0,100.0/,(VG(I),I=1
     3,8)/1.0,1.5,2.0,3.0,4.0,6.0,8.0,10.0/,(D50G(I),I=1,6)/0.10,0.20
     4,0.30,0.40,0.60,0.80/,(TEMP(I),I=1,7)/32.0,40.0,50.0,70.0,80.0,
     590.0,100.0/
      REMARK=5HOK
      IF ((D50.LT.D50G(1)).OR.(D50.GT.D50G(6))) GO TO 101
      GO TO 102
  101 REMARK=5HOOR
      GO TO 155
C
C     LOCATE APPROPRIATE   V,D,D50 GRID
C
  102 CONTINUE
      IF ((D.LT.DG(1)).OR.(D.GT.DG(4))) GO TO 103
      GO TO 104
  103 REMARK=5HOOR
      GO TO 155
  104 IF ((V.LT.VG(1)).OR.(V.GT.VG(8))) GO TO 105
      GO TO 106
  105 REMARK=5HOOR
      GO TO 155
  106 IF (TF.EQ.0.) TF=60.
      IF (TF-32.) 107,108,108
  107 REMARK=5HTOOR
      TF=32.
  108 IF (TF-100) 110,110,109
  109 REMARK=5HTOOR
```

```
      TF=100.                                                           COL 1190
110   CONTINUE                                                          COL 1200
      IF1=0                                                             COL 1210
      ID2=0                                                             COL 1220
      DO 113 I=1,3                                                      COL 1230
         IF ((D.GE.DG(I)).AND.(D.LE.DG(I+1))) GO TO 111                 COL 1240
         GO TO 112                                                      COL 1250
111      ID1=I                                                          COL 1260
         ID2=I+1                                                        COL 1270
         GO TO 114                                                      COL 1280
112      CONTINUE                                                       COL 1290
113   CONTINUE                                                          COL 1300
114   IV1=0                                                             COL 1310
      IV2=0                                                             COL 1320
      DO 117 I=1,7                                                      COL 1330
         IF ((V.GE.VG(I)).AND.(V.LE.VG(I+1))) GO TO 115                 COL 1340
         GO TO 116                                                      COL 1350
115      IV1=I                                                          COL 1360
         IV2=I+1                                                        COL 1370
         GO TO 118                                                      COL 1380
116      CONTINUE                                                       COL 1390
117   CONTINUE                                                          COL 1400
118   ID501=0                                                           COL 1410
      ID502=0                                                           COL 1420
      DO 121 I=1,5                                                      COL 1430
         IF ((D50.GE.D50G(I)).AND.(D50.LE.D50G(I+1))) GO TO 119         COL 1440
         GO TO 120                                                      COL 1450
119      ID501=I                                                        COL 1460
         ID502=I+1                                                      COL 1470
         GO TO 122                                                      COL 1480
120      CONTINUE                                                       COL 1490
121   CONTINUE                                                          COL 1500
122   CONTINUE                                                          COL 1510
      II(1)=ID1                                                         COL 1520
      II(2)=ID2                                                         COL 1530
      JJ(1)=IV1                                                         COL 1540
      JJ(2)=IV2                                                         COL 1550
      KK(1)=ID501                                                       COL 1560
      KK(2)=ID502                                                       COL 1570
      DO 130 I=1,2                                                      COL 1580
         I1=II(I)                                                       COL 1590
         XX(I)=ALOG10(DG(I1))                                           COL 1600
         DO 129 J=1,2                                                   COL 1610
            J1=JJ(J)                                                    COL 1620
            YY(J)=ALOG10(VG(J1))                                        COL 1630
         DO 129 K=1,2                                                   COL 1640
            K1=KK(K)                                                    COL 1650
            ZZ(K)=ALOG10(D50G(K1))                                      COL 1660
            IF (G(I1,J1,K1)-0.) 123,123,127                             COL 1670
123         DO 125 J3=J1,7                                              COL 1680
               IF (G(I1,J3,K1)-0.) 124,124,126                          COL 1690
124            CONTINUE                                                 COL 1700
125         CONTINUE                                                    COL 1710
126         X(J,K)=ALOG10(G(I1,J3,K1))+(ALOG10(VG(J1)/VG(J3)))*(ALOG1COL 1720
     1      0(G(I1,J3+1,K1)/G(I1,J3,K1)))/(ALOG10(VG(J3+1)/VG(J3)))     COL 1730
            GO TO 128                                                   COL 1740
127         CONTINUE                                                    COL 1750
            X(J,K)=ALOG10(G(I1,J1,K1))                                  COL 1760
128         CONTINUE                                                    COL 1770
```

```
  129         CONTINUE
              XD=ALOG10(D50)-ZZ(1)
              XN1=X(1,2)-X(1,1)
              XN2=X(2,2)-X(2,1)
              XDEN=ZZ(2)-ZZ(1)
              XA(1)=X(1,1)+XN1*XD/XDEN
              XA(2)=X(2,1)+XN2*XD/XDEN
              XNM=XA(2)-XA(1)
              XV=ALOG10(V)-YY(1)
              XDY=YY(2)-YY(1)
              XG(I)=XA(1)+XNM*XV/XDY
  130     CONTINUE
          XNM=XG(2)-XG(1)
          XD=ALOG10(D)-XX(1)
          XDEN=XX(2)-XX(1)
          GTUC=XG(1)+XNM*XD/XDEN
          GTUC=10.**GTUC
C
C     GTUC  IS UNCORRECTED GT IN LB/SEC/FT
C
C
C     NEXT APPLY  F.M.LOAD AND  TEMPERATURE CORRECTIONS
C
          IF (TF-60.) 132,131,132
  131     CFT=1.
          GO TO 137
  132     CONTINUE
          IT1=0
          IT2=0
          DO 135 I=1,6
              IF ((TF.GE.TEMP(I)).AND.(TF.LE.TEMP(I+1))) GO TO 133
              GO TO 134
  133         IT1=I
              IT2=I+1
              GO TO 136
  134         CONTINUE
  135     CONTINUE
  136     CONTINUE
          XT(1,1)=ALOG10(T(IT1,ID1))
          XT(2,1)=ALOG10(T(IT2,ID1))
          XT(1,2)=ALOG10(T(IT1,ID2))
          XT(2,2)=ALOG10(T(IT2,ID2))
          XNT=ALOG10(TF/TEMP(IT1))/ALOG10(TEMP(IT2)/TEMP(IT1))
          XCT(1)=XT(1,1)+XNT*(XT(2,1)-XT(1,1))
          XCT(2)=XT(1,2)+XNT*(XT(2,2)-XT(1,2))
          CFT=XCT(1)+(XCT(2)-XCT(1))*XD/XDEN
          CFT=10.**CFT
C
C     FINE MATERIAL LOAD  CORRECTION
C
  137     CONTINUE
          IF (FML-10.) 138,138,139
  138     CFF=1.
          GO TO 149
  139     CONTINUE
          IF (FML.GT.1.E+5) REMARK=5HFOOR
          ID1=0
          ID2=0
          DO 141 I=1,9
```

```
         IF ((D.GE.DF(I)).AND.(D.LE.DF(I+1))) GO TO 140                COL 2370
         GO TO 141                                                     COL 2380
140      ID1=I                                                         COL 2390
         ID2=I+1                                                       COL 2400
         GO TO 142                                                     COL 2410
141   CONTINUE                                                         COL 2420
142   CONTINUE                                                         COL 2430
      IF (REMARK.EQ.5HFOOR   )143,144                                  COL 2440
143   IF1=4                                                            COL 2450
      IF2=5                                                            COL 2460
      GO TO 148                                                        COL 2470
144   CONTINUE                                                         COL 2480
      IF1=0                                                            COL 2490
      IF2=0                                                            COL 2500
      DO 147 I=1,4                                                     COL 2510
         IF ((FML.GE.CF(I)).AND.(FML.LE.CF(I+1))) GO TO 145            COL 2520
         GO TO 146                                                     COL 2530
145      IF1=I                                                         COL 2540
         IF2=I+1                                                       COL 2550
         GO TO 148                                                     COL 2560
146      CONTINUE                                                      COL 2570
147   CONTINUE                                                         COL 2580
148   CONTINUE                                                         COL 2590
      XF(1,1)=ALOG10(F(IF1,ID1))                                       COL 2600
      XF(2,2)=ALOG10(F(IF2,ID2))                                       COL 2610
      XF(1,2)=ALOG10(F(IF1,ID2))                                       COL 2620
      XF(2,1)=ALOG10(F(ID2,ID1))                                       COL 2630
      XNT=(FML-CF(IF1))/(CF(IF2)-CF(IF1))                              COL 2640
      XCT(1)=XF(1,1)+XNT*(XF(2,1)-XF(1,1))                             COL 2650
      XCT(2)=XF(1,2)+XNT*(XF(2,2)-XF(1,2))                             COL 2660
      XNT=ALOG10(D/DF(ID1))/ALOG10(DF(ID2)/DF(ID1))                    COL 2670
      CFF=XCT(1)+XNT*(XCT(2)-XCT(1))                                   COL 2680
      CFF=10.**CFF                                                     COL 2690
149   CONTINUE                                                         COL 2700
      TCF=CFT*CFF-1.                                                   COL 2710
      CFD=1.                                                           COL 2720
      IF ((D50.GE.0.20).AND.(D50.LE.0.30)) GO TO 154                   COL 2730
      IP1=0                                                            COL 2740
      IP2=0                                                            COL 2750
      DO 152 I=1,10                                                    COL 2760
         IF ((D50.GE.DP(I)).AND.(D50.LE.DP(I+1))) GO TO 150            COL 2770
         GO TO 151                                                     COL 2780
150      IP1=I                                                         COL 2790
         IP2=I+1                                                       COL 2800
         GO TO 153                                                     COL 2810
151      CONTINUE                                                      COL 2820
152   CONTINUE                                                         COL 2830
153   CONTINUE                                                         COL 2840
      P2=ALOG10(P(IP2))                                                COL 2850
      P1=ALOG10(P(IP1))                                                COL 2860
      XNT=ALOG10(D50/DP(IP1))/ALOG10(DP(IP2)/DP(IP1))                  COL 2870
      CFD=P1+XNT*(P2-P1)                                               COL 2880
      CFD=10.**CFD                                                     COL 2890
154   CONTINUE                                                         COL 2900
      FFF=CFD*TCF                                                      COL 2910
      FFF=FFF+1.                                                       COL 2920
      GT=FFF*GTUC                                                      COL 2930
      GT=GT*W                                                          COL 2940
      WRITE (6,161) GT,REMARK                                          COL 2950
```

```
      GO TO 156                                                         COL 2960
155   CONTINUE                                                          COL 2970
      WRITE (6,158) REMARK                                              COL 2980
156   CONTINUE                                                          COL 2990
157 CONTINUE                                                            COL 3000
C                                                                       COL 3010
158 FORMAT (5X, 38HCOMPUTATIONS COULD NOT BE CARRIED OUT ,/5X, 24HDUE COL 3020
   1TO DATA OUT OF RANGE,/5X,  8HREMARK= ,R10///)                       COL 3030
159 FORMAT (1H1,9X, 33HCOMPUTATION OF TOTAL BED MATERIAL,/10X, 26HTRANCOL 3040
   1SPORT BY COLBYS METHOD,//)                                          COL 3050
160 FORMAT (5X,4HSET ,I5/5X,27HAVERAGE VELOCITY            ,F12.2,12H FCOL 3060
   1T./SEC.    ,/5X,27HHYDRAULIC DEPTH          ,F12.2,12H FT.        COL 3070
   2 ,/5X,27HWATER SURFACE WIDTH       ,F12.2,12H FT.         ,/5X,27HCOL 3080
   3TEMPERATURE                 ,F12.2,12H DEG.FAHREN.,/5X,27HMEDIAN BECOL 3090
   4D MATERIAL SIZE   ,F12.2,12H MM.         ,/5X,27HFINE MATERIAL CONCCOL 3100
   5ENTRATION,F12.2,12H PPM.        ,/)                                  COL 3110
161 FORMAT (5X,24HBED MATERIAL TRANSPORT =,F15.5,12H TONS/DAY   ,/5X,9COL 3120
   1HREMARK = ,R10///)                                                  COL 3130
162 FORMAT (I5)                                                         COL 3140
163 FORMAT (6F10.0)                                                     COL 3150
C                                                                       COL 3160
    END                                                                 COL 3170
```

MEYER

LISTING

```
      PROGRAM MEYER (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C
C     DEVELOPED          COLORADO STATE UNIVERSITY ENGINEERING RESEARCH
C                        CENTER, FORT COLLINS, COLORADO 80523
C     PURPOSE            CALCULATION OF BED LOAD TRANSPORT BY MEYER-PETER
C                        AND MULLER FORMULA(1948)
C     REFERENCE          MEYER-PETER, E. AND MULLER, R., FORMULAS FOR
C                        BED LOAD TRANSPORT, INTERNATIONAL ASSOCIATION
C                        FOR HYDRAULIC RESEARCH, SECOND MEETING,
C                        STOCKHOLM, 1948.
C     CORE USAGE         CDC 6400 SCOPE 3.3 SYSTEM DEFAULT VALUE,
C                        43000 OCTAL.
C
C
C     INPUT AND OUTPUT DESCRIPTION
C
C     INPUT CONSISTS OF THE FOLLOWING
C     1) VARIABLES V,R,W,S,D90, AND ND, TO BE READ IN FORMAT(5F10.0,I10)
C        V          AVERAGE VELOCITY                        FT./SEC.
C        R          HYDRAULIC RADIUS                        FT.
C        W          WATER SURFACE WIDTH                     FT.
C        S          ENERGY GRADIENT                         FT./FT.
C        D90        DIAMETER FOR 90 PERCENT FINER           MM.
C        ND         NO. OF FRACTIONS IN BED MATERIAL
C     2) ARRAYS FB(ND),DRL(ND),DRU(ND), TO BE READ IN FORMAT(3F10.0)
C        FB(J)      FRACTION OF BED MATERIAL IN SIZE FRACTION
C        DRL(J)     LOWER LIMIT OF SIZE FRACTION, IN MM.
C        DRU(J)     UPPER LIMIT OF SIZE FRACTION, IN MM.
C
C     OUTPUT CONSISTS OF THE BED LOAD TRANSPORT IN TONS/DAY.
C
C
      DIMENSION FB(10), DRL(10), DRU(10)
      READ (5,102) V,R,W,S,D90,ND
      WRITE (6,104)
      WRITE (6,105) V,R,W,S,D90
      READ (5,103) (FB(J),DRL(J),DRU(J),J=1,ND)
      D90=D90*0.001
      V=V*0.3048
      R=R*0.3048
      DM=0.
      DO 101 J=1,ND
         DM=DM+FB(J)*(DRL(J)+DRU(J))/2.
  101 CONTINUE
      DM=DM*0.001
      XKS=V/R**0.6667/S**0.5
      XKR=26./D90**0.1667
      RAT=XKS/XKR
      GAM=1000.
      GR=9.81
      GAMP=1650.
      RO=1000./GR
      X=GAM*R*S*RAT**1.5
      Y=0.047*DM*GAMP
      Z=0.25*RO**0.333
      GSP=((X-Y)/Z)**1.5
      GS=GSP*2.65/1.65
      GS=GS*2.2/3.28
```

```
      GS=GS*43.2*W
      WRITE (6,108)
      WRITE (6,106) (J,FB(J),DRL(J),DRU(J),J=1,ND)
      WRITE (6,107) GS
      STOP
C
  102 FORMAT (5F10.0,I10)
  103 FORMAT (3F10.0)
  104 FORMAT (1H1,//10X, 34HCOMPUTATION OF BED LOAD TRANSPORT ,/10X, 40H
     1BY MEYER-PETER AND MULLER FORMULA(1948) ,//)
  105 FORMAT (5X,27HAVERAGE VELOCITY            ,F10.2,10H FT./SEC. ,/5X,
     127HHYDRAULIC RADIUS           ,F10.2,10H FT.       ,/5X,27HWATER SU
     2RFACE WIDTH         ,F10.2,10H FT.       ,/5X,27HENERGY GRADIENT
     3          ,F10.7,10H FT./FT.  ,/5X,27HDIAMETER 90 PERCENT FINER  ,F1
     40.3,10H MM.       ,//)
  106 FORMAT (1X,I10,F10.2,2F10.4)
  107 FORMAT (//,5X, 25HTOTAL BED LOAD TRANSPORT=,F15.3,  9H TONS/DAY)
  108 FORMAT (5X, 40H      J        FB(J)     DRL(J)     DRU(J)   ,/)
C
      END
```

MODEINS

LISTING

```
      PROGRAM MODEINS (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C
C     DEVELOPED         COLORADO STATE UNIVERSITY ENGINEERING RESEARCH
C                       CENTER, FORT COLLINS, COLORADO, 80523.
C     PURPOSE           COMPUTATION OF TOTAL SEDIMENT DISCHARGE BY
C                       THE MODIFIED EINSTEIN PROCEDURE.
C     REFERENCES        U.S. BUREAU OF RECLAMATION PUBLICATION
C                       STEP METHOD FOR COMPUTING TOTAL SEDIMENT LOAD
C                       BY THE MODIFIED EINSTEIN PROCEDURE, JULY 1955
C                       (REVISED) AND ADDENDUM  COMPUTATION OF Z FOR USE
C                       IN THE MODIFIED EINSTEIN PROCEDURE, JUNE 1966.
C     CORE USAGE        CDC 6400 SCOPE 3.3 SYSTEM DEFAULT VALUE,
C                       43000 OCTAL.
C     COMPILATION TIME  APPROXIMATELY 8 SEC.
C     CENTRAL PROCESSOR
C     TIME FOR ONE
C     SET OF DATA       APPROXIMATELY 1 SEC.
C
C
C     INPUT AND OUTPUT DESCRIPTION
C
C     THE FIRST CARD IN THE INPUT LOGICAL RECORD SHOULD CONTAIN
C     THE VALUE OF NDATA, IN FORMAT I5.  NDATA IS THE NUMBER OF SETS
C     OF INPUT DATA TO BE FED TO THE COMPUTER AT A TIME.  A SET OF INPUT
C     DATA CONSISTS OF A GROUP OF VARIABLES NECESSARY TO SPECIFY
C     A PROBLEM, AS DETAILED BELOW.
C
C     THE FIRST CARD IS TO BE FOLLOWED BY THE NUMBER OF SETS OF INPUT
C     DATA, EACH ONE CONSISTING OF THE FOLLOWING, IN THE ORDER SHOWN
C
C     1) GENERAL DATA, 13 VARIABLES TO BE PUNCHED IN FORMAT (8F10.0)
C        FOLLOWING IS A LIST OF THE VARIABLES, FORTRAN NAME AND UNITS.
C        WATER DISCHARGE                 DISCH          CFS.
C        AVERAGE VELOCITY                UAVE           FT./SEC.
C        HYDRAULIC DEPTH                 DEPTH          FT.
C        WATER SURFACE WIDTH             W              FT.
C        AREA                            AREA           SQ.FT.
C        TEMPERATURE                     TEMP           DEG.FARENH.
C        KINEMATIC VISCOSITY             XNU            SQ.FT./SEC.
C        65 PERCENT FINER DIAMETER
C             FOR BED-MATERIAL           D65            FT.
C        35 PERCENT FINER DIAMETER
C             FOR BED-MATERIAL           D35            FT.
C        AVERAGE CONCENTRATION           CONC           PPM.
C        SAMPLED SUSPENDED LOAD          QSM            TONS/DAY
C        PORTION OF DEPTH NOT SAMPLED    DN             FT.
C        AVERAGE DEPTH OF SAMPLING       DS             FT.
C
C     2) INTEGER SELECTORS JIN AND JOUT, TO BE PUNCHED IN FORMAT 2I1.
C        JIN SELECTS THE NUMBER AND RANGE IN THE COMPUTATIONAL
C        SIZE FRACTIONS. ND IS THE NUMBER OF SIZE FRACTIONS.
C        IF JIN=1, THE SIZE FRACTIONS IN THE USBR PUBLICATION WILL BE
C        USED. THE FIRST TWO SIZE FRACTIONS WILL BE USED AND THE THIRD
C        DELETED, RESULTING IN ND= 10
C        IF JIN=2, THE SIZE FRACTIONS IN THE USBR PUBLICATION WILL BE
C        USED. IN THIS CASE THE FIRST TWO SIZE FRACTIONS WILL BE DELETED
C        AND THE THIRD USED INSTEAD, RESULTING IN ND=9
C        IF JIN=3, THE USER HAS THE OPTION OF SPECIFYING THE NUMBER AND
```

```
C        RANGE OF COMPUTATIONAL SIZE FRACTIONS. IF THIS OPTION IS
C        CHOSEN, ND SHOULD BE READ IN THE CARD IMMEDIATELY FOLLOWING,
C        IN FORMAT I1.
C        JOUT SELECTS THE TYPE OF OUTPUT DESIRED.
C        IF JOUT=1, OUTPUT WILL CONSIST OF THE GENERAL DATA, CHECK ON
C        CONVERGENCE OF Z PRIME, AND THE FINAL RESULTS IN 20 COLUMNS,
C        AS FOLLOWS.
C
C        1) GEOMETRIC MEAN DIAMETER, IN FT.
C        2) PSI
C        3) PHI SHEAR
C        4) PERCENTAGE OF BED MATERIAL IN SIZE FRACTION
C        5) BED LOAD TRANSPORT, TONS/DAY
C        6) PERCENTAGE OF SUSPENDED LOAD IN SIZE FRACTION
C        7) SAMPLED TRANSPORT IN SIZE FRACTION
C        8) MULTIPLIERS
C        9) Z PRIME VALUES
C        10) A DOUBLE PRIME VALUES
C        11) GEOMETRIC MEAN DIAMETER, IN FT
C        12) J ONE PRIME
C        13) J TWO PRIME
C        14) J ONE DOUBLE PRIME
C        15) J TWO DOUBLE PRIME
C        16) PRODUCT OF JS
C        17) I ONE DOUBLE PRIME
C        18) I TWO DOUBLE PRIME
C        19) PRODUCT OF IS
C        20) COMPUTED LOAD, IN TONS/DAY
C        IF JOUT=2 IS SELECTED, MOST OF THE 20 COLUMNS WILL BE OMITTED
C        IN THE PRINTOUT, AND INSTEAD ONLY COLUMNS 1,4,5,6 AND 20 WILL
C        BE PRINTED. ADDITIONALLY, DRL(J) AND DRU(J), LOWER AND UPPER
C        LIMITS OF THE SIZE FRACTION RANGE, IN MM, WILL BE PRINTED TO
C        THE LEFT OF THE 5 COLUMNS PREVIOUSLY MENTIONED.
C
C     3) DATA ARRAYS.
C        IF JIN=1, THE PERCENT OF BED MATERIAL INSIZE FRACTIONS FB(10),
C        AND PERCENT OF SUSPENDED LOAD IN SIZE FRACTIONS FS(10)
C        SHOULD BE PUNCHED IN FORMAT 2F10.0
C        IF JIN=2, FB(9) AND FS(9) SHOULD BE PUNCHED IN FORMAT 2F10.0
C        IF JIN=3, THE RANGE OF COMPUTATIONAL SIZE FRACTIONS SHOULD BE
C        SPECIFIED IN ADDITION TO THE PERCENTAGES FB AND FS.
C        IF THIS OPTION IS CHOSEN, DRL(ND), DRU(ND), FB(ND) AND FS(ND)
C        SHOULD BE PUNCHED IN FORMAT 4F10.0
C        DRL(J) AND DRU(J) ARE THE LOWER AND UPPER LIMITS OF THE SIZE
C        FRACTION RANGE, IN MM, RESPECTIVELY. NOTE THAT SIZE FRACTIONS
C        SHOULD BE PUNCHED IN ORDER OF INCREASING SIZE.
C
C
      COMMON /ALL/ DISCH,UAVE,DEPTH,W,AREA,TEMP,XNU,D65,D35,CONC,QSM,DN,
     1DS
      COMMON /ALLB/ D(11),VS(11),FB(10),FS(10),XMULT(10),JIN,JOUT,ND,ND1
     1,ND2
      COMMON /ALLC/ QSP(10),XIBQB(10),FQL(10)
      COMMON /ALLD/ P,AP,APP(10),ZP(10)
      COMMON /ALLE/ DRL(11),DRU(11)
      COMMON /CEF/ CJ0(2),CJ1(2),CJ2(2),CJ3(2),CJ(2),C1(2),C2(2),C3(2),C
     14(2),M
      DIMENSION COL16(10), COL17(10), COL18(10), COL19(10), COL20(10), C
     1OL21(10), COL22(10), COL23(10), PSI(10), PHISH(10)
```

```
      READ (5,120) NDATA
      DO 119 L=1,NDATA
         WRITE (6,125)
         WRITE (6,126)
         CALL INPUT1
         CALL INPUT2
         WRITE (6,127)
         WRITE (6,128) L,DISCH,UAVE,DEPTH,W,AREA,TEMP,XNU,D65,D35,CONC,Q
     1   SM,DN,DS
C
C     CALCULATING HYDRAULIC RADIUS*SLOPE RS, PERCENTAGE OF FLOW SAMPLED
C     PFS, AND SEDIMENT DISCHARGE THROUGH THE SAMPLED ZONE QSPT
C
         CALL RSCOM (X,RS)
         CALL PLATE4 (X,PFS,XKS)
         QSPT=QSM*PFS
C
C     CALCULATING PSI(J)
C
         DO 102 J=1,ND
            XPSI=1.65*D35/RS
            YPSI=0.66*D(J)/RS
            XYPSI=XPSI-YPSI
            IF (XYPSI.LT.0) GO TO 101
            PSI(J)=XPSI
            GO TO 102
  101       PSI(J)=YPSI
  102    CONTINUE
C
C     CALCULATING BED LOAD DISCHARGE XIBQB(J) AND PERCENTAGE OF
C     SUSPENDED MATERIAL IN VARIOUS SIZE FRACTIONS QSP(J)
C
         DO 103 J=1,ND
            XX=PSI(J)
            CALL PLATE5 (XX,YY)
            PHISH(J)=YY
            XIBQB(J)=43.2*W*1200.*PHISH(J)/2.*D(J)**1.5*FB(J)
            QSP(J)=FS(J)*QSPT
  103    CONTINUE
C
C     CALCULATING P, APRIME AP, AND A DOUBLE PRIME APP(J)
C
         DXKS=30.2*X*DEPTH/XKS
         P=2.303*ALOG10(DXKS)
         AP=DN/DS
         DO 104 J=1,ND
            APP(J)=2*D(J)/DEPTH
  104    CONTINUE
         CALL SDR (N,K)
         N1=N+1
         NK=N+K
C
C     IF K IS GREATER THAN 2, CONTROL BRANCHES TO STATEMENT 30
C     CALCULATING MULTIPLIERS XMULT(J) , AND ZPRIME ZP(J)
C
         IF (K.GT.2) GO TO 106
         CALL MULCOM (K,N1,NK,KK)
         M=0
         CALL ZPCOM (KK)
```

```
      DO 105 J=1,ND
         ZP(J)=ZP(KK)*XMULT(J)
  105 CONTINUE
      GO TO 110
C
C     CALCULATING ZP AND VS ARRAYS TO BE FED TO LEAST SQUARE SUBROUTINE
C     LSZPVS
C
  106 CONTINUE
      M=0
      DO 107 J=N1,NK
         CALL ZPCOM (J)
  107 CONTINUE
      IF (JOUT.EQ.2) GO TO 108
      WRITE (6,121)
      WRITE (6,123)
      WRITE (6,124) (J,ZP(J),VS(J),J=N1,NK)
  108 CONTINUE
      CALL LSZPVS (N1,NK,K,VS,ZP,A,B)
      A=EXP(A)
      DO 109 J=1,ND
         XMULT(J)=0.0
         ZP(J)=A*VS(J)**B
  109 CONTINUE
C
C     CALCULATING SEDIMENT LOAD BY USING MODIFIED EINSTEINS INTEGRAL
C     CHARTS 9,10,11 AND 12
C
  110 CONTINUE
      IF (JOUT.EQ.2) GO TO 112
      IF (K.LT.3) GO TO 111
      WRITE (6,122)
  111 CONTINUE
      IF (JOUT.EQ.2) GO TO 112
      WRITE (6,123)
      WRITE (6,124) (J,ZP(J),VS(J),J=1,ND)
  112 CONTINUE
      TQL=0
      TBL=0
      DO 116 I=1,ND
         XM=APP(I)
         ZM=ZP(I)
         IF (FB(I).LT.0.01.AND.FS(I).LT.0.01) GO TO 114
         IF (FB(I).LT.0.01) GO TO 113
         CALL POLYNML (XM,ZM,COL21(I),COL22(I),DUM1,DUM2)
         COL23(I)=P*COL21(I)+COL22(I)+1.
         FQL(I)=XIBQB(I)*COL23(I)
         COL16(I)=0.
         COL17(I)=0.
         COL18(I)=0.
         COL19(I)=0.
         COL20(I)=0.
         GO TO 115
  113    CONTINUE
         CALL POLYNML (AP,ZM,DUM1,DUM2,COL16(I),COL17(I))
         CALL POLYNML (XM,ZM,DUM3,DUM4,COL18(I),COL19(I))
         COL20(I)=(P*COL18(I)+COL19(I))/(P*COL16(I)+COL17(I))
         FQL(I)=QSP(I)*COL20(I)
         COL21(I)=0.
```

```
            COL22(I)=0.
            COL23(I)=0.
            GO TO 115
  114       CONTINUE
            FQL(I)=0.0
            COL16(I)=0.
            COL17(I)=0.
            COL18(I)=0.
            COL19(I)=0.
            COL20(I)=0.
            COL21(I)=0.
            COL22(I)=0.
            COL23(I)=0.
  115       CONTINUE
            TQL=TQL+FQL(I)
            TBL=TBL+XIBQB(I)
  116     CONTINUE
          TSL=TQL-TBL
C
C       PRINTING OUTPUT
C
          WRITE (6,129)
          IF (JOUT.EQ.2) GO TO 117
          WRITE (6,130)
          WRITE (6,131) (J,D(J),PSI(J),PHISH(J),FB(J),XIBQB(J),FS(J),QSP(
     1    J),XMULT(J),ZP(J),APP(J),J=1,ND)
          WRITE (6,132)
          WRITE (6,133) (J,D(J),COL16(J),COL17(J),COL18(J),COL19(J),COL20
     1    (J),COL21(J),COL22(J),COL23(J),FQL(J),J=1,ND)
          GO TO 118
  117     CONTINUE
          WRITE (6,134)
          WRITE (6,135) (DRL(J),DRU(J),D(J),FB(J),XIBQB(J),FS(J),FQL(J),J
     1    =1,ND)
  118     CONTINUE
          WRITE (6,136) TBL,TSL,TQL
  119 CONTINUE
C
C       FORMAT STATEMENTS
C
      STOP
C
  120 FORMAT (I5)
  121 FORMAT (//,10X, 41H ARRAYS ZP AND VS BEFORE LEAST SQUARE FIT,/)
  122 FORMAT (//,10X, 40H ARRAYS ZP AND VS AFTER LEAST SQUARE FIT,/)
  123 FORMAT (//,10X, 35H           J        ZP(J)        VS(J),/)
  124 FORMAT (10X,I12,2F12.6)
  125 FORMAT (1H1)
  126 FORMAT (40X, 70HCOMPUTATION OF TOTAL SEDIMENT LOAD BY THE MODIFIED
     1  EINSTEIN PROCEDURE,///)
  127 FORMAT (32X, 10HDATA INPUT,//)
  128 FORMAT (10X,4HSET ,I5,/10X, 34HWATER DISCHARGE                   ,
     1F12.2, 13H C.F.S.       ,/10X, 34HAVERAGE VELOCITY
     2 ,F12.2, 13H FT./SEC.    ,/10X, 34HHYDRAULIC DEPTH
     3   ,F12.2, 13H FT.          ,/10X, 34HWATER SURFACE WIDTH
     4     ,F12.2, 13H FT.          ,/10X, 34HAREA
     5       ,F12.2, 13H SQ.FT.       ,/10X, 34HTEMPERATURE
     6         ,F12.2, 13H DEG.FAHREN. ,/10X, 34HKINEMATIC VISCOSITY
     7           ,F12.7, 13H SQ.FT./SEC. ,/10X, 34HD65
```

```
     8              ,F12.6, 13H FT.          ,/10X, 34HD35                   MOD 2960
     9                ,F12.6, 13H FT.          ,/10X, 34HAVERAGE CONCENTRAMOD 2970
     *TION              ,F12.2, 13H PPM.         ,/10X, 34HSAMPLED SUSPENDMOD 2980
     *ED LOAD             ,F12.4, 13H TONS/DAY     ,/10X, 34HPORTION OF DEMOD 2990
     *PTH NOT SAMPLED       ,F12.2, 13H FT.          ,/10X, 34HAVERAGE DEPMOD 3000
     *TH AT SAMPLING          ,F12.2,  4H FT.)                               MOD 3010
  129 FORMAT (//)                                                            MOD 3020
  130 FORMAT (5X,1HJ,11X,4HD(J),7X,6HPSI(J),5X,8HPHISH(J),7X,5HFB(J),4X,MOD 3030
     18HXIBQB(J),7X,5HFS(J),6X,6HQSP(J),4X,8HXMULT(J),7X,5HZP(J),5X,6HAPMOD 3040
     2P(J)/)                                                                 MOD 3050
  131 FORMAT (4X,I2,4X,F12.6,F12.3,F12.5,6F12.3,F12.6)                       MOD 3060
  132 FORMAT (//5X,1HJ,11X,4HD(J),6X,8HCOL16(J),4X,8HCOL17(J),4X,8HCOL18MOD 3070
     1(J),4X,8HCOL19(J),4X,8HCOL20(J),4X,8HCOL21(J),4X,8HCOL22(J),4X,8HCMOD 3080
     2OL23(J),4X,9HCOMP.LOAD/)                                               MOD 3090
  133 FORMAT (4X,I2,4X,F12.6,8F12.3,F12.4)                                   MOD 3100
  134 FORMAT (10X, 84H      DRL(J)       DRU(J)        D(J)         FB(J)   MOD 3110
     1  XIBQB(J)        FS(J)        FQL(J),/)                               MOD 3120
  135 FORMAT (10X,6F12.6,F12.3)                                              MOD 3130
  136 FORMAT (///,5X, 34HTOTAL BED LOAD                        ,F16.4,  9H TMOD 3140
     1ONS/DAY,/5X, 34HTOTAL SUSPENDED BED MATERIAL LOAD ,F16.4,  9H TONSMOD 3150
     2/DAY,/5X, 34HTOTAL BED MATERIAL LOAD            ,F16.4,  9H TONS/DAMOD 3160
     3Y)                                                                     MOD 3170
C                                                                            MOD 3180
      END                                                                    MOD 3190
```

```
      SUBROUTINE INPUT1                                                 MOD 3200
C                                                                       MOD 3210
C     THIS SUBROUTINE READS IN THE BASIC VARIABLES OF THE PROBLEM       MOD 3220
C                                                                       MOD 3230
      COMMON /ALL/ DISCH,UAVE,DEPTH,W,AREA,TEMP,XNU,D65,D35,CONC,QSM,DN,MOD 3240
     1DS                                                                MOD 3250
      READ (5,101) DISCH,UAVE,DEPTH,W,AREA,TEMP,XNU,D65,D35,CONC,QSM,DN,MOD 3260
     1DS                                                                MOD 3270
      RETURN                                                            MOD 3280
C                                                                       MOD 3290
  101 FORMAT (8F10.0)                                                   MOD 3300
C                                                                       MOD 3310
      END                                                               MOD 3320
```

```
      SUBROUTINE INPUT2
C
C     THIS SUBROUTINE READS IN ADDITIONAL INPUT AND FINDS THE VALUE OF
C     ND, THE NUMBER OF SIZE FRACTIONS TO BE USED IN THE COMPUTATION
C
      COMMON /ALL/ DISCH,UAVE,DEPTH,W,AREA,TEMP,XNU,D65,D35,CONC,QSM,DN,
     1DS
      COMMON /ALLB/ D(11),VS(11),FB(10),FS(10),XMULT(10),JIN,JOUT,ND,ND1
     1,ND2
      COMMON /ALLE/ DRL(11),DRU(11)
      DATA (DRL(J),J=1,11)/0.002,0.0156,0.002,0.0625,0.125,0.250,0.500,1
     1.000,2.000,4.000,8.000/
      DATA (DRU(J),J=1,11)/0.0156,0.0625,0.0625,0.125,0.250,0.500,1.000,
     12.000,4.000,8.000,16.000/
      READ (5,109) JIN,JOUT
      IF (JIN.EQ.3) GO TO 106
      DO 101 J=1,11
         D(J)=(DRL(J)*DRU(J))**0.5/304.8
         VS(J)=((2./3.*32.17*1.65*D(J)**3.+36.*XNU**2.)**0.5-6.*XNU)/D(J
     1    )
  101 CONTINUE
      ND1=10
      ND2=9
      IF (JIN.EQ.2) GO TO 103
      DO 102 J=3,ND1
         D(J)=D(J+1)
         VS(J)=VS(J+1)
  102 CONTINUE
      ND=ND1
      GO TO 105
  103 DO 104 J=1,ND2
         D(J)=D(J+2)
         VS(J)=VS(J+2)
  104 CONTINUE
      ND=ND2
  105 CONTINUE
      READ (5,110) (FB(J),FS(J),J=1,ND)
      GO TO 108
  106 CONTINUE
      READ (5,111) ND
      READ (5,112) (DRL(J),DRU(J),FB(J),FS(J),J=1,ND)
      DO 107 J=1,ND
         D(J)=(DRU(J)*DRL(J))**0.5/304.8
         VS(J)=((2./3.*32.17*1.65*D(J)**3.+36.*XNU**2.)**0.5-6.*XNU)/D(J
     1    )
  107 CONTINUE
  108 CONTINUE
      RETURN
C
  109 FORMAT (2I1)
  110 FORMAT (2F10.0)
  111 FORMAT (I1)
  112 FORMAT (4F10.0)
C
      END
```

```
      SUBROUTINE RSCOM (X,RS)
C
C     THIS SUBROUTINE COMPUTES THE VALUE OF RS BY ITERATION
C
      COMMON /ALL/ DISCH,UAVE,DEPTH,W,AREA,TEMP,XNU,D65,D35,CONC,QSM,DN,
     1DS
      X=1.6
      TOL=0.001
      XKS=D65
  101 XDKS=12.27*X*DEPTH/XKS
      SRRS=UAVE/(32.63*ALOG10(XDKS))
      USHP=SRRS*5.68
      DEL=11.6*XNU/USHP
      DELKS=XKS/DEL
      CALL PLATE3 (DELKS,X2)
      DELX=X-X2
      IF (ABS(DELX).LT.TOL) GO TO 102
      X=X2
      GO TO 101
  102 CONTINUE
      XDKS=12.27*X*DEPTH/XKS
      SRRS=UAVE/(32.63*ALOG10(XDKS))
      RS=SRRS*SRRS
      RETURN
C
      END
```

```
      SUBROUTINE PLATE4 (X,PFS,XKS)
C
C     THIS SUBROUTINE SUBSTITUTES PLATE FOUR FOR THE ANALYTICAL
C     EXPRESSION OF PFS
C
      COMMON /ALL/ DISCH,UAVE,DEPTH,W,AREA,TEMP,XNU,D65,D35,CONC,QSM,DN,
     1DS
      XKS=D65
      A=30.2*X/XKS
      YDS=DS*ALOG(A*DS)-DS
      YDN=DN*ALOG(A*DN)-DN
      PFS=(YDS-YDN)/YDS
      RETURN
C
      END
```

```
      SUBROUTINE SDR (N,K)
C
C     THIS SUBROUTINE COUNTS THE NUMBER OF SIZE FRACTIONS K FOR WHICH
C     THERE IS BOTH BED AND SUSPENDED DISCHARGE, AND THE NUMBER OF SIZE
C     FRACTIONS N SMALLER THAN FIRST K.
C
      COMMON /ALLB/ D(11),VS(11),FB(10),FS(10),XMULT(10),JIN,JOUT,ND,ND1
     1,ND2
      J=0
      K=0
      N=0
  101 CONTINUE
      IF (FB(J+1).GT.0.00.AND.FS(J+1).GT.0.00) GO TO 103
      IF (K.NE.0) GO TO 102
      N=N+1
  102 J=J+1
      IF (J.EQ.ND) RETURN
      GO TO 101
  103 CONTINUE
      K=K+1
      J=J+1
      IF (J.EQ.ND) RETURN
      GO TO 101
C
      END
```

```
      SUBROUTINE LSZPVS (N1,NK,K,X,Y,A,B)
C
C     THIS SUBROUTINE CALCULATES A LEAST SQUARE FIT FOR ZPRIME ZP(K) AND
C     VS(K)
C
      DIMENSION X(11), Y(10)
      SUMX=0.
      SUMY=0.
      SUMXY=0.
      SUMX2=0.
      DO 101 J=N1,NK
         XL=ALOG(X(J))
         SUMX=SUMX+XL
         YL=ALOG(Y(J))
         SUMY=SUMY+YL
         XY=XL*YL
         SUMXY=SUMXY+XY
         X2=XL*XL
         SUMX2=SUMX2+X2
  101 CONTINUE
      XMEAN=SUMX/K
      YMEAN=SUMY/K
      B=(SUMXY-SUMX*SUMY/K)/(SUMX2-SUMX*SUMX/K)
      A=YMEAN-B*XMEAN
      RETURN
C
      END
```

```
      SUBROUTINE MULCOM (K,N1,NK,KK)                                    MOD 4810
C                                                                       MOD 4820
C     THIS SUBROUTINE CALCULATES THE MULTIPLIERS XMULT(J)               MOD 4830
C                                                                       MOD 4840
      COMMON /ALLB/ D(11),VS(11),FB(10),FS(10),XMULT(10),JIN,JOUT,ND,ND1MOD 4850
     1,ND2                                                              MOD 4860
      DIMENSION SBS(9)                                                  MOD 4870
      IF (K.EQ.0) GO TO 106                                             MOD 4880
      IF (K.EQ.2) GO TO 101                                             MOD 4890
      KK=N1                                                             MOD 4900
      GO TO 104                                                         MOD 4910
  101 CONTINUE                                                          MOD 4920
      DO 102 J=N1,NK                                                    MOD 4930
         SBS(J)=FB(J)+FS(J)                                             MOD 4940
  102 CONTINUE                                                          MOD 4950
      IF (SBS(N1).GT.SBS(NK)) GO TO 103                                 MOD 4960
      KK=NK                                                             MOD 4970
      GO TO 104                                                         MOD 4980
  103 KK=N1                                                             MOD 4990
  104 CONTINUE                                                          MOD 5000
      DO 105 J=1,ND                                                     MOD 5010
         XMULT(J)=(VS(J)/VS(KK))**0.7                                   MOD 5020
  105 CONTINUE                                                          MOD 5030
      GO TO 107                                                         MOD 5040
  106 WRITE (6,108)                                                     MOD 5050
  107 CONTINUE                                                          MOD 5060
      RETURN                                                            MOD 5070
C                                                                       MOD 5080
  108 FORMAT (10X, 97HBECAUSE NO SIZE FRACTION CONTAINS BOTH BED AND SUSMOD 5090
     1PENDED DISCHARGE, RESULTS COULD NOT BE OBTAINED)                  MOD 5100
C                                                                       MOD 5110
      END                                                               MOD 5120
```

```
      SUBROUTINE PLATE8 (X,Y)
C
C     THIS SUBROUTINE APPROXIMATES PLATE 8 BY A LINE IN LOG-LOG PAPER
C
      Y=-0.33*ALOG10(X)+1.08
      RETURN
C
      END
```

```
      SUBROUTINE ZPCOM (J)
C
C     THIS SUBROUTINE COMPUTES ZPRIME ZP BY ITERATION
C     FIRST, ATRIAL VALUE OF ZP IS CALCULATED, AND THEN, WITH ANOTHER
C     TRIAL, A LINEAR INTERPOLATION IS MADE. CONVERGENCE IS VERY FAST.
C
      COMMON /ALLB/ D(11),VS(11),FB(10),FS(10),XMULT(10),JIN,JOUT,ND,ND1
     1,ND2
      COMMON /ALLC/ QSP(10),XIBQB(10),FQL(10)
      COMMON /ALLD/ P,AP,APP(10),ZP(10)
      XM=APP(J)
      RQSP=QSP(J)/XIBQB(J)
      CALL PLATE8 (RQSP,ZTRY)
      STEP=0.01
      KOUNT=0
      IF (JOUT.EQ.2) GO TO 101
      WRITE (6,106)
  101 CONTINUE
  102 CONTINUE
      KOUNT=KOUNT+1
      IF (KOUNT.GT.10) GO TO 104
      CALL POLYNML (XM,ZTRY,XI1PP,DUM1,XJ1PP,DUM2)
      CALL POLYNML (AP,ZTRY,DUM3,DUM4,XJ1P,XJ2P)
      CRQSP=XI1PP/XJ1PP*(P*XJ1P+XJ2P)
      DCRQ=CRQSP-RQSP
      IF (JOUT.EQ.2) GO TO 103
      WRITE (6,108) KOUNT,ZTRY,RQSP,CRQSP,DCRQ
  103 CONTINUE
      TOL=0.01*RQSP
      IF (ABS(DCRQ).LT.TOL) GO TO 105
      IF (CRQSP.LT.RQSP) ZTRY1=ZTRY-STEP
      IF (CRQSP.GT.RQSP) ZTRY1=ZTRY+STEP
      CALL POLYNML (XM,ZTRY1,XI1PP,DUM1,XJ1PP,DUM2)
      CALL POLYNML (AP,ZTRY1,DUM3,DUM4,XJ1P,XJ2P)
      CRQSP1=XI1PP/XJ1PP*(P*XJ1P+XJ2P)
      TEMP=(RQSP-CRQSP)*STEP/(CRQSP1-CRQSP)
      IF (CRQSP.LT.RQSP) ZTRY=ZTRY-TEMP
      IF (CRQSP.GT.RQSP) ZTRY=ZTRY+TEMP
      GO TO 102
  104 CONTINUE
      WRITE (6,107)
  105 CONTINUE
      ZP(J)=ZTRY
      RETURN
C
  106 FORMAT (//,10X, 75H CONVERGENCE OF SUBROUTINE ZPCOM IS CHECKED BY
     1PRINTING OUT VALUES INVOLVED,//,19X,5HITER.,5X,5HZTRY ,6X,4HRQSP,8
     2X,5HCRQSP,7X,4HDCRQ)
  107 FORMAT (/10X, 76HZPCOM DOES NOT CONVERGE WITH 10 ITERATIONS, LAST
     1VALUE OF ZP(J) WILL BE USED,/)
  108 FORMAT (/10X,I12,4F12.5)
C
      END
```

```
      SUBROUTINE PLATE3 (X,Y)
C
C     THIS SUBROUTINE APPROXIMATES PLATE 3 BY A SERIES OF EQUATIONS
C
      IF (X.LE.0.40) GO TO 101
      GO TO 102
  101 Y=1.769*ALOG10(X/0.080)
      GO TO 117
  102 IF (X.GT.0.40.AND.X.LE.0.56) GO TO 103
      GO TO 104
  103 Y=1.495*ALOG10(X/0.059)
      GO TO 117
  104 IF (X.GT.0.56.AND.X.LE.0.76) GO TO 105
      GO TO 106
  105 Y=0.92*ALOG10(X/0.0145)
      GO TO 117
  106 IF (X.GT.0.76.AND.X.LE.0.96) GO TO 107
      GO TO 108
  107 Y=0.292*ALOG10(X/2.9E-06)
      GO TO 117
  108 IF (X.GT.0.96.AND.X.LE.1.35) GO TO 109
      GO TO 110
  109 Y=0.277*ALOG10(632000.0/X)
      GO TO 117
  110 IF (X.GT.1.35.AND.X.LE.3.00) GO TO 111
      GO TO 112
  111 Y=1.115*ALOG10(34.4/X)
      GO TO 117
  112 IF (X.GT.3.00.AND.X.LE.4.00) GO TO 113
      GO TO 114
  113 Y=0.725*ALOG10(128.0/X)
      GO TO 117
  114 IF (X.GT.4.00.AND.X.LE.6.70) GO TO 115
      GO TO 116
  115 Y=0.399*ALOG10(2160.0/X)
      GO TO 117
  116 IF (X.GT.6.70) Y=1.0
  117 RETURN
C
      END
```

```
      SUBROUTINE PLATE5 (X,Y)
C
C     THIS SUBROUTINE APPROXIMATES PLATE 5 BY A SERIES OF EQUATIONS
C
C
      IF (X.LE.0.77) Y=(7.56/X)**1.01
      IF (X.GT.0.77.AND.X.LE.2.12) Y=(5.35/X)**1.19
      IF (X.GT.2.12.AND.X.LE.4.10) Y=(4.10/X)**1.67
      IF (X.GT.4.10.AND.X.LE.6.10) Y=(4.10/X)**2.30
      IF (X.GT.6.10.AND.X.LE.11.0) Y=(4.60/X)**3.23
      IF (X.GT.11.0.AND.X.LE.16.7) Y=(5.66/X)**4.26
      IF (X.GT.16.7.AND.X.LE.22.5) Y=(9.28/X)**7.81
      IF (X.GT.22.5) Y=(13.10/X)**12.66
      RETURN
C
      END
```

```
      SUBROUTINE POLYNML (A,Z,XI1,XI2,XJ1,XJ2)
C
C     THIS SUBROUTINE COMPUTES THE VALUE OF THE INTEGRALS FROM MODIFIED
C     EINSTEINS PLATES NINE TO TWELVE. THE VALUES IN THE FORMAL PARA-
C     METERS ARE ARRANGED IN THE FOLLOWING ORDER  INPUT A, INPUT Z,
C     OUTPUT PLATE NINE, OUTPUT PLATE TWELVE, OUTPUT PLATE TEN, OUTPUT
C     PLATE ELEVEN
C
      COMMON /CEF/ CJ0(2),CJ1(2),CJ2(2),CJ3(2),CJ(2),C1(2),C2(2),C3(2),C
     14(2),M
      IS=0
      X1=0.
      X2=0.
      IF (A.GE.0.0050.OR.Z.GE.0.8) GO TO 101
      A1=A
      A=0.0050
      IS=1
  101 CONTINUE
      IF (M.EQ.0) CALL COEF (A)
      DO 102 I=1,2
  102 CJ(I)=10.**(C1(I)+C2(I)*Z+C3(I)*Z*Z+C4(I)*Z**3)
      IF (IS.NE.1) GO TO 103
      DAC=0.005
      CALL SIMPSON (A1,DAC,Z,X1,X2)
      A=A1
  103 CONTINUE
      FACT=0.216*A**(Z-1.)/(1.-A)**Z
      XJ1=X1+CJ(1)
      XJ2=X2-CJ(2)
      XI1=FACT*XJ1
      XI2=FACT*XJ2
      RETURN
C
      END
```

```
      SUBROUTINE COEF (A)                                               MOD 6640
      COMMON /CEF/ CJ0(2),CJ1(2),CJ2(2),CJ3(2),CJ(2),C1(2),C2(2),C3(2),CMOD 6650
     14(2),M                                                            MOD 6660
      AP=ALOG(A)                                                        MOD 6670
      CL1=1.-A                                                          MOD 6680
      CL2=A-1.-A*AP                                                     MOD 6690
      CJ0(1)=ALOG10(CL1)                                                MOD 6700
      CJ1(1)=ALOG10(-CL1-AP)                                            MOD 6710
      CJ2(1)=ALOG10(1./A+2.*AP-A)                                       MOD 6720
      CJ3(1)=ALOG10(1.5+A-3./A+0.5/(A*A)-3.*AP)                         MOD 6730
      CJ0(2)=ALOG10(-CL2)                                               MOD 6740
      CJ1(2)=ALOG10(-(-CL2-AP**2/2.))                                   MOD 6750
      CJ2(2)=ALOG10(-(-2.+(1.+AP)/A+A*(1.-AP)+AP**2))                   MOD 6760
      CJ3(2)=ALOG10(-(3.75+(AP+0.5)/(2.*A*A)-3./A*(1.+AP)+A*(AP-1.)-1.5*MOD 6770
     1AP**2))                                                           MOD 6780
      DO 101 I=1,2                                                      MOD 6790
         C1(I)=CJ0(I)                                                   MOD 6800
         C2(I)=-1.83333*CJ0(I)+3.*CJ1(I)-1.5*CJ2(I)+0.3333*CJ3(I)       MOD 6810
         C3(I)=(2.*CJ0(I)-5.*CJ1(I)+4.*CJ2(I)-CJ3(I))/2.                MOD 6820
         C4(I)=(-CJ0(I)+3.*CJ1(I)-3.*CJ2(I)+CJ3(I))/6.                  MOD 6830
  101 CONTINUE                                                          MOD 6840
      RETURN                                                            MOD 6850
C                                                                       MOD 6860
      END                                                               MOD 6870
```

```
      SUBROUTINE SIMPSON (XM,XC,Z,XJ1,XJ2)
      DIMENSION YI1(51), YI2(51)
      SUMI=0.
      SUMJ=0.
      XB=50.0
      XI1=0.
      XI2=0.
      XM1=0.1
      INDI=0
      IF (XM1.GT.XC) XM1=XC/10.
  101 IF (XM1.LE.XM) INDI=1
      IF (INDI.EQ.1) XM1=XM
      DX1=(XM1-XC)/XB
      NXB=XB+1.1
      DO 102 I=1,NXB
         XI=I
         X=XC+(XI-1.)*DX1
         YI1(I)=((1.-X)/X)**Z
         YI2(I)=YI1(I)*ALOG(X)
  102 CONTINUE
      NXB1=NXB-2
      DO 103 I=1,NXB1,2
         SUMI=SUMI+(YI1(I)+4.*YI1(I+1)+YI1(I+2))
         SUMJ=SUMJ+(YI2(I)+4.*YI2(I+1)+YI2(I+2))
  103 CONTINUE
      XI1=XI1+SUMI*DX1/3.
      XI2=XI2+SUMJ*DX1/3.
      IF (INDI.EQ.1) GO TO 104
      XC=XM1
      XM1=XM1/10.
      SUMI=0.0
      SUMJ=0.0
      GO TO 101
  104 CONTINUE
      XJ1=-XI1
      XJ2=-XI2
      RETURN
C
      END
```