Image Cover Sheet

CLASSIFICATION	SYSTEM NUMBER 142053
UNCLASSIFIED	

TITLE

EXPERIMENTATION AND CREATION OF A COMPUTER MODEL FOR TRAFFICABILITY OF SNOW.
TMTRACK 93: COMPUTER PROGRAM FOR ANALYSIS OF TRACK-SNOW INTERACTION

AN: 94-05568

System Number:

Patron Number:

Requester:

Notes:

DSIS Use only:

Deliver to: BA

DRES

DEFENCE RESEARCH ESTABLISHMENT SUFFIELD

CR 94-008

UNCLASSIFIED

EXPERIMENTATION AND CREATION OF A COMPUTER MODEL FOR TRAFFICABILITY OF SNOW
COMPUTER PROGRAM FOR ANALYSIS OF TRACK-SNOW INTERACTION (U)

BY

PROF. R.N. YONG,
DR. A.M.O. MOHAMED,
DR. D.M. XU, AND
DR. H. ANTIA

MCGILL UNIVERSITY

DRES SCIENTIFIC AUTHORITY: DR. G. IRWIN

MARCH 1994

WARNING
"The use of this information is permitted subject to recognition of proprietary and patent rights."

CRAD

National Defence Défence nationale Canada

McGill

Geotechnical Research Centre

McGill University
Macdonald Engineering Building
817 Sherbrooke Street West
Montreal, Quebec, Canada H3A 2K6

Centre de recherche géotechnique

Université McGill
Pavillon Macdonald de génie
817, rue Sherbrooke ouest
Montréal (Québec) Canada H3A 2K6

Tel.:
Telex: Canada and USA: 05-268510
Other countries: 5268510
Fax: (514) 398-7361

EXPERIMENTATION AND CREATION OF A COMPUTER MODEL FOR TRAFFICABILITY OF SNOW

TMTRACK 93

"Computer Program for Analysis of Track-Snow Interaction"

Submitted to

DEFENCE RESEARCH ESTABLISHMENT SUFFIELD

SCC File No. XSG90-00251-(603)
Contract No. W7702-0-R229/01-XSG

prepared by

GEOTECHNICAL RESEARCH CENTRE
McGILL UNIVERSITY

Scientific Adviser	**Prof. R.N. Yong**
Researchers	**Dr. A.M.O. Mohamed** **Dr. D.M. Xu** **Dr. H. Antia**

November 1993

McGill

Geotechnical Research Centre

McGill University
Macdonald Engineering Building
817 Sherbrooke Street West
Montreal, QC H3A 2K6

Centre de recherche géotechnique

Université McGill
Pavillon Macdonald de génie
817, rue Sherbrooke ouest
Montréal, QC H3A 2K6

Tel: (514) 398-6672
Telex: Canada and USA: 05-268510
Other countries: 5268510
Fax: (514) 398-7361

COMPUTER PROGRAM FOR ANALYSIS OF TRACK-SNOW INTERACTION VERSION 2

The following modifications and enhancements have been made to version 1 (Nov. 1993):

- A flag has been introduced to handle the problem of excessive snow sinkage. The program is terminated if the snow sinkage exceeds the pressure bulb height. The program may to be re-run with a higher snow density and/or smaller vehicle load.
- An option has been provided to enable the user import the raw penetrometer data from a file (e.g. PENET.DAT). The user may still enter the data interactively. The updated version (2) also prompts the user for a file name to which the raw data is to be saved.
- The curve fitting routine (PENET.FOR) now presents the fitness in relative, rather than absolute, terms. The correlation coefficient – a better indicator of fitness than the absolute error – is adopted.
- Unit of snow density is now specified in Mg/m^3 rather than kg/m^3, in conformity with that in the tyre-snow interaction program. Thus TMSNOW.DAT (used in track-snow program) is now identical with SNOW.DAT (used in tyre-snow program).
- The output is now saved in TMTRACK.OUT rather than TMRESULT.OUT.

"TMTRACK 93"

Computer Program for Analysis of Track-Snow Interaction

Fortran Edition

November 1993

Geotechnical Research Center

McGill University

Montreal

NOTE

"TMTRACK93" is a FORTRAN 77 computer program for the analysis of track-snow interaction. It was compiled with Microsoft Optimizing Compiler version 5.10, and linked with Microsoft Segmented-Executable linker, version 5.15.
The accompanying diskette contains the following files:

- PENET.DAT (penetrometer data)
- PENET.OUT (penetrometer output file)
- TMSNOW.DAT (snow data file)
- TMSIZE.DAT (track data file)
- TMLOAD.DAT (load data file)
- TMRESULT.OUT (output file)
- TMTRACK.EXE (executable file)
- TMTRACK.FOR (source file)

It is strongly recommended that a back-up copy of these files be made prior to running the program. To run the program, type TMTRACK at the DOS prompt.

PENET.DAT

PENET.DAT

DFILE
LINE
N
D
P
V
PENETROMETER(1), SINKAGE(1), MASS(1)
⋮
PENETROMETER(N), SINKAGE(N), MASS(N)
RESP

Notation

DFILE	=	filename to save the data set in
LINE	=	one-line description of the data set
N	=	number of measurements taken
D	=	penetrometer plate diameter in centimeters, *cm*
P	=	penetrometer sensitivity in Newtons per division, *N/div*
V	=	volume of snow weighed (cm^3) - to calculate the density
PENETROMETER	=	penetrometer reading in divisions (*div*)
SINKAGE	=	sinkage in millimetres (*mm*)
MASS	=	mass of the snow block in grams (*gm*)
RESP	=	prompt for an additional data set (Y/N)

This penetrometer analysis file is for the prime user, the person who takes measurements in the snow. It prompts for information in terms of variables one usually knows, formats and saves the data written on the back of a napkin, and evaluates the engineering parameters K (i.e. KK), C (i.e. CC) and PHI using the curve fitting (optimization) program PENET.FOR - A FORTRAN version of McGill's BASIC CURFIT.BAS.

The program echoes all its data (noted above) and the final result of the calculations to both the screen and the output file. It conducts up to 9999 iterations of a least-squared-error search for K from density data, then searches for the optimum values of C and PHI using the optimized K value and the sinkage-pressure data set. On exiting PENET.FOR, the user is prompted for the initial snow density RHO (kg/m^3) of the terrain over which the track will travel. Finally, the user is provided with the option to overwrite the snow data file TMSNOW.DAT with the just-computed snow data.

Note that the file PENET.DAT is not read by the program. It contains the data used in the sample calculation contained herein and has been included for completeness.

PENET.DAT

```
Schefferville trial, December 15, 1991.
6
15
1.3344
1000
30 70 160
80 100 227
130 300 97
180 440 307
270 450 307
230 480 275
n
```

PENET.OUT

PENET.OUT

```
Schefferville trial, December 15, 1991
Number of data points in data set:    6
Penetrometer plate diameter (cm):  15.0
Penetrometer sensitivity (N/div):  1.33
Volume of snow weighed (cm^3):  1000.00

Pressure(kPa)  Sinkage(m)  Density(g/cc)
     2.27         .070          .160
     6.04         .100          .227
     9.82         .300          .097
    13.59         .440          .307
    20.39         .450          .307
    17.37         .480          .275

Virgin density of data set (g/cm^3):  .097
Critical density for data (g/cm^3):   .307
Pressure at critical density (kPa):  20.39

After  455 cycles, K =   .78 at 14.3% error

After 7099 cycles, C =   .40 kPa and PHI =  10.50 deg, at 13.2% error
```

TMSNOW.DAT

TMSNOW.DAT

RHO, PHI, CC, RHOC, KK, SIGMAC

Notation

RHO	=	initial snow density, Kg/m^3
PHI	=	snow friction angle, rad
CC	=	snow cohesive strength, kPa
RHOC	=	snow critical density, Kg/m^3
KK	=	snow-plate footing parameter
SIGMAC	=	snow normal stress at critical density, kPa

This data file contains information regarding snow data. The option to calculate the snow data from raw penetrometer data is provided in the program. The user will be prompted for a file name to which the raw penetrometer and the computed snow data will be saved. For details, see PENET.DAT (page 3).

TMSNOW.DAT

300.0	.1832	.4049	307.0	.7846	20.39

TMSIZE.DAT

TMSIZE.DAT

'BV206'
RS, XS, YS
RR, N2
RSU, XSU, YSU
RTE, XTE, YTE
B
XR(1), YR(1)
⋮
XR(N2), YR(N2)

Notation

'BV206'	=	track model
RS	=	sprocket radius, m
XS	=	sprocket x-coordinate, m
YS	=	sprocket y-coordinate, m
RR	=	road wheel radius, m
N2	=	number of road wheels
RSU	=	support roller radius, m
XSU	=	support roller x-coordinate, m
YSU	=	support roller y-coordinate, m
RTE	=	tension wheel radius, m
XTE	=	tension wheel x-coordinate, m
YTE	=	tension wheel y-coordinate, m
B	=	belt width, m
XR(1)	=	1^{st} road wheel x-coordinate, m
YR(1)	=	1^{st} road wheel y-coordinate, m
⋮		
XR(N2)	=	n^{th} road wheel x-coordinate, m
YR(N2)	=	n^{th} road wheel y-coordinate, m

TMSIZE.DAT

```
'BV206'
 .19          0           0
 .19          4
 .1           1.28        .02
 .19          2.3313      .305
 .62
 .34          .305
 .8628        .305
 1.2928       .305
 1.7328       .305
```

TMLOAD.DAT

TMLOAD.DAT

W, WS, WNS, BW, TT, MU1
KS, M

Notation

W	=	track weight, kN
WS	=	sprung weight, kN
WNS	=	unsprung weight, kN
BW	=	belt width per unit length, kN/m
TT	=	preadjusted tension in belt, kN
MU1	=	internal friction coefficient
KS	=	suspension spring constant
M	=	torque applied on sprocket, kN/m

TMLOAD.DAT

```
28.06       23.56       4.5       .33       3.53       .3
69.0        2.0
```

TMRESULT.OUT

TMRESULT.OUT

Notation

FMAX	=	maximum force that can be applied, kN
MMAX	=	maximum torque that can be applied, $kN.m$
P	=	pressure, kPa
H	=	pressure bulb thickness, m
DELTA	=	displacement, mm
M(I)	=	applied torque, kN
TORAV(I)	=	shear stress, kPa
UU(I)	=	shear displacement, mm
SP(I)	=	slip rate, %
DP(I)	=	draw-bar pull, kN
ETAA(I)	=	tractive efficiency, %
MR(I)	=	motion resistance, kN

TMRESULT.OUT

1. MAX. FORCE POSSIBLE (FMAX) .... = 31.567 kN
2. MAX. MOMENT POSSIBLE (MMAX) ... = 7.797 kN.m
3. PRESSURE (P) .... = 23.820 kPa
4. PRESSURE BULB HEIGHT (H) ...... = 1353.601 mm
5. SINKAGE (DELTA) .... = 23.790 mm

No.	APPLIED TORQUE (kN.m)	SHEAR STRESS (kPa)	SHEAR DISPL (mm)	SLIP RATE (%)
1	.78	1.22	.08	2.28
2	1.56	2.44	.15	4.57
3	2.34	3.66	.23	6.85
4	3.12	4.88	.30	9.14
5	3.90	6.10	.38	11.42
6	4.68	7.32	.46	13.70
7	5.46	8.54	.53	15.99
8	6.24	9.75	.61	18.27
9	7.02	10.97	.69	20.56
10	7.80	12.19	.76	22.84

cont...

BEFORE SHEAR FAILURE:

No.	APPLIED TORQUE (kN.m)	SLIP RATE (%)	DRAW-BAR PULL (kN)	TRACTIVE EFFICIENCY (%)	MOTION RESIST (kN)
1	.78	2.28	2.12	50.53	1.98
2	1.56	4.57	4.90	57.02	3.30
3	2.34	6.85	7.62	57.67	4.69
4	3.12	9.14	10.27	56.85	6.15
5	3.90	11.42	12.84	55.44	7.68
6	4.68	13.70	15.33	53.74	9.29
7	5.46	15.99	17.74	51.88	10.99
8	6.24	18.27	20.05	49.91	12.78
9	7.02	20.56	22.25	47.86	14.68
10	7.80	22.84	24.35	45.78	16.69
11	7.80	27.84	24.35	42.81	16.69

AFTER SHEAR FAILURE:

No.	APPLIED TORQUE (kN.m)	SLIP RATE (%)	DRAW-BAR PULL (kN)	TRACTIVE EFFICIENCY (%)	MOTION RESIST (kN)
12	7.80	32.84	24.35	39.84	16.69
13	7.80	37.84	24.35	36.88	16.69
14	7.80	42.84	24.35	33.91	16.69
15	7.80	47.84	24.35	30.94	16.69
16	7.80	52.84	24.35	27.98	16.69
17	7.80	57.84	24.35	25.01	16.69
18	7.80	62.84	24.35	22.05	16.69
19	7.80	67.84	24.35	19.08	16.69
20	7.80	72.84	24.35	16.11	16.69
21	7.80	77.84	24.35	13.15	16.69

TMTRACK.FOR

```
C**********************************************************
C                                                          *
C   FORTRAN PROGRAM FOR ANAYSIS OF TRACK-SNOW INTERACTION  *
C                                                          *
C               GEOTECHNICAL RESEARCH CENTER               *
C                    McGILL UNIVERSITY                     *
C                        MONTREAL                          *
C                                                          *
C                     NOVEMBER 1993                        *
C                                                          *
C**********************************************************

      PROGRAM TMTRACK93

      IMPLICIT REAL(A-Z)
      INTEGER I,II,IM,N2
      CHARACTER*8  NAME
      DIMENSION XR(10),YR(10),M(25),TORAV(10),UU(10),
     +SP(25),DP(25),ETAA(25),MR(25)
      COMMON /SIZE/ NAME,RS,XS,YS,RR,N2,RSU,XSU,YSU,
     +RTE,XTE,YTE,B,XR,YR
      COMMON /LOAD/ W,WS,WNS,BW,T,MU1,KS,MM
      COMMON /SNOW/ RHO,PHI,CC,RHOC,KK,SIGMAC

C     **** INPUT DATA CONTAINED IN 3 FILES: *****
C     **** TMSIZE.DAT, TMLOAD.DAT, TMSNOW.DAT ***

      DO 5 I=1,10
         XR(I)=0.0
         YR(I)=0.0
5     CONTINUE
      OPEN (1,FILE='TMSIZE.DAT')
      READ (1,*) NAME,RS,XS,YS,RR,N2,RSU,XSU,YSU,RTE,XTE,YTE,B
      DO 10 I=1,N2
         READ (1,*) XR(I),YR(I)
10    CONTINUE
      CLOSE (1)

      OPEN (1,FILE='TMLOAD.DAT')
      READ (1,*) W,WS,WNS,BW,T,MU1,KS,MM
      CLOSE (1)

C     OPEN (1,FILE='TMSNOW.DAT')
```

```
C       READ (1,*) RHO,PHI,CC,RHOC,KK,SIGMAC
C       CLOSE (1)

C       **** OUTPUT TO BE STORED IN TMRESULT.OUT *****
        OPEN (2,FILE='TMRESULT.OUT')

        CALL CLEARSCREEN
        CALL FIG1()
        CALL CLEARSCREEN


        CALL FIG2()
        CALL WAIT ()
        CALL CLEARSCREEN

        CALL FIG3()
        CALL WAIT ()
        CALL CLEARSCREEN

        CALL SNOW_DATA (RHO,PHI,CC,RHOC,KK,SIGMAC)
        CALL CLEARSCREEN

        CALL FIG4()
        CALL WAIT ()
        CALL CLEARSCREEN

        CALL COM (FMAX,MMAX,P,H,DELTA,II,IM,M,SP,DP,ETAA,MR,UU,TORAV)

        CALL FIG5 (FMAX,MMAX,P,H,DELTA,II,M,TORAV,UU,SP)
        CALL WAIT ()
        CALL CLEARSCREEN

        CALL FIG5B (FMAX,MMAX,P,H,DELTA,II,M,TORAV,UU,SP)

        CALL FIG6 (II,IM,M,SP,DP,ETAA,MR)
        CALL WAIT ()
        CALL CLEARSCREEN

        CALL FIG6B (II,IM,M,SP,DP,ETAA,MR)
        CLOSE (2)

        CALL FIG7 ()
        CALL WAIT_EXIT ()
```

```
      CALL CLEARSCREEN
      END

      SUBROUTINE WAIT ()
        PAUSE    '                         Press <ENTER> to continue ...'
      END


      SUBROUTINE WAIT_EXIT ()
        PAUSE    '                            Press <ENTER> to exit.'
      END

      SUBROUTINE FIG1 ()
        WRITE (6,'(A)')
     +'       TTTTTTTTTT  RRRRRRRR   AAAAAAAA    CCCCCCC   KK    KKKKK',
     +'           TT      RR    RRR  AA     AA  CCC        KK   KKK  ',
     +'           TT      RR    RRR  AA     AA  CCC        KK  KK    ',
     +'           TT      RRRRRR     AAAAAAAA   CCC        KKKK      ',
     +'           TT      RR RRRR    AA     AA  CCC        KK  KK    ',
     +'           TT      RR    RRR  AA     AA  CCC        KK   KKK  ',
     +'           TT      RR     RRR AA     AA   CCCCCCC   KK    KKKK'

      CALL SPACE (3)
      WRITE (6,*)  '                        TRACK-SNOW INTERACTION MODEL'
      WRITE (6,*)  '                               FORTRAN EDITION'
      WRITE (6,*)  '                                NOVEMBER 1993'
      DO 55 N=1,3
55    WRITE (*,*)
      WRITE (6,*)  '                         GEOTECHNICAL RESEARCH CENTER'
      WRITE (6,*)  '                              McGILL UNIVERSITY'
      WRITE (6,*)  '                                   MONTREAL'
      DO 57 N=1,3
57    WRITE (*,*)
         PAUSE    '                         Press <ENTER> to continue ...'

      END


      SUBROUTINE SNOW_DATA (RHOO,PHI_,CC,RHOC,KK,SIGMAC)
      REAL K, KK, RHOO, PHI_, PI
      INTEGER RESPN
      COMMON N, D, SIGMA(100), RHO(100), SINK(100),
     +       SIGMAX, RHOMAX, RHOMIN, K, C, PHI
```

```
      PI = 3.1415926
      PRINT *,'                                    SNOW DATA   '
      PRINT *,'                                    ---------   '
 1    PRINT *, ' '
      PRINT *,'                            To obtain snow data from: '
      PRINT *, ' '
      PRINT *,'                        1. file TMSNOW.DAT, Enter 1'
      WRITE (*,'(A\)')
     +      '                  2. raw PENETROMETER data, Enter 2: '
      READ(*,*) RESPN
      IF(RESPN .EQ. 1) THEN
        OPEN (1,FILE='TMSNOW.DAT')
        READ (1,*) RHO0,PHI_,CC,RHOC,KK,SIGMAC
        CLOSE (1)
      ENDIF
      IF (RESPN .EQ. 2) THEN
        CALL SPACE (2)
        CALL PENET
        CC = C
        KK = K
        SIGMAC = SIGMAX
        RHOC = RHOMAX*1000.0
        PHI_ = PHI*PI/180.0
        CALL SPACE (3)
        WRITE (*,'(A\)') ' Enter initial snow density RHO (kg/m^3): '
        READ (*,*) RHO0
      ENDIF
      IF (RESPN .NE. 1 .AND. RESPN .NE. 2) GOTO 1
      END


      SUBROUTINE CLEARSCREEN ()
        DO 19 I=1,25
 19     WRITE (*,*)
      END

      SUBROUTINE SPACE (N)
        DO 21 I=1,N
 21     WRITE (*,*)
      END

      SUBROUTINE FIG2 ()
      IMPLICIT REAL(A-H,O-Z)
```

```
      INTEGER N2
      CHARACTER*8  NAME
      DIMENSION XR(10),YR(10)
      COMMON /SIZE/ NAME,RS,XS,YS,RR,N2,RSU,XSU,YSU,RTE,XTE,YTE,
     +              B,XR,YR

      WRITE (6,'(A,A)')
     +  '                          ---> TRACK MODEL: ', NAME
      CALL SPACE (10)
      CALL WAIT ()
      CALL CLEARSCREEN
      PRINT *, ' '
      PRINT *, ' '
      PRINT *, '                               TRACK SIZE DATA '
      PRINT *, '                               --------------- '
      PRINT *, ' '

      WRITE (6,'(A)')
     +  '            1. SPROCKET   '
        PRINT *, ' '
        WRITE (6,'(A,F9.4,A)')
     +  '                  radius (RS) ............... = ', RS, ' m'
        WRITE (6,'(A,F9.4,A)')
     +  '                  x-coordinate (XS) .......... = ', XS, ' m'
        WRITE (6,'(A,F9.4,A)')
     +  '                  y-coordinate (YS) .......... = ', YS, ' m'

      CALL SPACE (6)
      PRINT *, ' '
      CALL WAIT ()
      CALL CLEARSCREEN

      PRINT *, ' '
      PRINT *, '                             TRACK SIZE DATA (contd)'
      PRINT *, '                             ----------------------- '
      PRINT *, ' '
      WRITE (6,'(A)')
     +  '            2. ROAD WHEELS    '
        PRINT *, ' '
        PRINT *,
     +  '              -------------------------------------------'
        PRINT *,
     +  '              No.       RADIUS       X-COORD       Y-COORD'
```

```
      PRINT *,
     +  '                              (m)            (m)            (m)'
      PRINT *,
     +  '                  -------------------------------------------'
     DO 450 I=1,N2
        PRINT 400,I,RR,XR(I),YR(I)
400     FORMAT(16X,I2,3(5X,F9.4))
450  CONTINUE

     CALL SPACE (2)
     PRINT *, ' '
     CALL WAIT ()
     CALL CLEARSCREEN

     PRINT *, ' '
     PRINT *, '                         TRACK SIZE DATA (contd)'
     PRINT *, '                         ----------------------- '
     PRINT *, ' '
       WRITE (6,'(A)')
     +  '            3. SUPPORT ROLLER '
       PRINT *, ' '
       WRITE (6,'(A,F9.4,A)')
     +  '                 radius (RSU) ............... = ', RSU, ' m'
       WRITE (6,'(A,F9.4,A)')
     +  '                 x-coordinate (XSU) .......... = ', XSU, ' m'
       WRITE (6,'(A,F9.4,A)')
     +  '                 y-coordinate (YSU) .......... = ', YSU, ' m'

     CALL SPACE (2)

     WRITE (6,'(A)')
     +  '            4. TENSION WHEEL'
       PRINT *, ' '
       WRITE (6,'(A,F9.4,A)')
     +  '                 radius (RTE) ............... = ', RTE, ' m'
       WRITE (6,'(A,F9.4,A)')
     +  '                 x-coordinate (XTE) .......... = ', XTE, ' m'
       WRITE (6,'(A,F9.4,A)')
     +  '                 y-coordinate (YTE) .......... = ', YTE, ' m'

     CALL SPACE (2)

     WRITE (6,'(A)')
```

```
     +  '               5. BELT WIDTH'
          PRINT *, ' '
          WRITE (6,'(A,F9.4,A)')
     +  '                        belt width (B) ............. = ', RTE, ' m'

        CALL SPACE (2)
        PRINT *, ' '
        END

        SUBROUTINE FIG3 ()
        IMPLICIT REAL(A-Z)
        COMMON /LOAD/ W,WS,WNS,BW,T,MU1,KS,MM

        PRINT 50
50      FORMAT(///)
        PRINT *,'                             TRACK LOAD DATA  '
        PRINT *,'                             ---------------       '
        PRINT *, ' '
        PRINT 100, W
100     FORMAT(/,11X,' 1. TRACK WEIGHT (W) ............ = ',F8.3,' kN')
        PRINT 200,WS
200     FORMAT(/,11X,'      a. Sprung weight (WS) ...... = ',F8.3,' kN')
        PRINT 300,WNS
300     FORMAT(/,11X,'      b. Unsprung weight (WNS) ... = ',F8.3,' kN')
        PRINT 400,BW
400     FORMAT(/,11X,' 2. BELT WEIGHT/LENGTH (BW) ...... = ',F8.3,
     +        ' kN.m')
        PRINT 500,T
500     FORMAT(/,11X,' 3. TRACK INITIAL TENSION (T) .... = ',F8.3,' kN')
        PRINT 600,MU1
600     FORMAT(/,11X,' 4. INTERNAL FRICTION COEFF (MU1). = ',F8.3)
        PRINT 700,KS
700     FORMAT(/,11X,' 5. SUSPENSION SPRING CONST (KS) . = ',F8.3,
     +        ' kN/m')
        PRINT 800,MM
800     FORMAT(/,11X,' 6. TORQUE (M) .................. = ',F8.3,
     + ' KN.m'/)

        PRINT *, ' '
        END

        SUBROUTINE FIG4()
        IMPLICIT REAL(A-Z)
```

```
      CHARACTER RESP*1
      COMMON /SNOW/ RHO,PHI,CC,RHOC,KK,SIGMAC

      PRINT *,'                                         SNOW DATA  '
      PRINT *,'                                         ---------  '
      PRINT *, ' '
      PRINT 100, RHO
100   FORMAT(/,11X,' 1. INITIAL DENSITY (RHO) ........ = ',F10.3,
     +' kg/m3')
      PRINT 400, PHI
400   FORMAT(/,11X,' 2. FRICTION ANGLE (PHI) ......... = ',F10.3,
     +       ' rad')
      PRINT 500, CC
500   FORMAT(/,11X,' 3. COHESIVE STRENGTH (CC) ....... = ',F10.3,
     +' kPa')
      PRINT 600, KK
600   FORMAT(/,11X,' 4. PLATE FOOTING PARAMETER (KK) . = ',F10.3)
      PRINT 700, RHOC
700   FORMAT(/,11X,' 5. CRITICAL DENSITY (RHOC) ...... = ',F10.3,
     +       ' kg/m3')
      PRINT 800, SIGMAC
800   FORMAT(/,11X,' 6. NORMAL STRESS AT RHOC (SIGMAC) = ',F10.3,
     + ' kPa'/)

    2 CALL SPACE (2)
      WRITE (*,'(A\)')
     + '           ---> Replace TMSNOW.DAT with above snow data? (Y/N): '
      READ (*,'(A1)') RESP
      IF (RESP .EQ. 'Y' .OR. RESP .EQ. 'y') THEN
         OPEN (1,FILE='TMSNOW.DAT')
         WRITE (1,'(6G12.4,1X)') RHO,PHI,CC,RHOC,KK,SIGMAC
         CLOSE (1)
      ENDIF
         CALL SPACE (3)
      IF (RESP .NE. 'Y' .AND. RESP .NE. 'y' .AND.
     +   RESP .NE. 'N' .AND. RESP .NE. 'n') GOTO 2
      END


      SUBROUTINE COM (FMAX,MMAX,P,H,DELTA,II,IM,M,SP,DP,
     +                ETAA,MR,UU,TORAV)
      IMPLICIT REAL(A-Z)
      INTEGER I,II,IM,J,JJ,NN,N2
```

```
      CHARACTER*8  NAME
      DIMENSION XR(10),YR(10),M(25),TORAV(10),U(10),UU(10),S(25),
     +          SP(25),DP(25),ETA(25),ETAA(25),MR(25)
      COMMON /SIZE/ NAME,RS,XS,YS,RR,N2,RSU,XSU,YSU,RTE,XTE,YTE,
     +              B,XR,YR
      COMMON /LOAD/ W,WS,WNS,BW,T,MU1,KS,MM
      COMMON /SNOW/ RHO,PHI,CC,RHOC,KK,SIGMAC
      PI = 3.1415926

C     ***** PHYSICAL UNIT IN METRES *****
      L = 2*RR
      A = L*B

C     ***** PHYSICAL UNIT IN MM *****
      D = SQRT(4*A/PI)*1000.0
      P = W/A/(N2+1)
      H = P*D/2/(CC+P*TAN(PHI))
      DELTA = (RHOC-RHO)/RHO*H*2*(P/SIGMAC)**(KK)/3

C     ***** PHYSICAL UNIT IN METRES *****
      TORMAX = 0.5625*P
      FMAX = TORMAX*A*(N2+1)*2
      MMAX = FMAX*RR*(1+MU1)
      NN = 10
C     ***** UNIT TORQUE INCREMENT *****
      DM = MMAX/NN
      DO 302 I=1,NN
         M(I)=DM*I
         TORAV(I)=M(I)*(1-MU1)/RS/((N2+1)*2*A)
302   CONTINUE
      DO 304 I=1,NN
         U(I) = TORAV(I)/16/1000
         UU(I) = 1000*U(I)
304   CONTINUE
      DO 502 I=1,NN
         S(I) = U(I)*H/50.8/.0889
         IF (S(I) .GT. 1) GO TO 602
502   CONTINUE
      II = NN
      GO TO 700
602   II = I-1
700   CONTINUE
```

```
C       ***** PHYSICAL UNIT IN METRES *****
        HB = ABS(YS - YR(1))
        LL = ABS(XTE - XR(1))

        DO 800 I=1,II
           TERM1 = M(I)*(1-S(I)-MU1)
           TERM2 = 2*W*(DELTA/1000)*(1-S(I))*RS/LL
           TERM3 = BW*RS*HB/2
           DP(I) = (TERM1-TERM2-TERM3)/RS/(1-S(I))
           IF (TERM1 .LT. (TERM2+TERM3))  DP(I) = 0
           ETA(I) = DP(I)*RS*(1-S(I))/M(I)
           ETAA(I) = 100*ETA(I)
           MR(I) = M(I)/RS - DP(I)
           SP(I) = 100*S(I)
800     CONTINUE

        DPF = DP(II)
        MF = M(II)
        SF = S(II)

        DO 900 J=1,15
           JJ=II+J
           DP(II+J)=DPF
           S(JJ)=SF+ J*.05
           SP(JJ)=100*S(JJ)
           IF (S(JJ).GT. 0.8) GO TO 901
           M(JJ)=MF
           ETA(JJ)=DP(JJ)*RS*(1-S(JJ))/MF
           ETAA(JJ)=100*ETA(JJ)
           MR(JJ) = MF/RS - DP(JJ)
900     CONTINUE
        IM=JJ
        GO TO 902
901     IM=JJ-1
902     CONTINUE
        END

        SUBROUTINE FIG5 (FMAX,MMAX,P,H,DELTA,II,M,TORAV,UU,SP)
        IMPLICIT REAL(A-Z)
        INTEGER I,II
        DIMENSION M(25),TORAV(10),UU(10),SP(25)

        PRINT *,'                                        RESULTS    '
```

```
      PRINT *,'                                   -------    '
      PRINT *, ' '
      PRINT 100, FMAX
 100  FORMAT(/,11X,' 1. MAX. FORCE POSSIBLE (FMAX) .... = ',F10.3,
     +' kN')
      PRINT 400, MMAX
 400  FORMAT(/,11X,' 2. MAX. MOMENT POSSIBLE (MMAX) ... = ',F10.3,
     +        ' kN.m')
      PRINT 500, P
 500  FORMAT(/,11X,' 3. PRESSURE (P) ................. = ',F10.3,
     +' kPa')
      PRINT 600, H
 600  FORMAT(/,11X,' 4. PRESSURE BULB HEIGHT (H) ...... = ',F10.3,
     +' mm')
      PRINT 700, DELTA
 700  FORMAT(/,11X,' 5. SINKAGE (DELTA) ............... = ',F10.3,
     +        ' mm')

      CALL SPACE (5)
      CALL WAIT ()
      CALL CLEARSCREEN

      PRINT *,'                          RESULTS (contd)   '
      PRINT *,'                          ---------------   '
      PRINT *,'  '

      WRITE (*, '(A)')
     +        '          -------------------------------------------'
      WRITE (*, '(A)')
     +        '                  APPLIED     SHEAR      SHEAR      SLIP',
     +        '          No.     TORQUE      STRESS     DISPL      RATE'
      WRITE (*, '(A)')
     +        '                  (kN.m)      (kPa)      (mm)       (%)'
      WRITE (*, '(A)')
     +        '          -------------------------------------------'
      DO 1000 I=1, II
        PRINT 1001, I,M(I),TORAV(I),UU(I),SP(I)
1001    FORMAT(9X,I2,2X,F9.2,3X,F9.2,2X,F9.2,2X,F9.2)
1000  CONTINUE
      CALL SPACE (2)
      END

      SUBROUTINE FIG5B (FMAX,MMAX,P,H,DELTA,II,M,TORAV,UU,SP)
```

```
      IMPLICIT REAL(A-Z)
      INTEGER I,II
      DIMENSION M(25),TORAV(10),UU(10),SP(25)
      WRITE (2, 100) FMAX
100   FORMAT(/,8X,' 1. MAX. FORCE POSSIBLE (FMAX) .... = ',F10.3,
     +' kN')
      WRITE (2, 400) MMAX
400   FORMAT(/,8X,' 2. MAX. MOMENT POSSIBLE (MMAX) ... = ',F10.3,
     +       ' kN.m')
      WRITE (2, 500) P
500   FORMAT(/,8X,' 3. PRESSURE (P) ................. = ',F10.3,
     +' kPa')
      WRITE (2, 600) H
600   FORMAT(/,8X,' 4. PRESSURE BULB HEIGHT (H) ...... = ',F10.3,
     +' mm')
      WRITE (2, 700) DELTA
700   FORMAT(/,8X,' 5. SINKAGE (DELTA) ............... = ',F10.3,
     +       ' mm')

      DO 710 NO=1,4
710   WRITE (2,'(A)') ' '

      WRITE (2, '(A)')
     +       '         ------------------------------------------------'
      WRITE (2, '(A)')
     +       '                 APPLIED     SHEAR       SHEAR       SLIP',
     +       '         No.     TORQUE      STRESS      DISPL       RATE'
      WRITE (2, '(A)')
     +       '                 (kN.m)      (kPa)        (mm)        (%)'
      WRITE (2, '(A)')
     +       '         ------------------------------------------------'
      DO 1000 I=1, II
        WRITE(2,1001) I,M(I),TORAV(I),UU(I),SP(I)
1001    FORMAT(9X,I2,2X,F9.2,3X,F9.2,2X,F9.2,2X,F9.2)
1000  CONTINUE
      DO 1110 NO=1,4
1110  WRITE (2,'(A)') ' '
      END

      SUBROUTINE FIG6 (II,IM,M,SP,DP,ETAA,MR)
      IMPLICIT REAL(A-Z)
      INTEGER I,II,IM
      DIMENSION M(25),SP(25),DP(25),ETAA(25),MR(25)
```

```
C      **** PRINT RESULTS BEFORE SHEAR FAILURE ****

       PRINT *,'                                   RESULTS (contd)   '
       PRINT *,'                                   ---------------   '
       PRINT *, ' '
       PRINT *,
     + '                              BEFORE SHEAR FAILURE: '
       PRINT *, ' '

       WRITE (*,'(8X,A)')
     + ' ---------------------------------------------------------------',
     + '          APPLIED     SLIP       DRAW-BAR   TRACTIVE      MOTION',
     + ' No.      TORQUE      RATE        PULL     EFFICIENCY     RESIST',
     + '          (kN.m)      (%)         (kN)        (%)         (kN) ',
     + ' ---------------------------------------------------------------'
       DO 1200 I=1,II
          PRINT 911, I,M(I),SP(I),DP(I),ETAA(I),MR(I)
911       FORMAT(9X,I2,2(2X,F9.2),3(3X,F9.2))
1200   CONTINUE

       PRINT *, ' '
       CALL WAIT ()
       CALL CLEARSCREEN

C      ***** RESULTS AFTER SHEAR FAILURE *****

       PRINT *,'                                   RESULTS (contd)   '
       PRINT *,'                                   ---------------   '
       PRINT *, ' '
       PRINT *, ' '
       PRINT *,
     + '                              AFTER SHEAR FAILURE: '
         PRINT *, ' '
       WRITE (*,'(8X,A)')
     + ' ---------------------------------------------------------------',
     + '          APPLIED     SLIP       DRAW-BAR   TRACTIVE      MOTION',
     + ' No.      TORQUE      RATE        PULL     EFFICIENCY     RESIST',
     + '          (kN.m)      (%)         (kN)        (%)         (kN) ',
     + ' ---------------------------------------------------------------'
       II=II+1
       DO 1800 I=II,IM
          PRINT 911, I,M(I),SP(I),DP(I),ETAA(I),MR(I)
```

```
 1800 CONTINUE
      PRINT *, ' '
      END

      SUBROUTINE FIG6B (II,IM,M,SP,DP,ETAA,MR)
      IMPLICIT REAL(A-Z)
      INTEGER I,II,IM
      DIMENSION M(25),SP(25),DP(25),ETAA(25),MR(25)

C     **** PRINT RESULTS BEFORE SHEAR FAILURE ****

      WRITE (2,'(A)')
     + '                                  BEFORE SHEAR FAILURE: '
      WRITE (2,'(A)') ' '

      WRITE (2,'(8X,A)')
     + ' --------------------------------------------------------------',
     + '         APPLIED     SLIP      DRAW-BAR  TRACTIVE       MOTION',
     + ' No.     TORQUE      RATE        PULL    EFFICIENCY     RESIST',
     + '         (kN.m)      (%)         (kN)       (%)          (kN) ',
     + ' --------------------------------------------------------------'
      DO 1200 I=1,II
         WRITE (2,911) I,M(I),SP(I),DP(I),ETAA(I),MR(I)
 911     FORMAT(9X,I2,2(2X,F9.2),3(3X,F9.2))
 1200 CONTINUE

      DO 1210 NO=1,4
 1210 WRITE (2,'(A)') ' '

C     ***** RESULTS AFTER SHEAR FAILURE *****

      WRITE (2,'(A)')
     + '                                  AFTER SHEAR FAILURE: '
      WRITE (2,'(A)') ' '
      WRITE (2,'(8X,A)')
     + ' --------------------------------------------------------------',
     + '         APPLIED     SLIP      DRAW-BAR  TRACTIVE       MOTION',
     + ' No.     TORQUE      RATE        PULL    EFFICIENCY     RESIST',
     + '         (kN.m)      (%)         (kN)       (%)          (kN) ',
     + ' --------------------------------------------------------------'
      II=II+1
      DO 1800 I=II,IM
         WRITE (2,911) I,M(I),SP(I),DP(I),ETAA(I),MR(I)
```

```
 1800 CONTINUE
      END

      SUBROUTINE FIG7 ()
       WRITE (6,'(A)')
     +'                    EEEEEEEE    NNN          NN   DDDDDDDDDD ',
     +'                    EE          NN NN        NN    DD       DD',
     +'                    EE          NN  NN       NN    DD       DD',
     +'                    EEEEEEEE    NN    NN     NN    DD       DD',
     +'                    EE          NA      NN   NN    DD       DD',
     +'                    EE          NN        NN NN    DD       DD',
     +'                    EEEEEEEE    NN          NNN   DDDDDDDDDD '

      CALL SPACE (3)
      WRITE (6,'(A)')
     +'                  ---> RESULTS ARE STORED IN TMRESULT.OUT '
      CALL SPACE (3)
      WRITE (6,'(A)')
     +'                        GEOTECHNICAL RESEARCH CENTER',
     +'                             McGILL UNIVERSITY',
     +'                                  MONTREAL'
      CALL SPACE (3)

      END

C.....................................................................
C           PENET.FOR   - Alan McCormac (DRES)                        .
C                                                                     .
C     This program receives penetrometer field data and finds the     .
C     engineering parameters K, C, and PHI for the snow tested.       .
C.....................................................................
      SUBROUTINE PENET
      CHARACTER DFILE*20, LINE*80, RESP*1
      REAL K, K0, NEWERR
C
      COMMON N, D, SIGMA(100), RHO(100), SINK(100),
     +       SIGMAX, RHOMAX, RHOMIN, K, C, PHI
C
    1 SIGMAX=0.0
      RHOMAX=0.0
      RHOMIN=1.0
      K0=0.2
      C0=1.0
```

```
      PHI0=2.0
      DELTA=.001
      PI=3.1415926
C
      WRITE(*,'(A\)') ' Enter name of file to save data in: '
      READ(*,'(A)') DFILE
      OPEN(1,FILE=DFILE,STATUS='UNKNOWN',ERR=1)
      WRITE(*,*) 'Enter a one-line description of the data set:'
      READ(*,'(A)') LINE
      WRITE(1,'(1X,A)') LINE
      WRITE(*,'(A\)') ' Enter the number of data points: '
      READ(*,*) N
      WRITE(1,40) N
C
C     Program uses plate diameter in meters, but users prefer cm
C
      WRITE(*,'(A\)') ' Enter plate diameter in centimeters: '
      READ(*,*) D
      WRITE(1,41) D
      D=D/100.
C
C     Program uses plate pressure in kPa, but users read divisions
C     Require a conversion factor from user
C
      WRITE(*,'(A\)') ' Enter penetrometer constant in N/div: '
      READ(*,*) P
      WRITE(1,42) P
C
C     Program uses density in g/cc, but users know mass (g) of a volume (cc)
C
      WRITE(*,'(A\)') ' Enter volume of snow weighed in cc: '
      READ(*,*) V
      WRITE(1,43) V
C
C     Now receive the usual field measurements
C
      WRITE(*,*)
      WRITE(*,*) 'Enter each test point on one line, in the format:'
      WRITE(*,*) 'PENETROMETER(div)  SINKAGE(mm)  MASS(g)'
      WRITE(1,44)
      DO 10 I=1,N
        READ(*,*) SIGMA(I),SINK(I),RHO(I)
C
```

```
C     Convert SIGMA to kPa, SINK to meters, and RHO to g/cc, BEFORE saving
C
        SIGMA(I)=(SIGMA(I)*P/(PI*(D/2)**2))/1000.
        SINK(I)=SINK(I)/1000.
        RHO(I)=RHO(I)/V
        WRITE(1,45) SIGMA(I),SINK(I),RHO(I)
C
C     Grab the lowest density as virgin density, and the highest
C     (pressure,density) pair as the critical density point.
C
        IF(SIGMA(I).GT.SIGMAX) SIGMAX=SIGMA(I)
        IF(RHO(I).GT.RHOMAX) RHOMAX=RHO(I)
        IF(RHO(I).LT.RHOMIN) RHOMIN=RHO(I)
C
   10 CONTINUE

      WRITE (*,*)
   12 WRITE(*,'(A\)')' DISCARD DATA AND START AGAIN? Y/N: '
      READ(*,'(A1)') RESP
      IF (RESP .EQ. 'Y' .OR. RESP .EQ. 'y') GOTO 1
      IF (RESP .NE. 'Y' .AND. RESP .NE. 'y' .AND.
     +    RESP .NE. 'N' .AND. RESP .NE. 'n') GOTO 12
      WRITE(*,*)
      WRITE(*,*) 'This program now assumes the maximum pressure and ',
     + 'density entered, defines'
      WRITE(*,*) 'the critical density point, and evaluates K under ',
     + 'that assumption ...'
      WRITE(*,*)
      WRITE(1,46) RHOMIN
      WRITE(1,47) RHOMAX
      WRITE(1,48) SIGMAX
C
C     Using the initial value of K set at the start of the program,
C     compute the sum of squared errors between theoretical and
C     measured values of density for each measured pressure.
C
      K=K0
      CALL RHOERR(BASERR)
C
C     Perform a gradient search of the least squared error between
C     theoretical and measured densities, to find the value of K
C
      LOOP=0
```

```
   20 K=K+DELTA
      LOOP=LOOP+1
      CALL RHOERR(DELERR)
      K=K-10.0*(DELERR-BASERR)
      CALL RHOERR(NEWERR)
      IF(NEWERR.LE.BASERR.AND.LOOP.LT.9999) THEN
        BASERR=NEWERR
        GOTO 20
      ELSE
        K=K+10.0*(DELERR-BASERR)
      ENDIF
C
      WRITE(*,*)
      WRITE(*,49) LOOP,K,SQRT(BASERR)*100.
      WRITE(*,*)
      WRITE(1,49) LOOP,K,SQRT(BASERR)*100.
C
      WRITE(*,*)
      WRITE(*,*) 'The program now uses that K to find the best fit ',
     + 'values of C and PHI ...'
      WRITE(*,*)
C
C     Now using the initial values of C and PHI given at the start,
C     and the calculated value of K, the program iterates to find
C     the C and PHI for least squared error between theoretical and
C     measured sinkage for each point in the data set.
C
      C=C0
      PHI=PHI0
      CALL SNKERR(BASERR)
C
C     Perform a gradient search of the least squared error between
C     theoretical and measured sinkages, to find the values of C and PHI
C
      LOOP=0
   30 LOOP=LOOP+1
C
C     First tweak C and get the error sensitivity, CERR
C
      C=C+DELTA
      CALL SNKERR(DELERR)
      CERR=(DELERR-BASERR)/DELTA
      C=C-DELTA
```

```
C
C     Then tweak PHI and get the error sensitivity, PHIERR
C
      PHI=PHI+DELTA
      CALL SNKERR(DELERR)
      PHIERR=(DELERR-BASERR)/DELTA
      PHI=PHI-DELTA
C
C     Then use the RMS error sensitivity to alter both for the next try
C
      CERR=CERR/SQRT(CERR**2+PHIERR**2)
      PHIERR=PHIERR/SQRT(CERR**2+PHIERR**2)
      C=C-10*CERR*DELTA
      PHI=PHI-10*PHIERR*DELTA
C
      CALL SNKERR(NEWERR)
      IF(NEWERR.LT.BASERR.AND.LOOP.LT.9999) THEN
        BASERR=NEWERR
        GOTO 30
      ELSE
        C=C+10*CERR*DELTA
        PHI=PHI+10*PHIERR*DELTA
      ENDIF
C
      WRITE(*,50) LOOP,C,PHI,SQRT(BASERR)*100.
      WRITE(*,*)
      WRITE(1,50) LOOP,C,PHI,SQRT(BASERR)*100.
      WRITE (1,*)
      CLOSE(1)
C
C     Now see if there are more data sets to process
C

      WRITE(*,'(A\)') ' Do you wish to process another data set? Y/N: '
      READ(*,'(A1)') RESP
      IF(RESP.EQ.'Y'.OR.RESP.EQ.'y') GOTO 1
C
C     FORMAT statements for output on screen and to DFILE
C
   40 FORMAT(' Number of data points in data set: ',I4)
   41 FORMAT(' Penetrometer plate diameter (cm): ',F5.1)
   42 FORMAT(' Penetrometer sensitivity (N/div): ',F5.2)
   43 FORMAT(' Volume of snow weighed (cm^3): ',F8.2)
```

```
   44 FORMAT(/,' Pressure(kPa)  Sinkage(m)  Density(g/cc)')
   45 FORMAT(F10.2,F12.3,F14.3)
   46 FORMAT(/,' Virgin density of data set (g/cm^3): ',F5.3)
   47 FORMAT(' Critical density for data (g/cm^3): ',F6.3)
   48 FORMAT(' Pressure at critical density (kPa): ',F6.2)
   49 FORMAT(/,' After ',I4,' cycles, K = ',F5.2,' at ',F4.1,'% error')
   50 FORMAT(/,' After ',I4,' cycles, C = ',F5.2,' kPa and PHI = ',F6.2
     +       ,' deg, at ',F4.1,'% error')
      END
C
C     Subroutine RHOERR: finds the sum of the squared error between
C     theoretical density and measured density for each measured
C     pressure with an assumed value of K.
C
      SUBROUTINE RHOERR(ERROR)
C
      REAL K
      COMMON N, D, SIGMA(100), RHO(100), SINK(100),
     +       SIGMAX, RHOMAX, RHOMIN, K, C, PHI
C
      ERROR = 0.0
      DO 101 I=1,N
        ERROR = ERROR + ((RHO(I)-RHOMIN-(RHOMAX-RHOMIN)*
     +          (SIGMA(I)/SIGMAX)**K))**2
  101 CONTINUE
      RETURN
      END
C
C     Subroutine SNKERR: finds the sum of the squared error between
C     theoretical sinkage and measured sinkage for each measured
C     pressure with the previously calculated value of K, and the
C     assumed values of C and PHI.
C
      SUBROUTINE SNKERR(ERROR)
C
      REAL K
      COMMON N, D, SIGMA(100), RHO(100), SINK(100),
     +       SIGMAX, RHOMAX, RHOMIN, K, C, PHI
C
      ERROR = 0.0
      DO 201 I=1,N
        ERROR = ERROR + ((SINK(I)-(RHOMAX-RHOMIN)/(3.0*RHOMIN)*
     +          (SIGMA(I)*D)/(C+SIGMA(I)*TAN(PHI*3.1416/180.))*
```

```
     +              (SIGMA(I)/SIGMAX)**K))**2
 201 CONTINUE
     RETURN
     END
```

UNCLASSIFIED