

不同沈下シミュレーションプログラム FUT090 マニュアル

(独)港湾空港技術研究所
地盤・構造部土質研究室

注:本マニュアル記載の内容について、著作権は港湾空港技術研究所に属します。記載内容をご利用頂くことは自由(販売目的で利用される場合を除く)ですが、これを利用することにより生じるいかなる不利益に対しても、港湾空港技術研究所は一切の責任を負いません。

1. プログラムの概要

広域の埋立を行う場合、沈下量の予測はいくつかの限られた地点で実施し、これをそのエリアにおける代表的な沈下の予測値とすることが多い。しかし、海上を埋め立てて造成した空港施設の場合、滑走路や誘導路などの設計、さらには地盤改良の設計を合理的に行うには、代表点の沈下だけではなく埋立後の空港島内での面的な沈下のばらつき(不同沈下)を定量的に予測することが不可欠である。本プログラムはこの目的のために開発された。

本プログラムでは地盤の不均一性を確率モデルによって表すことが基本となっている。不均一性地盤をモデル化するためには、地盤の空間的な確率モデルを考える必要がある。圧密現象を支配する定数として次のものを用いている。

- 圧密係数 c_v
- 圧縮指数 C_c
- 圧密降伏圧力 p_c

これらの定数は統計的な性質を調べることによってほぼ正規分布または対数正規分布することが明らかになっており、計算においては確率変数として与えられる。このほかに軟弱層の厚さも確率変数と考えられるが、現在のところは軟弱層厚が均一で事前の調査によってそれぞれの位置で把握できる場合を考え、計算の中では確定値として与えている。

シミュレーションを行うにはまず海底地盤を図-1のように平面的なメッシュに分割し、それぞれのメッシュで独立に(隣あったメッシュとは無関係に)沈下が生じると仮定してメッシュ中心点での沈下を一次元圧密理論によって計算する。また、地盤の深度方向の不均一性を考慮するため地盤は図のように鉛直方向にも分割される。これら水平方向および鉛直方向のメッシュの大きさは海底地盤の土質諸定数が変化するサイズによって決められるが、これは定数の自己相関性を調べることによって明らかになる。地盤の自己相関性とは、どの程度の範囲の地盤を確率的に同一な地盤とみなし得るかという目安となるものである。通常は深さ方向に4~5m、水平方向に50~100m程度の相関があるとされており、これらの値を基準にして地盤を分割する。

このようにして分割された各ブロックの土質定数は、それぞれある確率モデルに従う確率変数としてランダムに割り付けられ、ブロック内の土質定数は均一である。すなわち、本プログラムでは同一種類の地盤であるということは、ブロックの土質定数が同じ確率分布に属することを意味する。同一種類の地盤でもブロッ

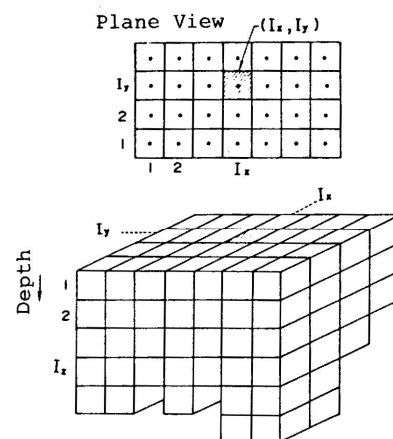


図-1 対象地盤の分割

クごとに異なった土質定数をもつ。また、モンテカルロシミュレーションであるため多数回計算を繰り返すが各ケースごとに同一ブロックでも土質定数は異なってくる。

沈下量の計算は次のような仮定で行っている。

- ①一次元圧密方程式を用い、圧密沈下は各平面メッシュごとに独立に沈下するものとして計算する。
- ②沈下量は載荷重ごとの沈下量の重ね合せで求める。
- ③ある荷重に対する最終沈下量の計算はmv または cv を用いる。
- ④各平面メッシュごとに深さ方向に多層地盤となるが、圧密度の計算には換算層厚法を用いる。
- ⑤地中応力の計算には地盤を弾性体としてブシネスクの式を用いる。

以上のような方法によりある大規模な埋立地の沈下のばらつきを計算し、実測沈下量のばらつきとほぼ一致するシミュレーション結果を得ている。

次に、不同沈下を求めたい2点間の距離が短くなると地盤定数の相関性および三次元圧密の影響の2つについて新たに考慮が必要になる。また、不同沈下を求めたい2点間の距離が短くなると各平面メッシュごとに独立して一次元圧密するという仮定も成立しにくくなり、隣接したメッシュ間の相互作用も考慮する必要が出てくる。以上のことから、本プログラムでは地盤の水平方向相関性を導入するとともに、沈下の影響係数とを用いて三次元圧密によるメッシュ間の相互作用の影響を考慮している。

(3)用いた土質定数と沈下量の計算法

本プログラムでは圧密による最終沈下量（各載荷荷重による圧密度 100%での沈下量）の計算に $e - \log p$ 曲線法を用いている。沈下量 s は、圧縮指数 C_c 、圧密係数 c_v 、初期間隙比 e_0 、圧密降伏圧力 p_y 、粘土層厚 H より次式で計算される。

$$s = H \cdot \Delta e / (1 + e_0) \cdot U$$

ここに Δe は間隙比の変化で C_c 、 p_y 、 e_0 によって決まる $e - \log p$ 曲線と載荷荷重によって求められる。また、 U は圧密度であり c_v と層厚、事前の沈下量によって計算される。

計算に用いられる $e - \log p$ 曲線は、圧密圧力 p の常用対数 $\log p$ に関する二次関数として近似されている。近似式は、載荷重 p と圧密降伏圧力 p_y との関係によって次式で表される（ただし $C_c > 0.15$ とする）。

$$\textcircled{1} p = < p^* \text{ のとき} \quad e = e_0 - 0.1 \cdot C_c \log \left(\frac{p}{p_0} \right)$$

$$\textcircled{2} p^* < p < \alpha p_c \quad e = A(\log p)^2 + B(\log p) + C$$

$$\textcircled{3} \alpha p_c = < p \quad e = -C_c \cdot \log p + C - A(\log p_c)^2$$

$$\text{ここに、} A = \frac{C_c(C'_c - C_c)}{2(\log \alpha)(2C_c - C'_c)}$$

$$B = -C_c - 2A \log(\alpha p_c)$$

$$C = e_0 - (0.1 \cdot C_c + B) \cdot \log p^* - A(\log p^*)^2 + 0.1C_c(\log p_0)$$

$$C_c' = 0.1 + 0.25C_c$$

$$p^* = 10^{-\frac{0.1C_c + B}{2A}}$$

$$p_0 = 0.1 \text{ kgf/cm}^2$$

α は実験定数であり、海成粘土については 1.5 を用いる。

3. 入力データの作成

3.1 概要

- (1) 平面図を長方形のピッチのメッシュに割る。
- (2) 計算する範囲をメッシュ内で決める。
- (3) 計算する範囲を平面で矩形になるように指定する。海底面の高さによってグループに分ける。
- (4) グループ分けは、プログラムの内部のワークエリアの必要量を減らす意味もあり、もしワークエリアの不足というエラーが出た時には、グループ分けを変えてみると実行できる時もある。
- (5) メッシュの平面座標は、下記のように左下を 1, 1 とした系とする。グループの位置は、そのグループの矩形の左下隅のメッシュが占める位置で示す。また、大きさはメッシュの横と縦のメッシュ数で示す。

		位置	大きさ
グループ	1	1, 2	2, 4
グループ	2	3, 1	3, 7

- (6) グループ内をさらに地層の違いによる地層の範囲分けをして割りつける。その位置および大きさはグループのデータと同じように考える。
- (7) 深さ方向の座標は水面を 0 とし、下向きに正の方向をとる。
- (8) 地層データを入力するには、海底下の土のデータを何種類かに分類しその土の物理的性質を先に番号をつけて入力する。次に、グループ毎に割りつけた地層群の範囲とその範囲の深さ方向の地層を土の番号と層厚をペアーにして上層から下層に向かって入力する。
- (9) 荷重載荷は、時間毎に載荷範囲（グループ入力と同じ）と埋立てる土の γ 、厚さと位置を入力する。範囲はグループ間にまたがってもよい。また、同じ時間に何ヶ所かの範囲に載荷してよい。時間は載荷開始時間を 0 とし、月を単位として昇順に入力する。
- (11) 地層のための土の γ は、海底下ではその浮力を引いたものを入力する。
- (12) 載荷のための土の γ は、乾燥時のものを入力すれば海底面の下にあるときは 1.0 を引いて計算に使われる。
- (13) 載荷中および載荷終了後の沈下量を計算できる。この時間を沈下計算時間と称し、月を単位に入力する。100 年後の沈下量はいつも出力される (1, 200 月)。この時間の内、任意の時（通常

は載荷終了時) を基準にし、その後の沈下量を計算して出力させることができる。

3.2 入力データごとの説明

(数値は各フィールド内、右よせまたはコンマを用いること)

(1) シミュレーションケースのコントロール

シミュレーションを全ケース一度にやらせるかわりに途中で打ち切り、次に中途からスタートすることができる。

NLOOP, IPRT, NPRV, NCNT, NHANTEI

NLOOP : シミュレーションの全ケース数。(50以下)

IPRT : 印刷モードの指定で通常は0でよい。入力データの確認をするとき、入力エラーをチェックするときは1を入れる。2はデバック用であり、各メッシュに割り付けられた土質定数、各段階ごとの計算結果がすべて印刷される。

NPRV : 前回までのケース数。

NCNT : 今回計算するケース数。

NHANTEI : 不同沈下の判定の有無。0の場合判定しない。1の場合判定する。

(2) データの大きさの定義

NGX, NGY, NGP, NKJ, NTUC, NCHNK, XPT, YPT

NGX : 全体のメッシュ割数 (X方向、50以下)

NGY : // (Y方向、50以下)

メッシュ数の全体は1500以下にする(ただし、ここはディメンション文を変更することで増やすことができる)。

NGP : 荷重載荷グループ数

NKJ : 荷重載荷データ数

NTUC : 土のデータ数 (99以下)

NCHNK : 沈下計算時間数 (20以下) :

XPT : 平面メッシュのピッチ (X方向)

YPT : // (Y方向)

(3) 沈下量の計算の基準となる時間

NTZ, NPT, CLEVEL, XCOR, YCOR

NTZ : 沈下量の基準としたい(ゼロセット)時間数

NPT : 全ケースの沈下量をプリントしたい点の数 (1にする)

CLEVEL : 不同沈下を計算するときの地表面のおおよその高さ。たとえば+5mまで埋立ててその後の不同沈下を求めたいときは5.00とする。また、三次元圧密の影響を無視する場合は3000を入れる。

XCOR : X方向の相関距離

YCOR : Y方向の相関距離

(4) 不同沈下の判定基準

判定しない場合でもカードは必要

KHXMIN, KHYMIN, KDHX, KDHY, RLIM, TLIM

KHXMIN : 不同沈下の判定と評価を行うメッシュの左下位置のX座標

KHYMIN : 不同沈下の判定と評価を行うメッシュの左下位置のY座標

KDHX : 不同沈下の判定と評価を行うメッシュのX方向範囲

KDHY : 不同沈下の判定と評価を行うメッシュのY方向範囲

RLIM : 舗装の耐力によって規定される限界曲率半径

TLIM : 舗装表面の許容勾配偏差

(5) グループの位置および大きさの指示 NGP 数必要

NXM, NYM, NX, NY, NZ

NMX : 全体メッシュの中のグループの左下のメッシュの位置 (X)

NMY : 全体メッシュの中のグループの左下のメッシュの位置 (Y)

NX : グループの大きさ (X方向メッシュ数)

NY : グループの大きさ (Y方向メッシュ数)

NZ : グループの最大地層数

(6) 土質データ

各土質番号について2枚ずつNTUC組必要。

① 1枚目

NTUC(1), NTUC(2), TUCHI(1), TUCHI(2), TUCHI(3), TUCHI(4), TUCHI(5), TUCHI(6), TUCHI(7), TUCHI(8)

② 2枚目

TUCHI(9), TUCHI(10), TUCHI(11), TUCHI(12), TUCHI(13), TUCHI(14), TUCHI(15), TUCHI(16)

NTUC(1) : 土質番号

NTUC(2) : 粘性土と砂質土の区別、粘性土は1砂質土は0を入力

TUCHI(1) : 単位体積重量 γ (t/m^3)

TUCHI(2) : m_v の基準となる圧密圧力 P_o (t/m^2)

Ccを用いて計算するように変更したので入力不要、0を入れる。

TUCHI(3) : 圧密圧力が p_o の時の m_v の平均値 (単位は m^2/ton)

Ccを用いて計算するように変更したので入力不要、0を入れる。

TUCHI(4) : // 標準偏差

Ccを用いて計算するように変更したので入力不要、0を入れる。

TUCHI(5) : 圧密係数の常用対数 $\log c_v$ の平均値、 c_v の単位は m^2/min

TUCHI(6) : // 標準偏差

TUCHI(7) : m_v 傾度の平均値

Ccを用いて計算するように変更したので入力不要、0を入れる。

- TUCHI(8) : // 標準偏差
Ccを用いて計算するように変更したので入力不要、0を入れる。
- TUCHI(9) : 圧縮指数Ccの平均値
- TUCHI(10) : Ccの標準偏差
- TUCHI(11) : 初期間隙比 e_0 の平均値
- TUCHI(12) : e_0 の標準偏差
- TUCHI(13) : 圧密降伏応力 p_c の平均の深度方向増加率 ($\text{kgf/cm}^2/\text{m}$)
- TUCHI(14) : 標高0mの時の p_c の平均値
- TUCHI(15) : p_c の変動係数
- TUCHI(16) : 層厚の変動係数

(7) グループ毎の地層データ (はじめに入力したグループ順にデータを入力する)

なお、地層データの左下および範囲は、今入力しているグループの範囲になければならないことに注意する。

① 1枚目

ZZ

ZZ : グループの標高 (m) (グループ毎に1枚)

② 2枚目

MXL, MYL, MX, MY, MZ

MXL : 地層データの範囲の指定、左下位置のX座標

MYL : // Y座標

MX : X方向の地層数

MY : Y方向の地層数

MZ : 深さ方向の地層数 (注意 : MZ が常に NZ に等しくなければならない。)

③ 3枚目以降

LSNO, HTT, LSNO, HTT, LSNO, HTT, LSNO, HTT

LSNO : 土の番号

HTT : 層厚 (m)

土の番号と層厚を4組ずつペアにして、上から下へMZ個入力する。-1をかくと、1つのグループの地層データの入力が終了する。下部に排水層があるときは必ず砂層を入れること。 (入れないと下部は不透水層になる)

(8) 荷重載荷データ (NKJ 枚)

RT, NKHN(1), NKHN(2), NKHN(3), NKHN(4), RKJU(2), RKJU(3), RKJU(4)

RT : 荷重載荷の時間 (月)

NKHN(1) : 載荷範囲の左下位置のX座標

NKHN(2) : " Y座標

NKHN(3) : 載荷範囲 X方向

NKHN(4) : " Y方向

RKJU(2) : 埋立土の単位体積重量 (ton/m²) (浮力を引かないこと)

RKJU(3) : 埋立層厚 (m)

RKJU(4) : 埋立前の標高 (m) 、この時点での前段階までの標高を入れる。

- ・荷重範囲はグループにまたがっていてもよい。
- ・同じ月に荷重載荷データが2枚以上あってもよい。
- ・通算月はだんだん増えるようにデータを入力すること。

(9) 沈下計算時間 (月) (0.であってはならない。0.001などとする。)

TIM(1), TIM(2), , , , , TIM(NCHNK)

時間は、荷重載荷と同じ時間軸で昇順に並べる。 NCHNK 個必要。

(10) 以後の沈下量の基準になる時間 (月)

(沈下計算時間の数字のどれかと一致していること。 NTZ 個)

TMZ(1), TMZ(2), , , , , TMZ(NTZ)

3.3 入力データの例

例題 空港エプロン舗装構造の解析

メッシュ数 40×5, グループ 1, 荷重載荷データ 4, 沈下計算時間数 8

メッシュ間隔 20m、沈下の基準となる時間数 1

基準標高 2m 相関距離 50m

舗装構造：限界曲率半径 1070m（アスファルト舗装）、許容勾配 0.25%

地盤定数

粘土層厚 20m（4m×5）

層厚の変動係数 0.1

海底地盤の基準標高 -5m、 両面排水

圧縮指数 平均 1.000 標準偏差 0.250

圧密係数の常用対数 平均 -4.96 標準偏差 0.289

初期間隙比 平均 2.345 標準偏差 0.665

圧密降伏圧力 平均 $0.066Z+0.175$ Z (m:深度)

変動係数 0.521

入力データは以下のようなになる。

```
2, 1, 0, 2, 1
40, 5, 1, 4, 2, 8, 20., 20.
1, 1, 2.0, 50., 50.
1, 1, 40, 5, 1070.0, 0.25
1, 1, 40, 5, 6
1, 1, 0.554, 10., 0, 0, -4.96, 0.289, 0.911, 0.18
1.000, 0.250, 2.345, 0.665, 0.066, 0.175, 0.521
2, 0, 0.8, 1,
5.0
1, 1, 40, 5, 6
1, 4.0, 1, 4.0, 1, 4.0, 1, 4.0,
1, 4.0, 2, 1.0
-1
0., 1, 1, 40, 5, 1.7, 2.0, 5.0
4., 1, 1, 40, 5, 1.7, 2.0, 3.0
8., 1, 1, 40, 5, 1.7, 2.0, 1.0
12., 1, 1, 40, 5, 1.7, 2.0, -1.0
24., 36., 48., 60., 84., 120., 240., 360.
24.
```

4. プログラムコード

```
C
C   MAIN PROGRAM
C
C   IMPLICIT REAL*8 (A-H,O-Z)
C   DIMENSION RDYNM(500000),NXYG(8,20),COV(1000,1000),CENT(3000,2)
C   COMMON /PARM/ IFL2, TMCO, JMAX, JJMIN
C---
C   COMMON /PRINT/ IPRT
C
C   DATA  JRS,JWK,JPP,JDL, JCV,JTV,JUU,JHD,JSL,JCH,JDM,JCM,JCS,JPR,
2        JDP, JMX, JCW, JLP, JMS, JZW / 20*0/
C   DATA JXM, JYM, JXS, JYS, JWS, JWZ, JPN, JLS5/8*0/
C
C       IFL2      SEQUENTIAL FILE      I/O NO.
C       MXARY     DYNAMIC ADDRESSING ARRAY SIZE
C
C   NFUTO=0
C   IFL2= 2
C   MXARY=500000
C   TMCO = 24.*60.*30.
C   KHMIN=1
C   KHYMIN=1
C
C   READ(5,1) NLOOP, IPRT, NPRV, NLCNT, NHANTEI
C   READ(5,1) NGX, NGY, NGP, NKJ, NTUCHI, NCHNK, XPT, YPT
C   READ(5,5) NTZ, NPT,CLEVEL,XCOR,YCOR
C
C   KHXMAX=NGX
C   KHYMAX=NGY
C   IF(NHANTEI.EQ.1) THEN
C       READ(5,6) KHMIN,KHYMIN,KHDX,KHDY,RLIM,TLIM
C       KHXMAX=KHMIN+KHDX-1
C       KHYMAX=KHYMIN+KHDY-1
C   END IF
C
C   WRITE(6,3)
C   WRITE(6,2) NLOOP, IPRT, NPRV, NLCNT
C   WRITE(6,2) NGX, NGY, NGP, NKJ, NTUCHI,NCHNK, XPT, YPT
C   WRITE(6,5) NTZ, NPT,CLEVEL,XCOR,YCOR
C   IF (NHANTEI.EQ.1) THEN
C       WRITE(6,6) KHMIN,KHYMIN,KHDX,KHDY,RLIM,TLIM
C   END IF
C   IF (XCOR*YCOR.LT.0.00001) NFUTO=1
C
C   IF( NGP.GT.50 ) GO TO 940
C
C   DO 100 I= 1, NGP
C       READ(5,1)(NXYG(J,I),J=1,5)
C       WRITE(6,2) (NXYG(J,I),J=1,5)
100 CONTINUE
C
C   CALCULATE ARRAY POINTER
C
C   NCHNP = NCHNK + 1
```



```

JLS3 = JCH + KXYS*NCHNP
IF( KLST .GT. JLS3 ) GO TO 120
KRTN = 3
KLST = JLS3
C
C
120 JMX = JRK
JCM = JMX + NCHNP*NLOOP
JCS = JCM + KXYS*NCHNP
JPR = JCS + KXYS*NCHNP
JCW = JPR + 122
JLP = JCW + KXYS
JMS = JLP + NPT*NCHNP*NLOOP
JZW = JMS + 2*NCHNP
JZW2= JZW + KXYS
JZW3= JZW2+ NCHNP*NLOOP
JZW4= JZW3+ NCHNP*NLOOP
JZW5= JZW4+ NCHNP*NLOOP
JZW6= JZW5+ NCHNP*NLOOP
JZW7= JZW6+ NCHNP*NLOOP
JZW8= JZW7+ NCHNP*NLOOP
JZW9= JZW8+ NCHNP*NLOOP
JZW10=JZW9+ NCHNP*NLOOP
JLS4= JZW10+ NCHNP*NLOOP
IF( KLST .GT. JLS4 ) GO TO 125
KRTN = 4
KLST = JLS4
C
125 CONTINUE
NTIMSW = 0
C
WRITE(6,1007) KLST, MXARY
IF( NPRV.GE.NLOOP ) GO TO 200
C
C
C MAIN CALCULATION SUB
C *****
C
IF( NFUTO.EQ.1) GO TO 150
DO 20 IX=1,NGX
DO 25 IY=1,NGY
IR=(IX-1)*NGY+IY
CENT(IR,1)=(IX-1)*XPT+XPT/2
CENT(IR,2)=(IY-1)*YPT+YPT/2
25 CONTINUE
20 CONTINUE
C
DO 30 IR=1,NGX*NGY
DO 40 JR=1,NGX*NGY
DX=CENT(IR,1)-CENT(JR,1)
DY=CENT(IR,2)-CENT(JR,2)
DST=SQRT(DX*DX+DY*DY)/SQRT(XCOR*XCOR+YCOR*YCOR)
COV(IR,JR)=DEXP(-DST)
40 CONTINUE
30 CONTINUE
C
N=NGX*NGY

```

```

CALL LU(COV,N)
C
C
150 CALL SBMAIN( RDYNM(JRK), RDYNM(JNK), NKJ, RDYNM(JNT), RDYNM(JTU),
2         NTUCHI, RDYNM(JME), KXYS, RDYNM(JDR), RDYNM(JDLF),
3         NGP, RDYNM(JTI), NCHNK, RDYNM(JNS), RDYNM(JHT),
4         RDYNM(JDL), RDYNM(JPP), NLOOP, NXYG, NGX, NGY,
5         RK, SK, RDYNM(JCV), RDYNM(JTV), RDYNM(JUU),
6         RDYNM(JHD), RDYNM(JSI), NZMX, RDYNM(JRS), RDYNM(JZK),
7         RDYNM(JCH), XPT, YPT, CLEVEL, COV, NCHNP,
8         RDYNM(JKS), NKJX, NPRV, NLCNT, NTIMSW, JJMAX, NFUTO)
C
IF( NTIMSW .NE. 0 ) STOP
KRTN = 5
C
C PRINT RESULTS SUB
C *****
C
200 CALL SBPOST( RDYNM(JTI), RDYNM(JCM), RDYNM(JCS), NHANTEI,
2 NCHNK, NLOOP, KXYS, NCHNP, NGX, NGY, KHXMIN, KHXMAX, KHYMIN, KHYMAX,
3 XPT, YPT, RLIM, TLIM, RDYNM(JCW), NPT, RDYNM(JTZ), NTZ,
4 RDYNM(JMS), RDYNM(JZW), RDYNM(JZW2), RDYNM(JZW3), RDYNM(JZW4),
5 RDYNM(JZW5), RDYNM(JZW6), RDYNM(JZW7), RDYNM(JZW8), RDYNM(JZW9),
6 RDYNM(JZW10) )
C
C NORMAL END OF JOB
C STOP
C
C WORK AREA OVER FLOW ERROR **
C
940 WRITE(6,1004)
GO TO 1000
950 WRITE(6,1003)
1000 DO 1010 I= 1, 2000
READ(5,1002,END=1020) J
1010 CONTINUE
1020 WRITE(6,1001) KRTN, KLST, MXARY
WRITE(6,1006) NCHNP, NTZ, NPT, NKJ, NTUCHI, KXYS, NGP, KXYZ,
1 NZMX, JJMAX, NKJX, NLOOP, JWK,
2 JTI, JTZ, JRP,
3 JRK, JNK, JKS, JNT, JTU, JME, JZZ, JNS, JHT,
4 JDR, JDLF, JLS, JTT, JTW, JLS1,
5 JRS, JPP, JDL, JCV, JTV, JUU, JHD, JSI, JLS2,
6 JCH, JLS3,
7 JMX, JCM, JCS, JPR, JCW, JLP, JMS, JZW, JLS4,
8 JXM, JYM, JXS, JYS, JWS, JWZ, JPN, JLS5
STOP
C
1 FORMAT(6I5, 2F10.0)
2 FORMAT(6I5, 2F10.2 )
3 FORMAT(35X, '** INPUT DATA ECHO **')
4 FORMAT(2F10.2 )
5 FORMAT(2I5, 3F10.2)
6 FORMAT(4I5, F10.0, F10.3)
1001 FORMAT(' ** ERROR 00 WORK AREA OVER FLOW ROUTINE=', I3,
2 ' AREA NEEDED=', I7, '(MAXARRAY = ', I7, ')')

```

```

1002 FORMAT(A1)
1003 FORMAT('** ERROR 12 (NCHNK+NKJX+6)*NZMX MUST BE LEAST THAN 1000K'
2      )
1004 FORMAT('** ERROR 13   NGP IS GREATER THAN 50' )
1006 FORMAT('NCHNP, NTZ, NPT, NKJ, NTUCHI, KXYS, NGP, KXYZ      ',
2      T52, 8I8 /
3      'NZMX, JJMAX,  NKJX, NLOOP, JWK  ', T52, 5I8 /
4      'JTI, JTZ, JRP      ', T52, 3I8 /
5      'JRK, JNK, JKS, JNT, JTU, JME, JZZ, JNS, JHT  ',
6      T52, 9I8 /
7      'JDR, JDLF, JLS, JTT, JTW, JLS1      ', T52, 5I8 /
8      'JRS, JPP, JDL, JCV, JTV, JUU, JHD, JSI, JLS2  ',
9      T52, 9I8 /
A      ',5X,'JCH, JLS3'      , T60, 2I8 /
B      'JMX, JCM, JCS, JPR, JCW, JLP, JMS, JZW, JLS4', T52, 9I8 /
C      'JXM, JYM, JXS, JYS, JWS, JWZ, JPN, JLS5', T52, 8I8)
1007 FORMAT(' USED ARRAY SIZE = ', I7, '      (MXARY =', I7, ')')
      END

```

```

C
C *****
C      MAIN PROGRAM
C *****
      SUBROUTINE  SBMAIN( RKJU, NKHN, NKJ, NTUC, TUCHI, NTUCHI, MESH,
2      KXYS, DRT, DLT, NGP, TIM, NCHNK, NSO, HTIN,
3      DLT, PP, NLOOP, NXYG, NGX, NGY, RK,
4      SK, CVV, TVV, UU, HDS, SSI, NZMX, RS, ZK,
5      CHINK, XPT, YPT, CLEVEL, COV, NCHNP, NKS, NKJX,
6      NPRV, NLCNT, NTIMSW, JJMAX, NFUTO )

C
C      MAIN SUBROUTINE
C
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER O
      DIMENSION  RKJU(4,NKJ), NKHN(4,NKJ), NTUC(2,NTUCHI), TIM(NCHNP)
      DIMENSION  TUCHI(16,NTUCHI), MESH( KXYS), DRT(KXYS), DLT(KXYS)
      DIMENSION  NSO(NZMX,KXYS), HTIN(NZMX,KXYS), DLT(2,KXYS), PP(KXYS)
      DIMENSION  NXYG( 8,NGP), CVV(NZMX), TVV(NZMX), UU(NZMX), HDS(NZMX)
      DIMENSION  SSI(NZMX,NKJ), RS(JJMAX,NZMX,KXYS), CHINK(KXYS,NCHNP)
      DIMENSION  COV(1000,1000)
      DIMENSION  NKS(7,NKJ), ZK(3000), DLTP(50,3000), HT(50,3000)

C
      COMMON /PARM/IFL2, TMCO, JMAX, JJMIN
      COMMON /PRINT/ IPRT

C
      RS(1) = PV
      RS(2) = MV0
      RS(3) = CV
      RS(4) = K
      RS(5) = SIGMA P(I)
      RS(6) = Z
      RS(10->NCHNK+9) = CHINKA RYOO (CHINKA KEISAN JI)
      RS(NCHNK+10-->) = SAISYUU CHINKA RYOO
      RS(7) = CC
      RS(8) = E(1.0)
      RS(9) = PYI

C
      IRANDX = 999999

```

```

O = 5
C
  REWIND IFL2
  IF( NPRV .EQ.0 ) GO TO 115
100 READ(IFL2,END=110) (CHINK(J,I),J=1,KXYS)
  GO TO 100
110 CONTINUE
C
115 IF( NLCNT .EQ.0 ) NLCNT = NLOOP
  NLPS = NPRV + 1
  NLPE = NLPS + NLCNT - 1
  IF( NLPE .GT. NLOOP ) NLPE = NLOOP
C
  DO 1000 NLP = NLPS, NLPE
C
  DO 128 J0=1,KXYS
    DO 125 J= 1, NZMX
      DO 120 I = 1, JJMAX
        RS(I,J,J0) = 0.
120    CONTINUE
125    CONTINUE
128 CONTINUE
C
  WRITE(6,1) NLP
  IF( IPRT.EQ. 2 ) WRITE(6,10)
C
C  SET VALUE TO MV,CV,K
C
  DO 150 IX=1,NGX
    DO 145 IY=1,NGY
      IR=(IX-1)*NGY+IY
      IF (NFUTO.EQ.0) MZ=NZMX
      IF (NFUTO.EQ.1) MZ=MESH(IR)
      DO 142 IZ=1,MZ
        HT(IZ,IR)=HTIN(IZ,IR)
142    CONTINUE
145    CONTINUE
150 CONTINUE
C
  IF (NFUTO.EQ.0) CALL SOLDAT(NGX,NGY,NTUC,COV,RS,HT,
2    TUCHI,IRANDX,NSO, JJMAX,NZMX,KXYS,NTUCHI )
  IF (NFUTO.EQ.1) CALL SOLDAT2(NGX,NGY,NTUC,RS,HT,MESH,
2    TUCHI,IRANDX,NSO,JJMAX,NZMX,KXYS,NTUCHI)
C
  IF( IPRT.NE.2) GO TO 350
C
  DO 300 IX=1,NGX
    DO 250 IY=1,NGY
      IR=(IX-1)*NGY+IY
      IF (NFUTO.EQ.0) MZ=NZMX
      IF (NFUTO.EQ.1) MZ=MESH(IR)
      DO 200 IZ=1,MZ
        NC=NSO(IZ,IR)
        WRITE(6,2) IX,IY,IZ,RS(7,IZ,IR),RS(3,IZ,IR),RS(9,IZ,IR),
2    RS(8,IZ,IR),RS(1,IZ,IR)*0.1,HT(IZ,IR),RS(6,IZ,IR),
3    NTUC(1,NC),NTUC(2,NC)
200    CONTINUE

```

```

250 CONTINUE
300 CONTINUE
C
350 IF(IPRT.EQ.3) WRITE(6,2)
C
C MAIN CALCULATION START
C
DO 370 I = 1, 3
DO 360 J = 1, KXYS
DLT(I,J) = 0.
360 CONTINUE
370 CONTINUE
C
IF( IPRT.EQ.3 ) WRITE(6,43) (MESH(J),J=1,KXYS)
C
DO 850 NKK = 1, NKJX
KSTRT = NKS(1,NKK)
KEND = NKS(2,NKK)
KRXMI = NKS(4,NKK)
KRYMI = NKS(5,NKK)
KRXXM = NKS(6,NKK)
KRYMX = NKS(7,NKK)
NP = NCHNK+9+NKK
TNOW = RKJU(1,KSTRT)
WRITE(6,3) NKK, NKJX, TNOW
DO 400 IR=1,NGX*NGY
PP(IR)=0.
ZK(IR)=0.
400 CONTINUE
C
C SAIKA KAJYUU KEISAN
C
DO 510 NK = KSTRT, KEND
C
KXMIN = NKHN( 1, NK )
KXMAX = NKHN( 3, NK )
KYMIN = NKHN( 2, NK )
KYMAX = NKHN( 4, NK )
RKGAM = RKJU( 2, NK )
WRKGAM= RKGAM-0.7
HKJ = RKJU( 3, NK )
ZKJ = RKJU( 4, NK )
C
IF(ZKJ.LT.0.) THEN
PT = RKGAM * HKJ
ELSE IF(ZKJ-HKJ.LT.0) THEN
PT = ZKJ *WRKGAM + (HKJ-ZKJ)*RKGAM
ELSE
PT = WRKGAM*HKJ
END IF
C
IF(IPRT.EQ.3) WRITE(6,9)
DO 500 IX = KXMIN, KXMAX
DO 450 IY = KYMIN, KYMAX
C
IR = (IX-1)*NGY+IY
ZK(IR)=ZKJ-HKJ

```



```

DCHN=DLT(2,IR)-DLT(1,IR)
IF(ZKJ.GE.0) THEN
  PP(IR)=PT+WRKGAM*DCHN
  ELSE IF(ZKJ.LT.-DCHN) THEN
    PP(IR)=PT+RKGAM*DCHN
  ELSE
    PP(IR)=PT-RKGAM*ZKJ+WRKGAM*(DCHN+ZKJ)
  END IF
IF (PP(IR).LT.0) THEN
  WRITE(6,4) IX,IY,RKGAM,HKJ,ZKJ,
2    DLT(2,IR),DLT(1,IR),PP(IR), IR
  PP(IR)=0.
  END IF
C
  IF(IPRT.EQ.3) WRITE(6,4) IX,IY,RKGAM,HKJ,ZKJ,
2    DLT(2,IR),DLT(1,IR),PP(IR), IR
  DLT(1,IR) = DLT(2,IR)
C
450  CONTINUE
500  CONTINUE
C
510  CONTINUE
C
  IF( IPRT.EQ.2 ) WRITE(6,42) ((NSO(J,J1),HT(J,J1),
1    J=1,NZMX),J1=1,KXYS)
C
  IXMAX = NGX
  IXMIN = 1
  IYMAX = NGY
  IYMIN = 1
  IF( IXMAX .GT. KRXXMX+O ) IXMAX = KRXXMX+O
  IF( IYMAX .GT. KRYMX+O ) IYMAX = KRYMX+O
  IF( IXMIN .LT. KRXMI-O ) IXMIN = KRXMI-O
  IF( IYMIN .LT. KRYMI-O ) IYMIN = KRYMI-O
C
  IFRYOK=0
515  CONTINUE
C
  Chikaryo no Keisan
C
c
DO 602 IX = IXMIN, IXMAX
DO 601 IY = IYMIN, IYMAX
C
  IR=(IX-1)*NGY+IY
  ZOR = DRT(IR)
  SX = (IX-1)*XPT+XPT*0.5
  SY = (IY-1)*YPT+YPT*0.5
  MZ = MESH(IR)
C
  IF(IPRT.EQ.3) WRITE(6,32) IX,IY, IR
  IF(IPRT.EQ.3) WRITE(6,5)
C
  RS(6,1,IR) =0.
  DO 520 IZ = 2, MZ
    RS(6,IZ,IR) = RS(6,IZ-1,IR) + HT(IZ-1,IR)
520  CONTINUE

```

```

C
DO 600 IZ = 1, MZ
C
DLTP(IZ,IR) = 0.
NC = NSO(IZ,IR)
IF ( NTUC(2,NC).EQ.0 ) GO TO 600
IF ( RS(7,IZ,IR).LT.0.15 ) GO TO 600
C
SZ =ZOR+DLTF(IR)+RS(6,IZ,IR)+HT(IZ,IR)*0.5
C
JXMAX = IX+O
JYMAX = IY+O
JXMIN = IX-O
JYMIN = IY-O
C
IF( JXMAX .GT. NGX) JXMAX = NGX
IF( JYMAX .GT. NGY) JYMAX =NGY
IF( JXMIN .LT. 1)      JXMIN = 1
IF( JYMIN .LT. 1)      JYMIN = 1
C
DO 560 KX = JXMIN, JXMAX
DO 540 KY = JYMIN, JYMAX
JR = (KX-1)*NGY+KY
XKJ = (KX-1)*XPT+XPT*0.5
YKJ = (KY-1)*YPT+YPT*0.5
DLTP(IZ,IR)=DLTP(IZ,IR)+BUSINQ(SX,SY,SZ,XKJ,
1      YKJ,ZKJ,XPT,YPT,PP(JR))
540 CONTINUE
560 CONTINUE
C
HH=HT(IZ,IR)
P1=(RS(1,IZ,IR)+RS(5,IZ,IR))*0.1
P2=P1+(DLTP(IZ,IR)*0.1)
C
CALL ELGP( RS(7,IZ,IR),RS(9,IZ,IR), RS(8,IZ,IR) ,
1      P1, P2, VOID1,VOID2,IR,IZ )
C
DVOID=VOID1-VOID2
IF (VOID1.GT.0.) THEN
VOID0=VOID1
ELSE
VOID0=0.
END IF
RS(NP,IZ,IR)=HH*DVOID/(1.0+VOID0)
C
IF(IPRT.EQ.3) WRITE(6,6) IZ, RS(8,IZ,IR),
2      RS(9,IZ,IR),RS(7,IZ,IR), P2, RS(5,IZ,IR)*.1,
3      DLTP(IZ,IR)*.1,SZ,ZKJ,HH, RS(NP,IZ,IR)
C
600 CONTINUE
601 CONTINUE
602 CONTINUE
C
Fryoku no Hosei
C
IF(IFRYOK.EQ.1) THEN

```

```

        IFRYOK=0
        GO TO 610
    END IF
C
    DO 605 IX=KRXMI,KRXMX
        DO 607 IY=KRYMI,KRYMX
            IR=(IX-1)*NGY+IY
            MZ=MESH(IR)
            IFC=0
            SETLF=0.
            DO 609 IZ=1,MZ
                DO 608 I=1,NKK
                    IP= NCHNK+9+I
                    SETLF=SETLF+RS(IP,IZ,IR)
608             CONTINUE
609             CONTINUE
                SETLF = SETLF - DLT(1,IR)
                IF(ZK(IR),LT.0) THEN
                    DLPCHNK=DMIN1(-ZK(IR),SETLF)*1.0
                    PPN= PP(IR)-DLPCHNK
                    IF (PPN.LT.0) PPN=0.
                    PP(IR)=DSQRT(PP(IR)*PPN)
                    IFC=IFC+1
                END IF
607             CONTINUE
605             CONTINUE
            IF(IFC.GT.0) THEN
                IF(IPRT.GT.0) THEN
                    WRITE(6,52)
                END IF
                IFRYOK=1
                GO TO 515
            END IF
C
610             CONTINUE
C
C             Jikan-Chinka Kankei
C
        DO 750 IX=IXMIN,IXMAX
            DO 700 IY=IYMIN,IYMAX
C
                IR=(IX-1)*NGY+IY
                MZ=MESH(IR)
                DLTF(IR)=0.
                DO 618 IZ=1,MZ
                    DL=0.
                    DO 617 I= 1, NKK
                        IP = NCHNK+9+I
                        SSI(IZ,I) = RS(IP,IZ,IR)
                        DL=DL+SSI(IZ,I)
617                 CONTINUE
                    CVV(IZ)=RS(3,IZ,IR)
                    HT(IZ,IR)=HT(IZ,IR) - 0.5*RS(NP,IZ,IR)
                    RS(5,IZ,IR) = RS(5,IZ,IR) + DLTP(IZ,IR)
                    DLTF(IR) = DLTF(IR) + DL
618                 CONTINUE
C

```

```

IF( NKK .EQ. NKJX ) GO TO 650
NKP = KEND + 1
TIMN = RKJU(1,NKP)
DLT(2,IR) = 0.
DO 640 I = 1, NKK
    TT = TIMN -RKJU(1,NKS(1,I))
    CALL CALCU (IX,IY,NTUC,NSO(1,IR),HT(1,IR),CVV,TVV,UU,
1         HDS,MZ,TT,NZMX,NTUCHI)
C
    IF( IPRT.EQ.4) WRITE(6,36) IX,IY,TT, (UU(J),J=1,MZ)
C
    DL = 0.
    DO 620 IZ = 1, MZ
        IF( NTUC(2,NSO(IZ,IR)).EQ.0 ) GO TO 620
        DL = DL+SSI(IZ,I)*UU(IZ)
620    CONTINUE
        DLT(2,IR) = DLT(2,IR) + DL
640    CONTINUE
C
650    CONTINUE
    DO 670 NN = 1, NCHNK
        TT = TIM(NN)-TNOW
        IF(TT.LT.0.001) GO TO 670
        CALL CALCU( IX,IY,NTUC,NSO(1,IR),HT(1,IR),CVV,TVV,UU,
$         HDS,MZ,TT,NZMX,NTUCHI)
C
        DO 660 IZ =1, MZ
            IF( NTUC(2,NSO(IZ,IR)).EQ.0 ) GO TO 660
            RS(NN+9,IZ,IR) = RS(NN+9,IZ,IR) + UU(IZ)*RS(NP,IZ,IR)
660        CONTINUE
670        CONTINUE
C
        -----
        DO 680 IZ=1,MZ
            HT(IZ,IR)=HT(IZ,IR) - 0.5*RS(NP,IZ,IR)
            IF( IPRT.EQ.4) THEN
                WRITE(6,33) IZ
                WRITE(6,35) (RS(J,IZ,IR),J=10,JMAX)
                WRITE(6,30) (RS(J,IZ,IR),J=JMIN,JJMAX)
            END IF
680        CONTINUE
700    CONTINUE
750    CONTINUE
C
850 CONTINUE
C
C     Data Seiri
C
    IF( IPRT.EQ.5 ) WRITE(6,7)
C
    DO 870 I = 1, NCHNP
        DO 860 J = 1, KXYS
            CHINK(J,I) = 0.
860    CONTINUE
870 CONTINUE
C
C
    DO 960 IX = 1,NGX

```

```

DO 940 IY = 1,NGY
C
  IR = (IX-1)*NGY+IY
  IF( IPRT.EQ.5 ) WRITE(6,38) IX,IY, IR
C
  KIX=IX
  KIY=IY
C
  DO 930 JX= MAX0(KIX-2,1),MIN0(KIX+2,NGX)
    DO 935 JY=MAX0(KIY-2,1),MIN0(KIY+2,NGY)
      JR = (JX-1)*NGY+JY
      MZ=MESH(JR)
C
      DO 920 JZ = 1, MZ
        IF( NTUC(2,NSO(JZ,JR)).EQ.0) GO TO 920
        IF( NTUC(2,NSO(JZ,JR)).EQ.3) GO TO 920
C
        DEPTH=RS(6,JZ,JR)+CLEVEL+DRT(JR)+DLTF(JR)*0.5
        DSTX=XPT*(IX-JX)
        DSTY=YPT*(IY-JY)
        DST=DSQRT(DSTX*DSTX+DSTY*DSTY)
        BCHINK=DSQRT(XPT*YPT/3.1416)*2.0
C
        IF(IPRT.EQ.5) WRITE(6,44) JX,JY,JZ
C
        DO 900 I = 1, NCHNK
          CHINK(IR,I) = CHINK(IR,I) + RS(I+9,JZ,JR)*DBFUNC(
2          DEPTH,BCHINK,DST)
900    CONTINUE
        IF(IPRT.EQ.5) WRITE(6,37) (RS(J,JZ,JR),J=10,JMAX)
C
        DO 910 I= 1, NKJX
          CHINK(IR,NCHNP) = CHINK(IR,NCHNP)+RS(I+NCHNK+9,JZ,JR)
2          *DBFUNC(DEPTH,BCHINK,DST)
910    CONTINUE
        IF(IPRT.EQ.5) WRITE(6,39) (RS(J,JZ,JR),J=JJMIN,JJMAX)
C
920    CONTINUE
C
935    CONTINUE
930    CONTINUE
C
      IF( IPRT.EQ.6) WRITE(6,40) IX,IY,IR,(CHINK(IR,I),I=1,NCHNP)
940 CONTINUE
960 CONTINUE
C
C
  DO 995 I= 1, NCHNP
    WRITE(IFL2) (CHINK(J,I),J=1,KXYS)
995 CONTINUE
C
1000 CONTINUE
C
  ENDFILE IFL2
C
  IF( NLPE .GE. NLOOP ) RETURN
  WRITE(6,8) NLPE

```

```

      NTIMSW = 1
C
C
      RETURN
C
C   * FORMAT *
1  FORMAT('1',          'CASE=', I5 ,5X,100('='))
2  FORMAT(' ',3I3,F7.3,1X,E10.4,3F7.3,2F8.3,I5,('(,I3,')')
3  FORMAT('  '0','NK =', I5 ,'/',I5,5X,60('='),'  TIME =',F10.0  )
4  FORMAT(' ', 2I3,6F8.3,3X,('(,I4,')')
5  FORMAT('/  IZ',5X,'E0',5X,'PC',5X,'CC',3X,'Pnow',
2      2X,' SIGMP', 2X,'DLP',5X,'Z', 5X,'ZKJ',5X,'H  ',
3      4X,'SI'/)
6  FORMAT(' ', I3,F7.2, F7.2, F7.2, F7.2, F7.2,
2      F7.2, F7.2, F7.2, F7.2, F7.2 )
7  FORMAT('1')
8  FORMAT(/'***** ',I5,' CASE END *****/)
9  FORMAT(/'  KX KY',3X,'GAMMA ',5X,'H  ',5X,'Z  ', 2X,'DLT(I)'
3      ,1X,'DLT(I-1)',3X,'PK' / )
10 FORMAT('  IX IY IZ',4X,'CC',6X,'CV',7X,'PC',5X,'E0',5X,'P0',6X,'H',
2      7X,'Z',5X,' No.' /)
30 FORMAT(' RS(RS(U)) =',15F8.3/,12X,15F8.3//)
32 FORMAT('/0', 20('-),'  IX, IY', 2I5, 5X,20('-),'  IR, KR',2I6)
33 FORMAT(' IZ =',I4)
34 FORMAT(' ', 25X,F10.0,3G15.5)
35 FORMAT(' RS(S) =',15F8.3/,12X,15F8.3)
36 FORMAT(' IX=',I4,' IY=',I4,'  TIME =', F20.0,' *UU =',10F8.4/,
2  41X,10F10.4  )
37 FORMAT('  RS(10-JMAX  ),8F9.3/5(' ', 6X,8F9.3//)
38 FORMAT(' IX=',I4,' IY=',I4,' IZ=',I4)
39 FORMAT('  RS(JJMIN-JJMAX),8F9.3/5(' ', 6X,8F9.3//)
40 FORMAT( 3I4,8F8.3/5(' ', 6X,8F8.3//)
41 FORMAT(' ', 35X,'SI =', 20F10.4)
42 FORMAT('ONSO,HT  ', 8(I5,F10.2)/30(9X, 8(I5, F10.2)))
43 FORMAT('OMESH  ',25I5/30(7X,25I5) )
44 FORMAT('  JX=',I4,' JY=',I4,' JZ=',I4)
45 FORMAT(2F10.5)
50 FORMAT(3I5,3F15.6)
51 FORMAT(8F10.5)
52 format(/'***** FRYOKU HOSEI *****/)
53 FORMAT(F8.4,2I5,F8.4)
55 FORMAT(I5,6F8.4)
56 FORMAT(8F10.5)
      END
C
C *****
C   DBFUNC
C *****
      FUNCTION DBFUNC(D,B,DST)
      IMPLICIT REAL*8(A-H,O-Z)
C
      IF (D.GT.1000.) GO TO 200
C
C
      ADST=DABS(DST)
      DB=D/B
C

```

```

      IF (DB.LT.3.0) THEN
        XI0=0.75/(DB+0.65)-0.16
        XI1=0.145-0.094/(DB+0.65)-0.026*DB
        XI2=0.013*DB
        GO TO 50
      END IF
C
      XI0=0.05
      XI1=0.043
      XI2=0.039
C
50 CONTINUE
C
      IF (ADST.LT.0.5*B) THEN
        DBFUNC=XI0
        GO TO 100
      END IF
C
      IF (ADST.LT.1.7*B) THEN
        DBFUNC=XI1
        GO TO 100
      END IF
C
      DBFUNC=XI2
C
100 CONTINUE
      IF (DBFUNC.LT.0) DBFUNC=0.
      RETURN
C
200 CONTINUE
      IF (DST.LT.0.1) THEN
        DBFUNC=1.0
      ELSE
        DBFUNC=0.
      END IF
      RETURN
C
C
1 FORMAT(8F10.5)
      END
C *****
C   ELGP ----- e-log P Curve
C *****
      SUBROUTINE ELGP( CC,PY, E0, P1, P2, VOID1,VOID2,IR,IZ )
      IMPLICIT REAL*8 (A-H,O-Z)
C       E0 = void ratio at p = 0.1 kg/cm2
C       CC must be greater than 0.1/0.75
      P00=0.1
      CC1=CC* 0.25 +0.1
      CC2 = CC *0.1
      ALP =1.5
      IF (PY.LT.0.1) PY=0.1
      A = CC* (CC1-CC)/ (4.0*CC -2.0*CC1)/ DLOG10(ALP)
      B = -CC- 2.0*A*DLOG10(ALP*PY)
      PX = 10.**(-(CC2+ B)/(2.0*A))
      PYX = PY*ALP
C

```

```

IF (PX.GE.P00) THEN
  C= E0+(CC2+B)*(CC2+B)/(4.0*A)+CC2*DLOG10(P00)
  GO TO 100
END IF
IF (PYX.LT.P00) THEN
  C=E0 +A*DLOG10(PYX)*DLOG10(PYX)+CC2*DLOG10(P00)
  GO TO 100
END IF
C = E0+DLOG10(P00)
100 CONTINUE
C
  IF (P1.LE.0.) GO TO 250
  IF (P2.LE.0.) GO TO 250
  VOID1 =EP(P1,A,B,C,PX,PYX,CC)
  VOID2 =EP(P2,A,B,C,PX,PYX,CC)
C
  RETURN
250 WRITE(6,2)IR,IZ,P1,P2
  STOP
  1 FORMAT(8F9.3)
  2 FORMAT(' ***** ELGP ERROR *****',2I4,2F8.4)
  END
C
C *****
C EP e-logp
C *****
C
  FUNCTION EP(PRES,A,B,C, PX,PYX,CC)
  IMPLICIT REAL*8 (A-H,O-Z)
  HOUBTSUSEN NO KINJI
C
  AGP= DLOG10(PRES)
  IF (PRES. LT. PX) THEN
    EP= -0.1*CC*AGP + C-(B+0.1*CC)*(B+0.1*CC)/(4.0*A)
    GO TO 50
  END IF
  IF (PRES. GT.PYX) THEN
    EP= -CC*AGP + C- A*DLOG10(PYX)*DLOG10(PYX)
    GO TO 50
  END IF
  EP= A*AGP*AGP +B*AGP +C
C
  WRITE(6,1) A,B,C,AGP,EP
50 CONTINUE
  RETURN
  1 FORMAT(' A,B,C=',3F10.5,'AGP,EP=',2F10.5)
  END
C
C *****
C SBPOST ---- DATA SEIRI
C *****
  SUBROUTINE SBPOST( TIM, CHINKM, CHINKS,NHANTEI,
  2 NCHNK, NLOOP, KXYS,NCHNP,NGX,NGY,KHXMIN,KHXMAX,KHYMIN,KHYMAX,
  3 XPT,YPT,RLIM,TLIM,CHINWK,NPT, ITMZ, NTZ,DLMS,
  4 ZEWK,FMEAN,FSD,DFMEAN,DFSD,DFMAX,TLTMAX,RMIN,RRATE,TRATE )
C
  IMPLICIT REAL*8 (A-H,O-Z)
  DIMENSION TIM(NCHNP), CHINKM(KXYS,NCHNP), CHINKS(KXYS,NCHNP),

```



```

2 CHINWK(KXYS),ITMZ(NTZ),DLMS(2,NCHNP),ZEWK(KXYS),
3 DFMEAN(NLOOP,NCHNP),DFSD(NLOOP,NCHNP),DFMAX(NLOOP,NCHNP),
4 TLTMAX(NLOOP,NCHNP),RMIN(NLOOP,NCHNP),RRATE(NLOOP,NCHNP),
5 TRATE(NLOOP,NCHNP),FMEAN(NLOOP,NCHNP),FSD(NLOOP,NCHNP)
C
  DIMENSION  MOJ(2),CHDD(3000)
  DIMENSION  DFMN(2,50),DFD(2,50),DFM(2,50),TLTM(2,50),RM(2,50),
2    RR(2,50),TR(2,50),FMN(2,50),FSDMN(2,50)
C
  COMMON /PARM/ IFL2, TMCO, JMAX, JJMIN
  COMMON /PRINT/ IPRT
C
  DATA MOJ/'M', 'S' /
  DATA  MBL/' '  '/'
C
  DO 90 I= 1, NCHNK
    TIM(I) = TIM(I)/TMCO+0.01
90 CONTINUE
  TIM(NCHNP) = 1200.
C
C
  DO 1000 IWP = 1, NTZ
C
  WRITE(8,6) IWP
  REWIND IFL2
C
  DO 110 I= 1, NCHNP
    DO 100 J= 1, KXYS
      CHINKS(J,I) = 0.
      CHINKM(J,I) = 0.
100 CONTINUE
110 CONTINUE
C
  FLOP = FLOAT(NLOOP)
  FLOPM = FLOP-1.
  IT = ITMZ(IWP)
C
C  RESIDUAL SETTLEMENT
c
  DO 170 NLP = 1, NLOOP
C
  DO 120 J= 1, KXYS
    ZEWK(J) = 0.
120 CONTINUE
  WRITE(8,7) NLP
C
  DO 160 I= 1, NCHNP
    READ(IFL2) (CHINWK(J),J= 1, KXYS)
    DMAX = 0.
    DMIN = 10000.
    DO 130 J= 1, KXYS
      IF ( I .EQ. NCHNP ) ZEWK(J) = 0.
      DD = CHINWK(J)- ZEWK(J)
      IF( DMAX .LT. DD ) DMAX = DD
      IF( DMIN .GT. DD ) DMIN = DD
      CHINKM(J,I) = CHINKM(J,I) + DD
      CHDD(J)=DD

```

```

130     CONTINUE
      IF (IT .EQ. I) THEN
          DO 140 J= 1, KXYS
              ZEWK(J) = CHINWK(J)
140     CONTINUE
      END IF
C
C     PRINT ---- LIST
C
      WRITE(8,8) TIM(I)
      DO 150 IX=KHXMIN,KHXMAX
          IX0=NGY*(IX-1)
          WRITE(8,9) (CHDD(J1)*100,J1=IX0+KHYMIN,IX0+KHYMAX)
150     CONTINUE
C
      ITIMK=I
      INL=NLP
      IF (NHANTELEQ.1) CALL DIFPRT( CHDD,KXYS,KHXMIN,KHXMAX,
2         KHYMIN,KHYMAX,XPT,YPT,NGY,RLIM,TLIM,FMEAN(INL,ITIMK),
3         FSD(INL,ITIMK),DFMEAN(INL,ITIMK),DFSD(INL,ITIMK),
4         DFMAX(INL,ITIMK),TLTMAX(INL,ITIMK),RMIN(INL,ITIMK),
5         RRATE(INL,ITIMK),TRATE(INL,ITIMK) )
C
160     CONTINUE
170     CONTINUE
      IF (NHANTELEQ.0) GOTO 190
C
C
      WRITE(8,10)
      DO 180 I=1,NCHNP
          CALL MESD(FMEAN(1,I),NLOOP,FMN(1,I),FMN(2,I))
          CALL MESD(FSD(1,I),NLOOP,FSDMN(1,I),FSDMN(2,I))
          CALL MESD(DFMEAN(1,I),NLOOP,DFMN(1,I),DFMN(2,I))
          CALL MESD(DFSD(1,I),NLOOP,DFD(1,I),DFD(2,I))
          CALL MESD(DFMAX(1,I),NLOOP,DFM(1,I),DFM(2,I))
          CALL MESD(TLTMAX(1,I),NLOOP,TLTM(1,I),TLTM(2,I))
          CALL MESD(RMIN(1,I),NLOOP,RM(1,I),RM(2,I))
          CALL MESD(RRATE(1,I),NLOOP,RR(1,I),RR(2,I))
          CALL MESD(TRATE(1,I),NLOOP,TR(1,I),TR(2,I))
          WRITE(8,11) TIM(I),TIM(IT),FMN(1,I),FSDMN(1,I),
1             DFMN(1,I),DFMN(1,I)/FMN(1,I),DFD(1,I),DFM(1,I),
2             DFM(1,I)/FMN(1,I), DFM(2,I),TLTM(1,I),TLTM(2,I),
3             RM(1,I),RM(2,I),RR(1,I),RR(2,I),TR(1,I),TR(2,I)
180     CONTINUE
C
190     CONTINUE
C
C     MEAN
C
      DO 210 I= 1, NCHNP
          DO 200 J= 1, KXYS
              CHINKM(J,I) = CHINKM(J,I)/FLOP
200     CONTINUE
210     CONTINUE
C
      IF( NLOOP.LE.1 ) GO TO 380
C

```

```

        REWIND IFL2
C
c Standard Deviation
C
        DO 270 NLP = 1, NLOOP
C
        DO 220 J= 1, KXYS
            ZEWK(J) = 0.
220    CONTINUE
        DO 260 I= 1, NCHNP
            READ(IFL2) (CHINWK(J),J=1, KXYS)
            DO 250 J= 1, KXYS
                IF( I .EQ. NCHNP ) ZEWK(J) = 0.
                CHINKS(J,I) = CHINKS(J,I)+ (CHINWK(J)-CHINKM(J,I)
1                    -ZEWK(J))**2
250    CONTINUE
            IF( IT .EQ. I ) THEN
                DO 255 J= 1, KXYS
                    ZEWK(J) = CHINWK(J)
255    CONTINUE
            END IF
260    CONTINUE
270    CONTINUE
C
        DO 310 I= 1, NCHNP
            DO 300 J= 1, KXYS
                CHINKS(J,I) = DSQRT( CHINKS(J,I)/FLOPM)
300    CONTINUE
310    CONTINUE
C
c PRINT ---- MEAN AND STANDARD DEVIATION
C
380    CONTINUE
C
        DO 400 I= 1, NCHNP
            DLMS(1,I) = 0.
            DLMS(2,I) = 0.
            CALL PRSUB(CHINKM(1,I),KXYS,NGX,NGY,TIM(I),TIM(IT))
            IF( NLOOP.LE.1 ) GO TO 400
            CALL PRSUB(CHINKS(1,I),KXYS,NGX,NGY,TIM(I),TIM(IT))
400    CONTINUE
C
1000 CONTINUE
C
        RETURN
C
1 FORMAT('1', 20X,'ROH RANGE'/ T 97,'KIJUN MONTH=',F6.0/
2          'CASE MONTH=',F8.0,17F6.0)
2 FORMAT( 15, 4X, 122A1 )
3 FORMAT( 4X,A1, 4X, 122A1 )
4 FORMAT('1',20X,'SPECIFIED POINT SETTLEMENT   POINT NO.=',15/
1          ', T97,'KIJUN MONTH=',F6.0/
2          'CASE MONTH=', F8.0, 17F6.0)
5 FORMAT( 65(A1,'-') )
6 FORMAT(' * List of Dif.Settlment *      Kijun Zikan No.=',I4)
7 FORMAT(/,' Case Number =',I4)
8 FORMAT(/,' Time =',F10.1)

```

```

9 FORMAT('          ',20F6.1)
10 FORMAT(/,' Time ',K.Tim',   M.S.', M.SD.', M.DS.', R.DS ',
2 ' SD.DS', M.DSmx', R.DSMX', D.DSmax', M.TLmax', D.TLmax',
3 ' M.Rmin ', D.Rmin ', M.Rrate', D.Rrate',
4 ' M.Trate', D.Trate')
11 FORMAT(F6.0,F5.0, F7.2,2F6.2,F6.3,F6.2,F7.2,F7.3,F7.2,
2      2F8.3,2F8.0,2F8.2,2F8.2)
      END
C
C *****
C   MESD
C *****
      SUBROUTINE MESD(A,N,MEAN,SD)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(3000)
      REAL*8 MEAN
C
      SG=0.
      SG2=0.
      DO 10 I=1,N
C        WRITE(6,1) I,A(I),SG,SG2
          SG=SG+A(I)
          SG2=SG2+A(I)*A(I)
10 CONTINUE
      MEAN=SG/N
      SD=SQRT( (SG2-SG*SG/N)/(N-1) )
1  FORMAT(I3,3F20.3)
      RETURN
      END
C
C *****
C   PRTSUB  ---- PRINT
C *****
      SUBROUTINE PRTSUB (CHINK,KXYS,NGX,NGY,TIM,TIMZ)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION CHINK(KXYS)
      WRITE(16,1) TIM,TIMZ
      DO 10 IX=1,NGX
          IX0=(IX-1)*NGY
          WRITE(16,2) (CHINK(I)*100.,I=IX0+1,IX0+NGY)
10 CONTINUE
      RETURN
1  FORMAT(/,2F10.0)
2  FORMAT(12F6.1)
      END
C
C *****
C   DIFPRT   Differential Settlements
C *****
      SUBROUTINE DIFPRT( CHDD,KXYS,KHXMIN,KHXMAX,KHYMIN,KHYMAX,
2  XPT,YPT,NGY,RLIM,TLIM,FMEAN,FSD,DFMEAN,DFSD,DFM,TLTM,R1M,
3  RRATE,TRATE )
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION CHDD(3000),A(300)
C

```

```

        NTOTAL=0
        NRLIM=0
        NTLIM=0
C
C   Y-LINE
C
        DFM=0.
        TLTM=0.
        R1M=1000000.
        R2M=1000000.
C
        IF (IPRT.GT.2) WRITE(8,1)
        DFMEAN=0.
        DF2=0.
        NDATA=0
        FMEAN=0.
        F2=0.
        NALL=0
C
        DO 100 I=KHYMIN,KHYMAX
            DO 105 J=KHXMIN,KHXMAX
                A(J)=CHDD(I+NGY*(J-1))
                FMEAN=FMEAN+A(J)
                F2=F2+A(J)*A(J)
105    CONTINUE
C
        DFMAX=0.
        DO 107 J=KHXMIN,KHXMAX-1
            DF=DABS(A(J+1)-A(J))
            NTOTAL=NTOTAL+1
            IF (DF.LT.0.0001) THEN
                R1=10000000.
            ELSE
                R1=XPT*XPT/(6.*DF)
            END IF
            TLT=DF/XPT*100.
            IF (R1.LT.RLIM) NRLIM=NRLIM+1
            IF (TLT.GT.TLIM) NTLIM=NTLIM+1
            IF (DF.GT.DFMAX) DFMAX=DF
            NDATA=NDATA+1
            DFMEAN=DFMEAN+DF
            DF2=DF2+DF*DF
107    CONTINUE
C
        DFMAX =DFMAX*100.
        TLTMAX=DFMAX/XPT
        IF (DFMAX.GT.0) R2MIN=XPT*XPT/(2.*DFMAX*0.01)
        R1MIN=R2MIN/3.
        IF (DFMAX.GT.DFM) DFM=DFMAX
        IF (TLTMAX.GT.TLTM) TLTM=TLTMAX
        IF (R1MIN.LT.R1M) R1M=R1MIN
        IF (R2MIN.LT.R2M) R2M=R2MIN
C
        IF (IPRT.GT.2)  WRITE(8,2) I,DFMAX,TLTMAX,R1MIN,R2MIN
C
100 CONTINUE
C

```

```

NALL=(KHYMAX-KHYMIN+1)*(KHXMAX-KHXMIN+1)
FSD=SQRT( (F2-FMEAN*FMEAN/NALL)/(NALL-1) )
FMEAN=FMEAN/NALL
FMEAN=FMEAN*100.
FSD=FSD*100.
WRITE(8,8) FMEAN,FSD
C
DFMY=DFM
TLTMY=TLTM
R1MY=R1M
R2MY=R2M
DFSD=SQRT( (DF2-DFMEAN*DFMEAN/NDATA)/(NDATA-1) )
DFSD=DFSD*100.
DFMEANY=DFMEAN/NDATA*100.
WRITE(8,6) DFMEANY,DFSD
WRITE(8,4) DFM,TLTM
WRITE(8,5) R1M,R2M
C
C X-LINE
C
DFM=0.
TLTM=0.
R1M=10000000.
R2M=10000000.
C
IF (IPRT.GT.2) WRITE(8,3)
DFMEAN=0.
DF2=0.
NDATA=0
C
DO 200 I=KHXMIN,KHXMAX-1
C
DO 205 J=KHYMIN,KHYMAX
A(J)=CHDD(J+NGY*(I-1))
205 CONTINUE
C
DFMAX=0.
DO 210 J=KHYMIN,KHYMAX-1
DF=DABS(A(J+1)-A(J))
NTOTAL=NTOTAL+1
IF (DF.LT.0.0001) THEN
R1=10000000.
ELSE
R1=YPT*YPT/(6.*DF)
END IF
TLT=DF/YPT*100.
IF (R1.LT.RLIM) NRLIM=NRLIM+1
IF (TLT.GT.TLIM) NTLIM=NTLIM+1
IF(DF.GT.DFMAX) DFMAX=DF
NDATA=NDATA+1
DFMEAN=DFMEAN+DF
DF2=DF2+DF*DF
210 CONTINUE
C
DFMAX =DFMAX*100.
TLTMAX=DFMAX/YPT
IF (DFMAX.GT.0) R2MIN=YPT*YPT/(2.*DFMAX*0.01)

```

```

R1MIN=R2MIN/3.
IF (DFMAX.GT.DFM)   DFM=DFMAX
IF (TLTMAX.GT.TLTM) TLTM=TLTMAX
IF (R1MIN.LT.R1M)   R1M=R1MIN
IF (R2MIN.LT.R2M)   R2M=R2MIN
C
IF (IPRT.GT.2) WRITE(8,2) I,DFMAX,TLTMAX,R1MIN,R2MIN
C
200 CONTINUE
C
DFMX=DFM
TLTMX=TLTM
R1MX=R1M
R2MX=R2M
DFSD=SQRT( (DF2-DFMEAN*DFMEAN/NDATA)/(NDATA-1) )
DFSDX=DFSD*100.
DFMEANX=DFMEAN/NDATA*100.
WRITE(8,6) DFMEANX,DFSDX
WRITE(8,4) DFMX,TLTMX
WRITE(8,5) R1MX,R2MX
C
DFMEAN= 0.5*(DFMEANX+DFMEANY)
DFSD  = 0.5*(DFSDX+DFSDY)
DFM   = DMAX1(DFMX,DFMY)
TLTM  = DMAX1(TLTMX,TLTMY)
R1M   = DMIN1(R1MX,R1MY)
R2M   = DMIN1(R2MX,R2MY)
RRATE = FLOAT(NRLIM)/NTOTAL*100.
TRATE = FLOAT(NTLIM)/NTOTAL*100.
WRITE(8,7) RRATE,TRATE
RETURN
C
1 FORMAT(/,' Y-Num.!'  Max.D.S.(cm)!'  Tmax(%)!'  Rmax1(m)',
2  ' Rmax2(m)')
2 FORMAT(' ',I5,' ',F6.2,' ',F6.3,' ',F8.0,' ',F8.0)
3 FORMAT(/,' X-Num.!'  Max.D.S.(cm)!'  Tmax(%)!'  Rmax1(m)',
2  ' Rmax2(m)')
4 FORMAT('  Max.Dif.Setl.(cm)='F6.2,'  Max.Tilt(%)='F6.3)
5 FORMAT('  Min.R1(m)='F8.0,'  Min.R2(m)='F8.0)
6 FORMAT(/,'  Ave.Dif.Setl(cm)='F6.2,
2  '  S.D.Dif.Setl.(cm)='F6.2)
7 FORMAT('  R-RATE(%)='F7.3,'  T-RATE='F7.3)
8 FORMAT('  * Mean OF Settlement ='F8.2,
2  '  * S.DEV of Settlement ='F8.2)
10 FORMAT(8F10.5)
END
C
C *****
C  DREAD --- INPUT DATA READ
C *****
SUBROUTINE  DREAD( RKJU, NKHN, NKJ, NTUC, TUCHI, NTUCHI, NSO, HT,
2             MESH, DRT, KXYS, ZZ, NGP, TIM, NCHNK,
3             NXYG, NGX, NGY, RK, SK, LSNO, HTT, NZMX
4             ,NKS, NKJX, NLOOP,NPT, TMZ, NTZ, NCHNP,
5             ITMZ, JJMAX )
C
IMPLICIT REAL*8 (A-H,O-Z)

```

```

DIMENSION  RKJU( 4,NKJ), NKHN( 4,NKJ), NTUC(2,NTUCHI),
2          TUCHI(16,NTUCHI), NSO(NZMX,KXYS),HT(NZMX,KXYS),
3          MESH(KXYS), DRT(  KXYS), ZZ(NGP), TIM(NCHNP),
4          NXYG( 8,NGP), LSNO(NZMX), HTT(NZMX), NKS(7,NKJ)
5          ,TMZ(NTZ), ITMZ(NTZ),DRT1(3000)
C
DIMENSION  MOJ(2), IWK(8)
DATA  MOJ/'X','Y'/
C
COMMON /PARM/ IFL2, TMCO, JMAX, JJMIN
COMMON /PRINT/ IPRT
C
          IEROR = 0
C
DO 100 I=1, NTUCHI
  READ(5,2) NTUC(1,I),NTUC(2,I), (TUCHI(J,I),J=1,8)
  READ(5,8) (TUCHI(J,I),J=9,16)
  IF(IPRT.EQ.1) THEN
    WRITE(6,12) NTUC(1,I), NTUC(2,I), (TUCHI(J,I),J=1,8)
    WRITE(6,8) (TUCHI(J,I),J=9,16)
  END IF
100 CONTINUE
C
DO 110 I= 1, KXYS
  DRT(I) = 0.
  MESH(I) = 0
110 CONTINUE
C
DO 250 NG= 1, NGP
  READ(5,1) ZZ(NG)
  IF(IPRT.EQ.1) WRITE(6,11) ZZ(NG)
C
  NS = NXYG(6,NG)
  NXX = NXYG(3,NG)
  NYY = NXYG(4,NG)
  NZZ = NXYG(5,NG)
  NXS = NXYG(1,NG)
  NYS = NXYG(2,NG)
  NXE = NXS+NXX -1
  NYE = NYS+NYY -1
C
DO 140 IX = NXS, NXE
  DO 130 IY = NYS, NYE
    NR = (IX-1)*NGY+IY
    DRT(NR) = ZZ(NG)
    DRT1(NR)= ZZ(NG)
130  CONTINUE
140  CONTINUE
C
150  READ(5,6)  MXL, MYL, MX, MY, MZ
  IF(IPRT.EQ.1) WRITE(6,16) MXL, MYL, MX, MY, MZ
  IF( MXL.LT.0 ) GO TO 240
C
  IF( MZ.EQ.0 ) MZ = NZZ
  READ(5,3) (LSNO(J),HTT(J),J=1,MZ)
  IF(IPRT.EQ.1) WRITE(6,13) (LSNO(J),HTT(J),J=1,MZ)
  IF( MZ.LE.NZZ ) GO TO 160

```



```

C
WRITE(6,21) MZ
      IEROR = IEROR +1
GO TO 150
C
160 MMJ= MOJ(1)
IF( MXL.GE.NXS .AND. MXL.LE.NXE .AND. MX.LE.NXX ) GO TO 180
170 WRITE(6,22) MMJ
      IEROR = IEROR +1
GO TO 150
C
180 MMJ = MOJ(2)
IF( MYL.LT. NYS .OR. MYL.GT.NYE .OR. MY.GT.NYY ) GO TO 170
C
MY = MYL+MY -1
MX = MXL+MX -1
C
DO 210 IX =MXL, MX
DO 200 IY =MYL, MY
C
C
NR = (IX-1)*NGY+IY
IF( MESH(NR) .EQ.0 ) GO TO 185
WRITE(6,25) IX,IY
      IEROR = IEROR +1
185 MESH(NR) = MZ
C
DO 230 I=1, MZ
  DO 215 J=1,NTUCHI
    IF( NTUC(1,J).EQ. LSNO(I) ) THEN
      LSNO(I)=J
      GO TO 225
    END IF
215 CONTINUE
WRITE(6,23) LSNO(I), I
      IEROR = IEROR +1
225 CONTINUE
IF( HTT(I) .LE.0. ) THEN
  WRITE(6,24) HTT(I), I
  IEROR = IEROR+1
END IF
IF( HTT(I).GT.30.0 ) THEN
  WRITE(6,31) HTT(I), I
END IF
NSO(I,NR)=LSNO(I)
HT(I,NR)=HTT(I)
230 CONTINUE
C
C
200 CONTINUE
210 CONTINUE
GO TO 150
C
240 CONTINUE
C
JX = 0
IERP = IEROR

```

```

DO 260 IX=NXS, NXE
  DO 255 IY = NYS, NYE
    NR = (IX-1)*NGY+IY
    IF( MESH(NR).GT.0 ) GO TO 255
      IEROR = IEROR +1
    IF( IEROR .GT. IERP+50 ) GO TO 255
    WRITE(6,26) IX, IY
255  CONTINUE
260  CONTINUE
C
250 CONTINUE
C
C  KAJYUU SAIKA DATA
C
ZL = 10000.
NKJX = 0
DO 295 I=1, NKJ
C
  READ(5,4) RT, (NKHN(J,I),J=1,4),(RKJU(J,I),J=2,4)
  IF(IPRT.EQ.1) WRITE(6,14) RT,(NKHN(J,I),J=1,4),(RKJU(J,I),J=2,4)
  RKJU(1,I) = RT* TMCO
C
  IF( I .EQ.1 ) GO TO 266
  RK = RKJU(1,I)-RKJU(1,I-1)
  IF( DABS(RK) .LT.0.001 ) GO TO 267
  IF( RK .GT. 0.001 ) GO TO 266
  WRITE(6,29 )
    IEROR = IEROR +1
266  NKJX = NKJX + 1
    NKS(1,NKJX) = I
267  IF( NKHN(1,I).GT.0 .AND. NKHN(1,I)+NKHN(3,I).LE.NGX+1) GO TO 270
C
  WRITE(6,22) MOJ(1)
    IEROR = IEROR +1
270  IF( NKHN(2,I).GT.0 .AND. NKHN(2,I)+NKHN(4,I).LE.NGY+1) GO TO 272
  WRITE(6,22) MOJ(2)
    IEROR = IEROR +1
272  KJXS = NKHN(1,I)
    KJYS = NKHN(2,I)
    KJXE = KJXS+NKHN(3,I)-1
    KJYE = KJYS+NKHN(4,I)-1
C
  IERP = IEROR
  DO 290 NG = 1, NGP
    NS = NXYG(6,NG)
    NXX = NXYG(3,NG)
    NYY = NXYG(4,NG)
    NXS = NXYG(1,NG)
    NYS = NXYG(2,NG)
    NXE = NXS+NXX -1
    NYE = NYS+NYY -1
    IF( NXS.GT.KJXE) GO TO 290
    IF( NYS.GT.KJYE ) GO TO 290
    IF( NXE.LT.KJXS) GO TO 290
    IF( NYE.LT.KJYS) GO TO 290
    LXS = NXS
    IF( NXS.LT.KJXS) LXS = KJXS

```

```

      LYS = NYS
      IF( NYS.LT.KJYS) LYS = KJYS
      LXE = NXE
      IF( NXE.GT.KJXE ) LXE = KJXE
      LYE = NYE
      IF( NYE.GT.KJYE ) LYE = KJYE
C
      DO 285 IX = LXS, LXE
      DO 280 IY= LYS, LYE
        NR = (IX-1)*NGY+IY
        IF( DABS(DRT1(NR)-RKJU(4,I)).LT.0.001 ) GO TO 275
        RKJU(4,I)=RKJU(4,I)+RKJU(3,I)
        IF(DABS(DRT1(NR)-RKJU(4,I)).LT.0.001) GO TO 275
        IF( IEROR .GT. IERP+20 ) GO TO 275
        WRITE(6,27) IX, IY,DRT1(NR), RKJU(4,I)
        RKJU(4,I)=DRT1(NR)
275      DRT1( NR)=RKJU(4,I)-RKJU(3,I)
280      CONTINUE
285      CONTINUE
290      CONTINUE
295 CONTINUE
C
      NKJM = NKJX -1
      IF( NKJM .LE.0 ) GO TO 310
      DO 300 I= 1, NKJM
        NKS(2,I) = NKS(1,I+1)-1
        NKS(3,I) = 0
300 CONTINUE
310 NKS(2,NKJX) = NKJ
      NKS(3,NKJX) = 0
C
C      CHINKA KEISAN JIKAN
C
      READ(5,5) (TIM(J),J=1, NCHNK)
      WRITE(6,15) (TIM(J),J=1,NCHNK)
      TIM(NCHNP) = 1200.
C
      DO 500 J= 1, NCHNP
        TIM(J) = TIM(J)* TMCO
        IF( J.EQ.1 ) GO TO 500
        IF( TIM(J-1).LT. TIM(J)) GO TO 500
        WRITE(6,29)
        IEROR = IEROR +1
500 CONTINUE
C
      READ(5,5) (TMZ(J),J=1,NTZ)
      IF(IPRT.EQ.1) WRITE(6,15) (TMZ(J),J=1,NTZ)
C
      DO 506 J= 1, NTZ
        TMZ(J) = TMZ(J)*TMCO
C
      DO 501 I= 1, NCHNP
        IF( DABS(TMZ(J)-TIM(I)).LT.0.001 ) GO TO 502
501 CONTINUE
C
      TMZ(J) = TMZ(J)/TMCO
      WRITE(6,30) TMZ(J)

```

```

IEROR = IEROR+1
GO TO 506
502 ITMZ(J)=I
506 CONTINUE
C
C
508 IF( IEROR .EQ.0 ) GO TO 509
WRITE(6,20 ) IEROR
C
509 WRITE(6,10)
DO 515 J= 1, NKJX
NS = NKS(1,J)
NE = NKS(2,J)
KJ = 0
KRXMI = 10000
KRYMI = 10000
KRXXM = 0
KRYMX = 0
DO 510 I= NS, NE
NX = NKHN(3,I)
NY = NKHN(4,I)
NXY = NX*NY
NKS(3,I) = KJ
IF( KRXXM .LT. NKHN(1,I)+NX-1 ) KRXXM = NKHN(1,I)+NX-1
IF( KRYMX .LT. NKHN(2,I)+NY-1 ) KRYMX = NKHN(2,I)+NY-1
IF( KRXMI .GT. NKHN(1,I) ) KRXMI = NKHN(1,I)
IF( KRYMI .GT. NKHN(2,I) ) KRYMI = NKHN(2,I)
KJ = KJ+ NXY
510 CONTINUE
NKS(4,J) = KRXMI
NKS(5,J) = KRYMI
NKS(6,J) = KRXXM
NKS(7,J) = KRYMX
515 CONTINUE
C
C
DO 520 I=1, NGP
NXYG(3,I) = NXYG(3,I)+ NXYG(1,I) -1
NXYG(4,I) = NXYG(4,I)+ NXYG(2,I) -1
520 CONTINUE
C
DO 530 I=1, NKJ
NKHN(3,I) = NKHN(3,I) + NKHN(1,I) -1
NKHN(4,I) = NKHN(4,I) + NKHN(2,I) -1
530 CONTINUE
C
JMAX = NCHNK + 9
JJMIN = JMAX + 1
JJMAX = JMAX + NKJX
BLK = 3840.
LNK2 = KXYS*(NCHNK+1)*NLOOP/BLK*1.2
IF( LNK2 .EQ.0 ) LNK2 = 1
LNK2 = (LNK2+4)/ 5*5
WRITE(6,41) IFL2
C
RETURN
C

```

```

C ** FORMAT **
  1 FORMAT(2F10.0)
  2 FORMAT(2I3,8F9.5)
  3 FORMAT(4(I10,F10.0))
  4 FORMAT(F10.0, 4I10, 3F10.0)
  5 FORMAT(8F10.0)
  6 FORMAT(8I10)
  7 FORMAT(8F10.3)
  8 FORMAT(8F8.4)

C
C *****
10 FORMAT(/' ', 30X, '** END OF INPUT DATA ECHO **'//)
11 FORMAT(' ', 2F15.4)
12 FORMAT(2I3,8F8.4)
13 FORMAT(' ', I10,F15.3, I10,F15.3, I10,F15.3, I10,F15.3)
14 FORMAT(' ', F10.0, 4I10, 3F15.2)
15 FORMAT(' ', 8F15.3)
16 FORMAT(' ', 8I10)
17 FORMAT(' CHINKA NO KIZYUNN TAKASA = ',F10.3)

C
20 FORMAT(/' * EXECUTION TERMINATED BECAUSE OF ABOVE ', I3, ' ERROR(S)
  2')
21 FORMAT('** ERROR 01 MZ OVER NZ ')
22 FORMAT('** ERROR 02 DATA RANGE OVER ',A1, ' DIRECTION' )
23 FORMAT('** ERROR 03 NO TUCHI DATA NO =',I3, ' Z=',I4)
24 FORMAT('** ERROR 04 SOOATSU IS LEAST THAN ZERO ',F6.2, ' Z=',
  2 I3)
25 FORMAT('** ERROR 05 OVERLAPPED CHISOO DATA X=',I4, ' Y=',I4)
26 FORMAT('** ERROR 06 NO CHISOO DATA X=',I4, ' Y=',I4)
27 FORMAT('** ERROR 07 KAJUU SAIKA DATA X=',I4, ' Y=',I4,
  2 ' NOW=',F6.2, ' DATA=',F6.2 )
28 FORMAT('** ERROR 08 MISSING KAJUU SAIKA DATA X=',I4, ' Y=',
  2 I4, ' NOW=',F6.2)
29 FORMAT('** ERROR 09 TIME IS NOT IN ASCENDING ORDER')
30 FORMAT('** ERROR 10 DATA ',F10.1, ' DON" T MATCH TO CHINKA DATA'
  2 )
31 FORMAT(' WARN. 01 SOOATSU IS GREATER THAN 20M ',F8.2, ' Z='
  2 ,I3)
32 FORMAT('** ERROR 11 X=',I4, ' Y=',I4, ' IS NOT IN ANY GROUP')

C
C
41 FORMAT(/' FILE LINK SIZE ...',5X,3(FILE CODE ',I2, '= ',I4,5X
  2 ))
  END

C
C *****
C BUSINQ DISTRIBUTION OF STRESS
C *****
C
  FUNCTION BUSINQ(SX,SY,SZ,XKJ,YKJ,ZKJ,XPT,YPT,Q)

C
C CHICYUU OORYOKU NO KEISAN
C
  IMPLICIT REAL*8 (A-H,O-Z)

C
  SA = XPT*0.5
  SB = YPT*0.5

```

```

X1 = SX-XKJ+SA
Y1 = SY-YKJ+SB
Z  = SZ-ZKJ
X2 = SX-XKJ-SA
Y2 = SY-YKJ-SB
C
ZZ = Z*Z
XX1 = X1*X1
YY1 = Y1*Y1
XX2 = X2*X2
YY2 = Y2*Y2
C
X1Z = XX1+ZZ
X2Z = XX2+ZZ
Y1Z = YY1+ZZ
Y2Z = YY2+ZZ
X1Y1Z = X1Z+YY1
X1Y2Z = X1Z+YY2
X2Y1Z = X2Z+YY1
X2Y2Z = X2Z+YY2
X1Y1 = X1*Y1
X2Y2 = X2*Y2
X1Y2 = X1*Y2
X2Y1 = X2*Y1
C
SX1Y1 = DSQRT( X1Y1Z )
SX1Y2 = DSQRT( X1Y2Z )
SX2Y1 = DSQRT( X2Y1Z )
SX2Y2 = DSQRT( X2Y2Z )
C
BUSINQ = DATAN2(X1Y1,Z*SX1Y1) + DATAN2(X2Y2,Z*SX2Y2)
2      -(DATAN2(X1Y2,Z*SX1Y2) + DATAN2(X2Y1,Z*SX2Y1))
3      +(Z*X1Y1*(X1Y1Z+ZZ)/(X1Z*Y1Z*SX1Y1)
4      +Z*X2Y2*(X2Y2Z+ZZ)/(X2Z*Y2Z*SX2Y2)-
5      (Z*X1Y2*(X1Y2Z+ZZ)/(X1Z*Y2Z*SX1Y2)
6      +Z*X2Y1*(X2Y1Z+ZZ)/(X2Z*Y1Z*SX2Y1)))
      IF( BUSINQ .LT. 1.E-5 ) BUSINQ = 0.
C      WRITE(6,10) SX,SY,SZ,XKJ,YKJ,BUSINQ
10 FORMAT(6F12.5)
C
      BUSINQ = BUSINQ* Q / 6.2831852
C
      RETURN
      END
C
C *****
C      CALCU    CALCULATION OF TIME-SETTLEMENT RELATION
C *****
C
      SUBROUTINE CALCU( IX,IY,NTUC,NSO,HT,CVV,TVV,UU,HDS,MZ,TT,NZMX,
2          NTUCHI
          )
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION  NTUC(2,NTUCHI), NSO(NZMX), HT(NZMX), CVV(NZMX)
      DIMENSION  TVV(NZMX), UU(NZMX), HDS(NZMX), DMM(10), DMZ(10)
      DIMENSION  DMA(10)
C

```

```

DATA DMM/1.570795, 4.712386, 7.853977, 10.99557, 14.13715,
2      17.27875, 20.42033, 23.56192, 26.70352, 29.8451 /
DATA DMZ/2.467397, 22.20657, 61.68495, 120.9025, 199.8591,
2      298.5549, 416.9900, 555.1641, 713.0799, 890.7305 /
C
C
TOLE = 5.E-4
PAI = 3.14159265
PX = -PAI*PAI/4.0
DO 50 I = 1, MZ
HDS(I) = HT(I)
UU(I) = 0.
50 CONTINUE
C
IZS = 1
100 IZE = IZS
IF(NTUC(2,NSO(IZS)).EQ.0) GO TO 500
110 IF(NTUC(2,NSO(IZE)).EQ.0) GO TO 120
IF( IZE.EQ.MZ) GO TO 125
IZE = IZE+1
GO TO 110
C
120 IZE = IZE-1
125 IF(IZS.EQ.IZE) GO TO 300
C
HD = 0.
DO 150 I = IZS, IZE
HDS(I) = HDS(I)*DSQRT(CVV(IZS)/CVV(I))
HD = HD+HDS(I)
150 CONTINUE
IF(IZE.NE.MZ) THEN
HD = HD*0.5
END IF
TV=CVV(IZS)*TT/(HD*HD)
C
1 FORMAT(4I4,TT=,F15.7, 'TV=,F15.7)
C
C ** kanzan soatsu hou ***
C
RVP = 0.
HS = 0.
C
DO 250 I=IZS, IZE
HS = HS+HDS(I)
RV = 0.
RVM =0.
C
DO 200 M = 1, 10
DM = DMM(M)
A1=-DMZ(M)*TV
DMA(M)=DEXP(A1)/DM
RVT =DSIN(DM*HS/HD)*DMA(M)
IF( DABS(RVT) .LT. TOLE ) GO TO 210
RV = RV + RVT
200 CONTINUE
C
210 CONTINUE

```

```

HSM = HS-HDS(I)*0.5
DO 220 M = 1, 10
  DM = DMM(M)
  RVMT = DSIN(DM*HSM/HD)*DMA(M)
  IF( DABS(RVMT) .LT. TOLE ) GO TO 230
  RVM = RVM + RVMT
220 CONTINUE
C
230 IF (RVM.GT.0.5) RVM=0.5
  UU(I) = 1.-(RV+RVP+2.*RVM)*0.5
  IF (UU(I).LT.0.) UU(I)=0.
  RVP = RV
250 CONTINUE
  GO TO 500
C
300 CONTINUE
  HD=HDS(IZS)
  IF(IZE.NE.MZ) THEN
    HD = HD*0.5
  END IF
  TV=CVV(IZS)*TT/(HD*HD)
  IF(TV.GT.0.2) THEN
    UU(IZS) = 1.-0.8106*DEXP(PX*TV)
  ELSE
    UU(IZS) = 2.0*DSQRT(TV/PAI)
  END IF
C
C
500 IZS = IZE +1
  IF( IZS.LE. MZ ) GO TO 100
  RETURN
  9 FORMAT('  UU ',I5,6F8.4)
  END
C
C *****
C GAUSS ----  RANDOM VARUABLES
C *****
C
  SUBROUTINE GAUSS(IRANDX, S, AM, V )
C
C RANSUU KEISAN SUB
C
  A= 0.
  DO 50 I=1, 12
    Y = RAN(IRANDX)
  50 A = A+Y
  V = (A-6.0)*S +AM
  RETURN
  END
C
C *****
C SOLDAT ---  SOIL DATA
C *****
  SUBROUTINE SOLDAT(NGX,NGY,NTUC,COV,RS,HT,
2 TUCHI,IRANDX,NSO, JJMAX,NZMX,KXYS,NTUCHI )
C
  IMPLICIT REAL*8 (A-H,O-Z)

```



```

REAL*8 MCV,MCC,MPY,ME0,MEAN
DIMENSION RS(IJMAX,NZMX,KXYS),TUCHI(16,NTUCHI),NSO(NZMX,KXYS),
2      NTUC(2,NTUCHI)
DIMENSION HT(50,3000),DEPTH(50,1000)
DIMENSION A(1000),B(1000),C(10,1000)
DIMENSION COV(1000,1000)
C
C      ***** H *****
DO 20 INC=1,NTUCHI
  DO 25 IR=1,NGX*NGY
    DO 30 K=1,10
      MEAN=0.
      SD=1.0
      CALL GAUSS(IRANDX,SD,MEAN,A(IR))
30    CONTINUE
25  CONTINUE
    DO 40 IR=1,NGX*NGY
      SDC=TUCHI(16,INC)
      C(INC,IR)=0.
      DO 45 JR=1,IR
        C(INC,IR)=C(INC,IR)+COV(IR,JR)*A(JR)*SDC
45    CONTINUE
40  CONTINUE
20 CONTINUE
C
  DO 50 IX=1,NGX
    DO 55 IY=1,NGY
      Z=0.
      PVS=0.
      IR=(IX-1)*NGY+IY
C
      DO 60 IZ=1,NZMX
        NC=NSO(IZ,IR)
        HT(IZ,IR)=HT(IZ,IR)*(1+C(NC,IR))
        IF (HT(IZ,IR).LE.0. ) HT(IZ,IR)=0.2
        GAM = TUCHI(1,NC)
        ZH  = HT(IZ,IR) *0.5
        DLP = GAM*ZH
        RS(1,IZ,IR) = DLP+PVS
        PVS  = PVS+DLP+DLP
        DO 65  I = 2, 5
          RS(I,IZ,IR) = 0.
65        CONTINUE
          RS(6,IZ,IR) = Z
          DEPTH(IZ,IR) = Z
          Z = Z+ZH+ZH
          DO 70 I=7,9
            RS(I,IZ,IR)=0.
70        CONTINUE
60      CONTINUE
C
55    CONTINUE
50 CONTINUE
C
C
C      DO 800 IZ=1,NZMX
C

```

```

NC=NSO(IZ,1)
IF (NTUC(2,NC).EQ.0) GO TO 800
IF (NTUC(2,NC).EQ.2) GO TO 180
C
C ***** CC *****
DO 80 IR=1,NGX*NGY
  DO 85 K=1,10
    MEAN=0.
    SD=1.0
    CALL GAUSS(IRANDX,SD,MEAN,A(IR))
85  CONTINUE
80 CONTINUE
C
DO 90 IR=1,NGY*NGX
  NC=NSO(IZ,IR)
  MCC=TUCHI(9,NC)
  SDCC1=TUCHI(10,NC)
C
  B(IR)=0.
  DO 95 JR=1,IR
    NC=NSO(IZ,JR)
    SDCC2=TUCHI(10,NC)
    B(IR)=B(IR)+COV(IR,JR)*A(JR)*SQRT(SDCC1*SDCC2)
95  CONTINUE
C
  RS(7,IZ,IR)=B(IR)+MCC
  IF (RS(7,IZ,IR).LT.0.15) RS(7,IZ,IR)=0.15
90 CONTINUE
C
C ***** E0 *****
C
DO 100 IR=1,NGX*NGY
  DO 105 K=1,10
    MEAN=0.
    SD=1.0
    CALL GAUSS(IRANDX,SD,MEAN,A(IR))
105  CONTINUE
100 CONTINUE
C
DO 120 IR=1,NGY*NGX
  NC=NSO(IZ,IR)
  ME0=TUCHI(11,NC)
  SDE01=TUCHI(12,NC)
  B(IR)=0.
C
  DO 125 JR=1,IR
    NC=NSO(IZ,JR)
    SDE02=TUCHI(12,NC)
    B(IR)=B(IR)+COV(IR,JR)*A(JR)*SQRT(SDE01*SDE02)
125  CONTINUE
C
  RS(8,IZ,IR)=B(IR)+ME0
  IF (RS(8,IZ,IR).LT.0.4) RS(8,IZ,IR)=0.4
120 CONTINUE
C
C***** PY *****
C

```

```

DO 130 IR=1,NGX*NGY
  DO 135 K=1,10
    MEAN=0.
    SD=1.0
    CALL GAUSS(IRANDX,SD,MEAN,A(IR))
135  CONTINUE
130 CONTINUE
C
DO 140 IR=1,NGY*NGX
  NC=NSO(IZ,IR)
  Z=DEPTH(IZ,IR)
  ZH=HT(IZ,IR)*0.5
  MPY=TUCHI(13,NC)*(Z+ZH)+TUCHI(14,NC)
  SDPY1=TUCHI(15,NC)*MPY
C
  B(IR)=0.
  DO 145 JR=1,IR
    NC=NSO(IZ,,JR)
    Z=DEPTH(IZ,,JR)
    ZH=HT(IZ,,JR)*0.5
    MPY=TUCHI(13,NC)*(Z+ZH)+TUCHI(14,NC)
    SDPY2=TUCHI(15,NC)*MPY
    B(IR)=B(IR)+COV(IR,JR)*A(JR)*SQRT(SDPY1*SDPY2)
145  CONTINUE
C
  RS(9,IZ,IR)=B(IR)+MPY
  IF (RS(9,IZ,IR).LT.0.1) RS(9,IZ,IR)=0.1
140 CONTINUE
C
C ***** CV *****
C
DO 150 IR=1,NGX*NGY
  MEAN=0.
  SD=1.0
  CALL GAUSS(IRANDX,SD,MEAN,A(IR))
150 CONTINUE
C
DO 160 IR=1,NGY*NGX
  NC=NSO(IZ,IR)
  MCV=TUCHI(5,NC)
  SDCV1=TUCHI(6,NC)
  B(IR)=0.
  DO 155 JR=1,IR
    NC=NSO(IZ,,JR)
    SDCV2=TUCHI(6,NC)
    B(IR)=B(IR)+COV(IR,JR)*A(JR)*SQRT(SDCV1*SDCV2)
155  CONTINUE
  RS(3,IZ,IR)=10.00**(B(IR)+MCV)
160 CONTINUE
  GO TO 800
C
180 CONTINUE
C
DO 190 IR=1,NGX*NGY
  NC=NSO(IZ,IR)
  MCC=TUCHI(9,NC)
  RS(7,IZ,IR)=MCC

```

```

      ME0=TUCHI(11,NC)
      RS(8,IZ,IR)=ME0
      Z=DEPTH(IZ,IR)
      ZH=HT(IZ,IR)*0.5
      MPY=TUCHI(13,NC)*(Z+ZH)+TUCHI(14,NC)
      RS(9,IZ,IR)=MPY
      MCV=TUCHI(5,NC)
      RS(3,IZ,IR)=10.00**MCV
190 CONTINUE
C
800 CONTINUE
C
      RETURN
C
1  FORMAT(I5,'NC=',I5)
2  FORMAT(2I5,'COV=',F15.6,'  MPY=',F15.6)
3  FORMAT(2I4,'NC=',I3,'Z=',F10.3,' TUCHI13=',F10.4,
2  'SDPY2=',F15.6)
      END
C
C *****
C  LU
C *****
      SUBROUTINE LU(A,N)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(1000,1000),B(1000,1000)
C
      B(1,1)=DSQRT(A(1,1))
      I=1
      DO 100 J=2,N
          B(I,J)=A(I,J)/B(1,1)
100 CONTINUE
C
      DO 200 I=2,N-1
C
          SGKI=0.
          DO 150 K=1,I-1
              SGKI=SGKI+B(K,I)*B(K,I)
150  CONTINUE
          IF( (A(I,I)-SGKI).LE.0. ) THEN
              WRITE(6,99) A(I,I),SGKI
              GO TO 155
          END IF
          99  FORMAT(4F15.8)
          B(I,I)=DSQRT(A(I,I)-SGKI)
155  CONTINUE
C
          DO 170 J=I+1,N
              SGKIJ=0.
              DO 180 K=1,I-1
                  SGKIJ=SGKI+B(K,I)*B(K,J)
180  CONTINUE
              B(I,J)=( A(I,J)-SGKIJ )/B(I,I)
170  CONTINUE
C
200 CONTINUE

```

```

C
  SGKN=0.
  DO 250 K=1,N-1
    SGKN=SGKN+B(K,N)*B(K,N)
250 CONTINUE
  IF ( DSQRT(A(N,N)-SGKN).LE.0. ) THEN
    B(N,N)=0.
    GO TO 255
  END IF
  B(N,N)=DSQRT(A(N,N)-SGKN)
255 CONTINUE
C
  DO 350 I=1,N
    DO 400 J=1,N
      A(I,J)=B(J,I)
400  CONTINUE
350 CONTINUE
  RETURN
C
  1 FORMAT(2I4,2F15.8)
  2 FORMAT(I4,F15.8)
  3 FORMAT('   *** SUB LU ****')
  4 FORMAT('  N=',I4)
  END
C
C *****
C   SOLDAT2 --- SOIL DATA
C *****
C   SUBROUTINE SOLDAT2(NGX,NGY,NTUC,RS,HT,MESH,TUCHI,IRANDX,NSO,
1  JJMAX,NZMX,KXYS,NTUCHI )
C
C   IMPLICIT REAL*8 (A-H,O-Z)
C   REAL*8 MCV,MCC,MPY,ME0,MH
C   DIMENSION RS(JJMAX,NZMX,KXYS),TUCHI(16,NTUCHI),NSO(NZMX,KXYS),
1  NTUC(2,NTUCHI),MESH(NZMX)
C   DIMENSION HT(50,3000)
C
C   DO 900 IX=1,NGX
C     DO 800 IY=1,NGY
C
C     IR=(IX-1)*NGY+IY
C     NC=0
C     Z=0.
C     PVS=0.
C     MZ=MESH(IR)
C
C     DO 700 IZ=1,MZ
C
C     NC=NSO(IZ,IR)
C     IF(NC.EQ.NC0) GO TO 30
C
C     ***** H *****
C   15  DO 20 K=1,5
C       MH=1.0
C       SDH=TUCHI(16,NC)
C       CALL GAUSS(IRANDX,SDH,MH,HRANDM)
C   20  CONTINUE

```

```

      IF (HRANDM.LE.0) GO TO 15
C
30   NC0=NC
C     WRITE(6,8) IR,IZ,HT(IZ,IR),HRANDM,NC,NC0
C 8   FORMAT(2I4,2F8.4,2I4)
      HT(IZ,IR)=HT(IZ,IR)*HRANDM
      GAM = TUCHI(1,NC)
      ZH  = HT(IZ,IR)*0.5
      DLP = GAM*ZH
      RS(1,IZ,IR) = DLP+PVS
      PVS  = PVS+DLP+DLP
      DO 40  I = 2, 5
          RS(I,IZ,IR) = 0.
40   CONTINUE
C
      RS(6,IZ,IR) = Z
      Z = Z+ZH+ZH
      DO 45  I = 7,9
          RS(I,IZ,IR)=0.
45   CONTINUE
C
      IF( NTUC(2,NC).EQ.0 ) GO TO 700
C
C     **** Coefficient of Consolidation   Cv
C     DO 50  J= 1, 5
          MCV = TUCHI(5,NC)
          SDCV= TUCHI(6,NC)
          CALL GAUSS (IRANDX,SDCV,MCV,CVP)
50   CONTINUE
      RS(3,IZ,IR) = 10.**CVP
C
C     **** Compression Index   Cc > 0.15
C     DO 60 J=1,5
          MCC = TUCHI(9,NC)
          SDCC= TUCHI(10,NC)
          CALL GAUSS(IRANDX,SDCC,MCC,RS(7,IZ,IR))
60   CONTINUE
      IF(RS(7,IZ,IR).LT.0.15) THEN
          GO TO 55
      END IF
C
C     ***** Initial Void ratio
C     DO 70  J=1,5
          ME0 = TUCHI(11,NC)
          SDE0= TUCHI(12,NC)
          CALL GAUSS(IRANDX,SDE0,ME0,RS(8,IZ,IR))
70   CONTINUE
      IF(RS(8,IZ,IR).LT.0.4) THEN
          RS(8,IZ,IR)=0.4
      END IF
C
C     ***** Py
C     MPY  = TUCHI(13,NC)*(Z-ZH) + TUCHI(14,NC)
      SDPY = TUCHI(15,NC)*MPY
75   DO 80  J=1,5
          CALL GAUSS(IRANDX,SDPY,MPY,RS(9,IZ,IR))
80   CONTINUE

```

```
IF(RS(9,IZ,IR).LT.0.) THEN  
  RS(9,IZ,IR)=0.1  
END IF
```

C

```
700 CONTINUE
```

```
800 CONTINUE
```

```
900 CONTINUE
```

```
  RETURN
```

C

```
11 FORMAT(3I5,3F15.5)
```

```
21 FORMAT(8F10.5)
```

```
  END
```

-