



土木研究所資料
第687号

二次元および軸対称問題における
有限要素法

昭和46年7月

建設省土木研究所

目 次

ま え が き	1
第1章 二次元問題解析用プログラム	3
1-1 概 要	3
1-2 フローチャート	10
1-3 入力データ作成の説明	11
1-4 プログラム その1	13
1-5 プログラム その2	57
第2章 軸対称問題解析用プログラム	99
2-1 概 要	99
2-2 入力データ作成の説明	102
2-3 プログラム	103
第3章 解 析 例	147
3-1 線形計算	147
3-2 非線形計算	152
あ と が き	155
参 考 文 献	156
付 録	157
プログラム AXIFEM	
プログラム NAXIFEM	
プログラム BO 2991	

ま え が き

最近の土工、特に盛土工事において盛土材料の選択が自由にならないことや工期が限定されていることのために、かなり圧縮性の高い材料（関東ローム等）を用いて急速施工がなされる場合が多い。このような盛土では落ち着くまでかなりの圧縮沈下、不等沈下等を生じ、年々増加の途をたどる交通荷重とも相伴って問題になっている。この盛土体の圧縮沈下量の推定のために作成したプログラムを報告するものである。

なお本報告は、さきに道路事業調査費「盛土の安定性に関する試験調査」において「有限要素法による盛土等の内部応力変形解析」を報告したが、さらにプログラムを拡張した二次元問題解析用プログラムおよび軸対称問題解析用プログラムについて述べたものである。

施工研究室 伊勢田 哲也
田 中 俊彦
岡 村 栄三

第 1 章 二次元問題解析用プログラム

§ 1-1 概 要

このプログラムは変形およびひずみが微小であると考えられる時に成立する微小変形理論による弾性体解析用に作成したものである。

プログラムの基となる計算式は土木研究所資料第 546 号「有限要素法による盛土等の内部応力変形解析」に記述されているものを用いている。(詳しい説明は上記資料を参照)

非線形計算には反復法を用いている。この反復法の弾性常数の計算には減衰定数 α なる常数を用いて $(n+1)$ サイクルの応力値 σ_{n+1} と n サイクルでの応力値 σ_n とにそれぞれ重みをつけ σ_{n+1} の値を補正している。

つまり

$$\sigma_{n+1} = (1 - \alpha) \sigma_n + \alpha \sigma_{n+1}$$

ここに $0 < \alpha \leq 1$ である。

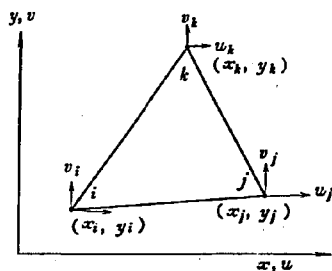
$\alpha = 1$ なる時は補正なしの σ_{n+1} の値を用いて計算している事になる。

また収束の判定として

- (イ) 全節点変位差の絶対値がある収束範囲内に収まるか
- (ロ) 全節点変位差の二乗の総和がある収束範囲内に収まるかどうかを考え、前者の方式によるプログラムを(その 1) 後者の方式によるそれを(その 2) として以下に示す。なおプログラム(その 2) では変形後の座標に基づいて剛性マトリクスを計算しなおしている。

このプログラムは線形計算と非線形計算をインプットデータで区別出来る。

計算に用いた計算式を以下に簡単に示しておく。



変位関数

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{Bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{Bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{Bmatrix} \dots (1-1)$$

用いた三角形要素

応力とひずみの関係式

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = C_1 \begin{Bmatrix} 1 & C_2 & 0 \\ C_2 & 1 & 0 \\ 0 & 0 & \frac{1-C_2}{2} \end{Bmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} \dots (1-2)$$

等方性体の場合平面応力状態では

$$C_1 = \frac{E}{1 - \nu^2}, \quad C_2 = \nu$$

平面ひずみ状態では

$$C_1 = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)}, C_2 = \frac{\nu}{1-\nu}$$

応力マトリクス

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \frac{C_1}{\phi} \begin{bmatrix} y_{ik} & C_2 x_{kj} & y_{ki} & C_2 x_{ik} & y_{ij} & C_2 x_{ji} \\ C_2 y_{jk} & x_{kj} & C_2 y_{ki} & x_{ik} & C_2 y_{ij} & x_{ji} \\ \lambda_1 x_{kj} & \lambda_1 y_{jk} & \lambda_1 x_{ki} & \lambda_1 y_{ki} & \lambda_1 x_{ji} & \lambda_1 y_{ji} \end{bmatrix} \times \begin{Bmatrix} u_i \\ v_i \\ u_j \\ v_j \\ u_k \\ v_k \end{Bmatrix} \dots (1-3)$$

ここに $x_{ik} = x_i - x_k$ etc,

$$\phi = x_{ji} * y_{ki} - x_{ki} * y_{ji},$$

$$\lambda_1 = \frac{1-C_2}{2}$$

剛性マトリクス

$$K_n = \frac{C_1 l}{2\phi} \begin{bmatrix} \lambda_1 x_{jk}^2 + y_{jk}^2 & & & & & \\ \lambda_2 x_{kj} y_{jk} & \lambda_1 y_{jk}^2 + x_{jk}^2 & & & & \\ \lambda_1 x_{ki} x_{kj} & \lambda_1 x_{ki} y_{kj} & \lambda_1 x_{ki}^2 + y_{ki}^2 & & & \\ +y_{jk} y_{ki} & +C_2 x_{kj} y_{ki} & & & & \\ \lambda_1 x_{kj} y_{ki} & \lambda_1 y_{jk} y_{ki} & \lambda_2 x_{ki} y_{ki} & \lambda_1 y_{ki}^2 + x_{ki}^2 & & \\ +C_2 x_{ki} y_{jk} & +x_{ki} x_{jk} & & & & \\ \lambda_1 x_{ji} x_{jk} & \lambda_1 x_{ji} y_{jk} & \lambda_1 x_{ji} x_{ki} & \lambda_1 x_{ji} y_{ki} & \lambda_1 x_{ji}^2 & \\ +y_{ji} y_{jk} & +C_2 x_{kj} y_{ji} & +y_{ji} y_{ki} & +C_2 x_{ki} y_{ji} & +y_{ji}^2 & \\ \lambda_1 x_{ki} y_{ji} & \lambda_1 y_{ji} y_{jk} & \lambda_1 x_{ki} y_{ji} & \lambda_1 y_{ji} y_{ki} & \lambda_2 x_{ji} y_{ji} & \lambda_1 y_{ji}^2 \\ +C_2 x_{ji} y_{jk} & +x_{ji} x_{jk} & +C_2 x_{ji} y_{ki} & +x_{ji} x_{ki} & & +x_{ji}^2 \end{bmatrix} \dots (1-4)$$

三角形要素の剛性マトリックス

$$\text{ここで } \lambda_2 = \frac{1+C_2}{2}$$

連立方程式の解法

有限要素法では節点変位と節点力を関係づける多元連立方程式をいかにうまく処理するかという事が問題になってくる。この連立方程式の解法には消去法と反復法および両者の特徴を組合せた共役傾斜などがあげられる。

これらの特徴を簡単に示したのが表 1-1 である。

この報文のプログラムでは消去法の流れをくむユニット分割法を用いている。

(I) UNIT 分割法は構造物を適当なユニットに分割し各ユニット間の接触状態により節点をクラスに分類し、そのクラス分類の特性を利用して全体の釣合方程式を書き変えて解く方法である。

このクラス分類の定義は次のようである。

表 1.1

連立1次方程式 $f = kx$ の解法 $k \cdots n$ 行 n 列のマトリックス (剛性マトリックス, 正値対称) $x \cdots n$ 次のベクトル (未知変位) $f \cdots n$ 次のベクトル (外力)			
解 法	ユニット分割法	SOR法 (Successive Over relaxation Method)	共役傾斜法 (Conjugate Gradient Method)
類 別	消 去 法	反 復 法	反 復 法
解法の概要	解析の対象とする構造をいくつかの部分 (ユニットと称する) に分割して、一つのユニットについてつり合方程式 $f_i = kx$ (ユニット i について) を立て、全体の計算を小規模な逆行列計算の積み重ねに変換する。逆行列演算が基本	初期操作 $x = x_0$ $r_0 = f - kx_0$ 反復操作 $r_i = f - kx_i$ $x_{i+1} = x_i + \beta r_i$ 収束判定 $ x_{i+1} - x_i / x_i < \epsilon$ ϵ は相対誤差でデータとして与える。 r 残差ベクトル β 加速常数 (Overrelaxation factor) 構造問題では $\beta = 1.8 \sim 1.95$ $\beta = 1$ が普通の relaxation method	初期操作 $p_0 = r_0 = f - kx_0$ 反復操作 $\alpha_i = (p_i, r_i) / (p_i, k p_i)$ $x_{i+1} = x_i + \alpha_i p_i$ $r_{i+1} = r_i - \alpha_i k p_i$ $\beta_i = (k p_i, r_{i+1}) / (p_i, k p_i)$ $p_{i+1} = r_{i+1} - \beta_i p_i$ 収束判定 $ x_{i+1} - x_i / x_i < \epsilon$
長 所	いくつもの荷重条件について計算できる。 計算時間が推定しやすい。 収束の心配がなく、一応解が得られる。	剛性マトリックスの非ゼロ要素のみをおぼえさせればよいので、メモリが有効に使える。 直接法なので誤差の評価がしやすい。	左と同じ 左と同じ n (方程式の元数) 回の反復によって必ず収束する (丸めの誤差のない場合)
短 所	構造の形状によってはユニット分割が用をなさない (3次元問題など)。 解の精度がわかりにくい。 (反力で推定している)。	荷重条件が変わると計算をもう一度やりなおさねばならない。 計算時間が推定しにくい。 加速常数 β の取り方によっては収束しないこともある。また β は経験的にしか決められない。	左と同じ 左と同じ
一 般 性	比較的小規模で荷重条件のいものにてきる。	大規模な問題により	大規模な問題により。

Class1: blockの境界に接しない節点

Class2: blockの境界に接し、属しているblock番号より少ない番号のblockにだけ接している節点

Class3: blockの境界に接し、属しているblock番号より少ない番号のblockと大きい番号のblockに接している節点

Class4: blockの境界に接し、属しているblock番号より大きい番号のblockにだけ接している節点

クラス分類は計算機内部で行なわれるが図1-1を分類したものを表1-2に示す。

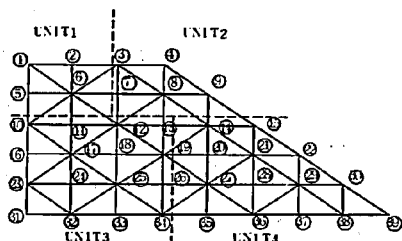


図 1-1

表 1-2

クラス UNIT	1	2	3	4
1	1, 2, 5, 6			3, 7, 10, 11, 12
2	4, 8, 9	3, 7	12, 10, 11	13, 14, 15
3	16, 17, 18, 23, 24, 25, 31, 32, 33	10, 11, 12	13, 14, 15	19, 26, 34
4	20, 21, 22, 27, 28, 29, 30, 35, 36, 37, 38, 39	13, 14, 15 19, 26, 34		

図1-1の構造物の解くべき荷重と節点変位の釣合方程式は

$$\begin{Bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{39} \end{Bmatrix} = \begin{bmatrix} K_{1,1} & \cdots & K_{1,39} \\ \vdots & & \vdots \\ K_{39,1} & \cdots & K_{39,39} \end{bmatrix} \begin{Bmatrix} D_1 \\ D_2 \\ \vdots \\ D_{39} \end{Bmatrix} \quad (1-5)$$

である。上式はクラス分類を考慮して式(1-6)の様にユニットごとに書き換える事が出来る。ここに D_i^j は i 番目のユニットの j クラスに含まれる節点を意味するものとする。荷重項も同様

$$\begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} \text{shaded} & & & \\ & \text{shaded} & & \\ & & \text{shaded} & \\ & & & \text{shaded} \end{bmatrix} \end{matrix} \quad \left\{ \begin{matrix} D_1^1 \\ D_2^1 \\ D_3^1 \\ D_4^1 \\ D_1^2 \\ D_2^2 \\ D_3^2 \\ D_4^2 \\ D_1^3 \\ D_2^3 \\ D_3^3 \\ D_4^3 \\ D_1^4 \\ D_2^4 \\ D_3^4 \\ D_4^4 \end{matrix} \right\} = \left\{ \begin{matrix} P_1^1 \\ P_2^1 \\ P_3^1 \\ P_4^1 \\ P_1^2 \\ P_2^2 \\ P_3^2 \\ P_4^2 \\ P_1^3 \\ P_2^3 \\ P_3^3 \\ P_4^3 \\ P_1^4 \\ P_2^4 \\ P_3^4 \\ P_4^4 \end{matrix} \right\} \quad (1-6)$$

である。表 1-2 を参照すれば、

$$D_1^1 = \begin{bmatrix} D_1 \\ D_2 \\ D_3 \\ D_6 \end{bmatrix}, \quad D_2^2 = \begin{bmatrix} D_3 \\ D_7 \end{bmatrix}$$

である。

いま $(n-1)$ ニットでの釣合方程式を支持節点の部分をも考慮して取り出して考えると(この操作段階を $(n-1)$ ステージということにする)

$$\begin{bmatrix} K_{n-1}^{11} & K_{n-1}^{12} & K_{n-1}^{13} & K_{n-1}^{14} & K_{n-1}^{1C} \\ K_{n-1}^{21} & K_{n-1}^{22} & K_{n-1}^{23} & K_{n-1}^{24} & K_{n-1}^{2C} \\ K_{n-1}^{31} & K_{n-1}^{32} & K_{n-1}^{33} & K_{n-1}^{34} & K_{n-1}^{3C} \\ K_{n-1}^{41} & K_{n-1}^{42} & K_{n-1}^{43} & K_{n-1}^{44} & K_{n-1}^{4C} \\ K_{n-1}^{C1} & K_{n-1}^{C2} & K_{n-1}^{C3} & K_{n-1}^{C4} & K_{n-1}^{CC} \end{bmatrix} \begin{Bmatrix} D_{n-1}^1 \\ D_{n-1}^2 \\ D_{n-1}^3 \\ D_{n-1}^4 \\ D_{n-1}^C \end{Bmatrix} = \begin{Bmatrix} P_{n-1}^1 \\ P_{n-1}^2 \\ P_{n-1}^3 \\ P_{n-1}^4 \\ P_{n-1}^C \end{Bmatrix} \quad \dots\dots\dots (1-7)$$

となりこれを

$$\begin{bmatrix} K_{n-1}^A & K_{n-1}^{AB} & K_{n-1}^{AC} \\ K_{n-1}^{BA} & K_{n-1}^B & K_{n-1}^{BC} \\ K_{n-1}^{CA} & K_{n-1}^{CB} & K_{n-1}^C \end{bmatrix} \begin{Bmatrix} D_{n-1}^A \\ D_{n-1}^B \\ D_{n-1}^C \end{Bmatrix} = \begin{Bmatrix} P_{n-1}^A \\ P_{n-1}^B \\ P_{n-1}^C \end{Bmatrix} \quad \dots\dots\dots (1-8)$$

と簡単に表わす。いま支持部の節点固定されている場合を考える。

すなわち境界条件として $D_{n-1}^C = 0$ と考える。

ただし支持節点で反力を表わす P_{n-1}^C は未知量である。

このとき式 (1-8) は次の 2 組の方程式に分割される。

$$\begin{bmatrix} K_{n-1}^A & K_{n-1}^{AB} \\ K_{n-1}^{BA} & K_{n-1}^B \end{bmatrix} \begin{Bmatrix} D_{n-1}^A \\ D_{n-1}^B \end{Bmatrix} = \begin{Bmatrix} P_{n-1}^A \\ P_{n-1}^B \end{Bmatrix} \quad \dots\dots\dots (1-8)'$$

$$\{K_{n-1}^{CA}\}\{D_{n-1}^A\} + \{K_{n-1}^{CB}\}\{D_{n-1}^B\} = \{P_{n-1}^C\} \quad \dots\dots\dots (1-8)''$$

ここに

A : クラス 1 とクラス 2 の節点部を意味する。

B : クラス 3 とクラス 4 の節点部を意味する。

C : 支持部を意味する。

K_{n-1}^A : $(n-1)$ ステージでのクラス 1 とクラス 2 の剛性マトリクス

K_{n-1}^B : $(n-1)$ ステージのクラス 3 とクラス 4 の剛性マトリクス

K_{n-1}^C : $(n-1)$ ステージでの支持部に関する剛性マトリクス

$K_{n-1}^{AB}, K_{n-1}^{BC}, K_{n-1}^{AC}$: $(n-1)$ ステージでの A と B, B と C および A と C の相関を表わす剛性マトリクス

$K_{n-1}^{BA}, K_{n-1}^{CB}, K_{n-1}^{CA}$: それらの転置マトリクス

P_{n-1}^A, D_{n-1}^A : $(n-1)$ ステージでのクラス 1, クラス 2 の荷重及び変位

P_{n-1}^B, D_{n-1}^B : $(n-1)$ ステージでのクラス 3, クラス 4 の荷重及び変位

P_{n-1}^C, D_{n-1}^C : $(n-1)$ ステージでの支持部節点の荷重及び変位

(II) 式(1-8)' で次のユニットに接続していないクラス1およびクラス2の節点変位を消去する。その結果次元を減じた釣合方程式を得る。これを次の釣合方程式と重ね合わせ最後のステージまで繰返す。

つまり式(1-8)' は

$$\{K_{n-1}^A\}\{D_{n-1}^A\} + \{K_{n-1}^{AB}\}\{D_{n-1}^B\} = \{P_{n-1}^A\} \quad (1-9)$$

$$\{K_{n-1}^{BA}\}\{D_{n-1}^A\} + \{K_{n-1}^B\}\{D_{n-1}^B\} = \{P_{n-1}^B\} \quad (1-10)$$

と書きかえられ(1-9)式から $\{D_{n-1}^A\}$ を消去する。

$$\{D_{n-1}^A\} = \{K_{n-1}^A\}^{-1} [\{P_{n-1}^A\} - \{K_{n-1}^{AB}\}\{D_{n-1}^B\}] \quad (1-11)$$

(1-11)式を(1-10)式に代入して

$$\begin{aligned} & [\{K_{n-1}^B\} - \{K_{n-1}^{BA}\}\{K_{n-1}^A\}^{-1}\{K_{n-1}^{AB}\}]\{D_{n-1}^B\} = \{P_{n-1}^B\} \\ & - \{K_{n-1}^{BA}\}\{K_{n-1}^A\}^{-1}\{P_{n-1}^A\} \quad (1-12) \end{aligned}$$

この式(1-12)を次のユニットnでの節点分類に従い行と列との入れかえを行なうとクラス2および3に関する成分のみとなる。

つまり

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & K_{nR}^{22} & K_{nR}^{23} & 0 \\ 0 & K_{nR}^{32} & K_{nR}^{33} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} D_n^1 \\ D_n^2 \\ D_n^3 \\ D_n^4 \end{Bmatrix} = \begin{Bmatrix} 0 \\ P_{nR}^2 \\ P_{nR}^3 \\ 0 \end{Bmatrix} \quad (1-13)$$

ここに

$$\begin{bmatrix} K_{nR}^{22} & K_{nR}^{23} \\ K_{nR}^{32} & K_{nR}^{33} \end{bmatrix} = [\{K_{n-1}^B\} - \{K_{n-1}^{BA}\}\{K_{n-1}^A\}^{-1}\{K_{n-1}^{AB}\}] \quad (1-14)$$

$$\begin{Bmatrix} P_{nR}^2 \\ P_{nR}^3 \end{Bmatrix} = \{P_{n-1}^B\} - \{K_{n-1}^{BA}\}\{K_{n-1}^A\}^{-1}\{P_{n-1}^A\} \quad (1-15)$$

この(1-13)式に(n-1)ステージと同様にして得られるステージnでの釣合方程式を重ね合わせると

$$\begin{bmatrix} K_n^{11} & K_n^{12} & K_n^{13} & K_n^{14} \\ K_n^{21} & K_n^{22} + K_{nR}^{22} & K_n^{23} + K_{nR}^{23} & K_n^{24} \\ K_n^{31} & K_n^{32} + K_{nR}^{32} & K_n^{33} + K_{nR}^{33} & K_n^{34} \\ K_n^{41} & K_n^{42} & K_n^{43} & K_n^{44} \end{bmatrix} \begin{Bmatrix} D_n^1 \\ D_n^2 \\ D_n^3 \\ D_n^4 \end{Bmatrix} = \begin{Bmatrix} P_n^1 \\ P_n^2 + P_{nR}^2 \\ P_n^3 + P_{nR}^3 \\ P_n^4 \end{Bmatrix} \quad (1-16)$$

または

$$\begin{bmatrix} K_n^A & K_n^{AB} \\ K_n^{BA} & K_n^B \end{bmatrix} \begin{Bmatrix} D_n^A \\ D_n^B \end{Bmatrix} = \begin{Bmatrix} P_n^A \\ P_n^B \end{Bmatrix}$$

が得られる。このような操作を最後のステージ N まで繰返していくと、最後のユニットでは節点はクラス 1 とクラス 2 だけであるから、その節点変位 D_N^A は

$$\{D_N^A\} = [K_N^A]^{-1} \{P_N^A\}$$

より求める事ができる。

- (Ⅳ) 最後のユニット N の節点変位が求まると、いままでのプロセスと逆向きに既知の節点荷重 P_n^A と節点変位 D_n^B を式 (1-11) に代入してクラス 1 およびクラス 2 の未知節点変位を求めていく。そしてステージ n でのクラス A の節点はステージ $(n-1)$ でのクラス B の節点となる。

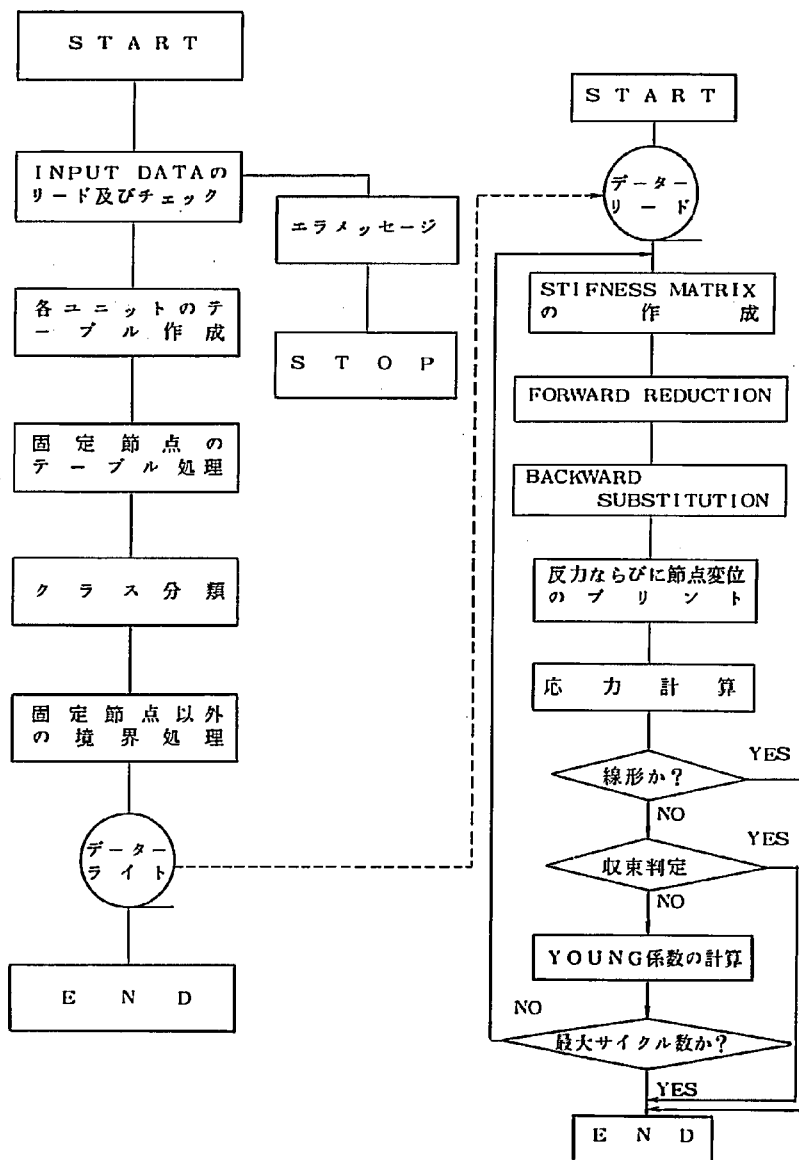
$$\{D_{n-1}^A\} = \{K_{n-1}^A\}^{-1} [\{P_{n-1}^A\} - \{K_{n-1}^{AB}\} \{D_{n-1}^B\}] \quad (1-11)$$

したがってステージ $(n-1)$ で $\{D_{n-1}^A\}$ が求まると支持されている節点反力

$$\{P_{n-1}^C\} = \{K_{n-1}^{CA}\} \{D_{n-1}^A\} + \{K_{n-1}^{CB}\} \{D_{n-1}^B\} \quad (1-8)''$$

は直ちに求められる。

§ 1-2 フローチャート



§ 1-3 入力データ作成の説明

このプログラムはFACOM270-30(65kw, 16ビット)用に作成したもので、使用コア約50数kwである。したがってとりうる変数には下記の制限がある。

つまり 最大節点数 300まで
最大要素数 500まで
最大パターン数 15まで
最大ユニット数 15まで

1ユニットにふくまれる最大節点数(クラス分類1~4までに含まれる合計節点数)
30まで

2 次元入力データシート

No.	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100																													
1	計 算 名																													
2	NP	NE	NU	PS	T	NOP	0	MCYCL	PCYCL	OPT1	LINE	EDP2																		
3	NFXV1 (固定節点番号)										OPT1=1 全部プリント LINE=1 線形計算 LINE=0 非線形計算										P ₀ = 1 平面応力 P ₀ = 0 平面歪									
4	NFXV2 (Xローラ番号)										EDP2, 減衰係数×10										NOP 弾性パターン数 MCYCL 最大サイクル数 PCYCL ポウズサイクル数									
5	NFXV3 (Yローラ番号)																													
6																														
7	NP	XD	YD	NP	XD	YD	(節点座標)																							
8	N	PE	PV	PG	N	PE	PV	PG	NOP (弾性パターン)																					
9	NUE	NU (ユニットに含まれる要素数)																												
10	NE	J	K	NOP	OW=I (自重あり)																									
11	ON	LUN	NU (ユニットに含まれる荷重数及初期変位数)																											
12	NP	LOADX	LOADY	NP	LOADX	LOADY	(荷重及初期変位)																							
13	N1	EDP	EDP-収束判定値 ③ ④は線形計算の時省く。																											
14	e	e	e	e	N1組																									
15	プランカード一枚加える																													

§ 1-4 プ ロ グ ラ ム

(その 1)

C ATANAKA
 C TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
 C A10, A50 = KO ZO BUSTU NAME
 C NP = SO-SETTENSUU
 C NE = SO-ELEMENTSUU
 C NU = SO-UNITSUU
 C PS = HEIMENO-RYOKU-HIZUMI NO KUBESTU O-HIZUMI.1-0-RYOKU
 C T = BANKOO
 C NOP = DANSEI PATTERNSUU

- 10-

C ISPU = SETTENZYOTAI
 C NUE = KAKU UNIT NO ELEMENTSUU
 C NJOI = JOINT-TABLE SAKUSEI ERIA
 C NUNJO = JOINT-TABLE SAKUSEI ERIA
 C NFIX = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C KJOI = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C NFJOI = UNIT NI FUKUMARERU KOTEI SETTEN
 C NFJOI = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C NWJOI = KAKU SETTEN GA ZOKUSHITEIRU
 C NWJO = KAKU SETTEN GA ZOKUSHITEIRU UNIT BANGOTABLE

- 20-

C NCLS = KAKU SETTEN GA ZOKUSHITEIRU
 C NCLS = KAKU SETTEN GA ZOKUSHITEIRU UNIT BANGOTABLE
 C JTBL = CLASS BUNRUITABLE
 C ITBL = CLASS BUNRUITABLE
 C LUN = KAKU UNIT NO KAZYUSUU
 C INOP = KAKU ELEMENT NO PATERNBANGO
 C M1 = ELEMENT BANGO
 C M1 = ELEMENT NO I.J.K YO-SO
 C MJ = ELEMENT NO I.J.K YO-SO
 C MK = ELEMENT NO I.J.K YO-SO

- 30-

C GAMMA = HIZYU
 C EE = YOUNG KEISU
 C V = POISPN HI
 C X = X-ZAHYO TI
 C Y = Y-ZAHYO TI
 C NFXV = KOTEI-ROALER SETTENBANGO
 C XD = ZAHYOTI NO WARK-ERIA OYOB I KAJYU YOMIKOMI ERIA
 C YD = ZAHYOTI NO WARK-ERIA OYOB I KAJYU YOMIKOMI ERIA
 C NTBL = SETTEN ZYOTAI TABLE
 C NWAT = KYOKAI ZYOKEN SYORI TABLE

- 40-

C NWAT = KAKU UNIT NO ZIYUDOSUU
 C NAPAR = KYOKAI SYORI GONO INDEX SUU
 C NNPAT = KYOKAI SYORI GONO INDEX SUU
 C PE = KAKU PATERN NO YOUNGKEISUU-POISONHI-HIZYU
 C PV = KAKU PATERN NO YOUNGKEISUU-POISONHI-HIZYU
 C PG = KAKU PATERN NO YOUNGKEISUU-POISONHI-HIZYU
 C ALP = HI SENKEI KYOKUSEN DATA
 C NEE = KAKU SETTEN NO KESTUGO ELEMENT BANGO TABLE
 C NEE = NEE SAKUSEI NOTAMENO WARK ERIA
 C IWO = WARK ERIA

```

C      NFX1 = KOTEI SETTEN SUU
C      NFX2 = KOTEI SETTEN SUU
C      NFX3 = KOTEI SETTEN SUU
      DIMENSION ISPU(300),NUE(15),NJOI(15),NUNJO(15,30)
      *NFX1(15,25),KJOI(15),NFJOI(15),NWJO(300),NCLS(300,4),ITBL(15,4)
      DIMENSION LUN(15),INOP(500),IWO(3),NEE(300,12),NEW(500)
      *GAMMA(500),EE(500),Y(500),X(500),Y(500),NMT(15)
      *JTBL(15,120),NTABL(4,2),NWT(300,2),NAPAR(60),NNPAR(60)
      DIMENSION MFX1(300),XD(300),YD(300),PE(15),PV(15),PG(15),
      LALP(20,20),M1(500),M1(500),MJ(500),MK(500)
      DOUBLE PRECISION A10,A20,A30,A40,A50
      INTEGER PS,OW,PCYCL,OPT1
      READ(40,99)A10,A20,A30,A40,A50
C
C      PARAMETER READ
C
      99 FORMAT(5A8)
      READ(40,100) NP,NE,NU,PS,T,NOP,ISW,HCYCL,PCYCL,OPT1,LINE,EDP2
      WRITE(50,800) NP,NE,NU,PS,T,NOP
      800 FORMAT(1H0,3HNPS=,15,3X,3HNE=,15,3X,3HNU=,15,3X,3HPS=,15
      1,2X,2HT=,F10.0,3X,4HNOP=,15)
      100 FORMAT(4I5,F10.0,6I5,F10.0)
      IF(NP.LE.300) GO TO 900
      III=1
      GO TO 999
      900 IF(NE.LE.500) GO TO 901
      III=2
      GO TO 999
      901 IF(NU.LE.15) GO TO 902
      III=3
      GO TO 999
      902 READ(40,101)NFX1,NFX2,NFX3
      REWIND 81
      WRITE(50,801)NFX1,NFX2,NFX3
      801 FORMAT(1H0,6HNFXY1=,13,3X,6HNFXY2=,13,3X,6HNFXY3=,13)
      101 FORMAT(3I3)
      DO 50 I=1,NP
      50 ISPU(I)=1
      IF(NFX1)10,9,10
      102 FORMAT(20I4)
      10 READ(40,102) (MFX1(I),I=1,NFX1)
      WRITE(50,802) (MFX1(I),I=1,NFX1)
      802 FORMAT(2X,22H* KOTEI SETTEN BANGO *,20I4)
      DO 51 I=1,NFX1
      NN=MFX1(I)
      ISPU(NN)=4
      51 CONTINUE
      9 IF(NFX2)12,11,12
      12 READ(40,102) (MFX1(I),I=1,NFX2)
      DO 52 I=1,NFX2

```

```

NN=MFX(1)
ISPU(NN)=2
52 CONTINUE
WRITE(50,3000)
3000 FORMAT(1H0,5H** 1 )
11 IF(NFXY3)13,14,13
13 READ(40,102) (MFX(I),I=1,NFXY3)
WRITE(50,803) (MFX(I),I=1,NFXY3)
803 FORMAT(2X,12H* Y-ROLLER *2014)
WRITE(50,4000)
- 100-

4000 FORMAT(2X,20H* SETTEN ZAHYO CHI *)
DO 53 I=1,NFXY3
NN=MFX(I)
ISPU(NN)=3
53 CONTINUE
- 110-

C
C
C
SETTEN ZASHYO DATA READ
14 READ(40,103) (MFX(I),XD(I),YD(I),I=1,NP)
WRITE(50,804) (MFX(I),XD(I),YD(I),I=1,NP)
- 120-

804 FORMAT(4(14,2F8.3))
103 FORMAT(4(14,2F8.3))
DO 54 I=1,NP
NN=MFX(I)
X(NN)=XD(I)
Y(NN)=YD(I)
54 CONTINUE
READ(40,104) (PE(I),PV(I),PG(I),I=1,NOP)
WRITE(50,805) (PE(I),PV(I),PG(I),I=1,NOP)
805 FORMAT(2X,17H* DANSEI PATARN *,3(3X,3F7.0),8X)
- 130-

WRITE(50,3002)
3002 FORMAT(1H0,27H* SETTEN NO SESSYOKU UNIT *)
104 FORMAT(3(3X,3F7.0),8X)
905 NSUM=0
READ(40,106) (NUE(I),I=1,NU)
106 FORMAT(2014)
DO 58 I=1,NU
NSUM=NSUM+NUE(I)
IF(NSUM.EQ.NE) GO TO 906
111=7
- 140-

GO TO 999
906 CONTINUE
DO 621 KK=1,NE
READ(40,107) K1,K2,K3,K4,K5
NEW(K2)=NEW(K2)+1
NEW(K3)=NEW(K3)+1
NEW(K4)=NEW(K4)+1
NN1=NEW(K2)
NN2=NEW(K3)
NN3=NEW(K4)

```

```

- 150-
NEE(K2,NN1)=K1
NEE(K3,NN2)=K1
NEE(K4,NN3)=K1
MI(KK)=K1
MJ(KK)=K2
MK(KK)=K3
INOP(KK)=K5
IF(0.LE.K2.AND.K2.LT.NP) GO TO 621
III=4

- 160-
GO TO 999
IF(0.LE.K3.AND.K3.LT.NP) GO TO 621
III=5
GO TO 999
IF(0.LE.K4.AND.K4.LT.NP) GO TO 621
III=6
GO TO 999
621 CONTINUE
WRITE(50,3510) ((NEE(L1,L2),L2=1,12),L1=1,NP)
3510 FORMAT(1H ,12I5)

- 170-
WRITE(50,4001)
4001 FORMAT(1H0,11H* ELEMENT *)
WRITE(50,107) (MI(I),MJ(I),MK(I),INOP(I),I=1,NE)
WRITE(50,4002)
4002 FORMAT(1H0,24H* NUNJO AND NJOI TABLE *)
107 FORMAT(20I4)
DO 59 I=1,NE
NNN=INOP(I)
GAMMA(I)=PG(NNN)
EE(I)=PE(NNN)

- 180-
V(I)=PV(NNN)
59 CONTINUE
C
C TABLE=1 SAKUSEI
C
DO 60 I=1,NU
60 NJOI(I)=0
IEND=0
ISTA=1
DO 61 IUNIT=1,NU

- 190-
IEND=IEND+NUE(IUNIT)
DO 62 II=ISTA,IEND
IF(0.LT.NJOI(IUNIT)) GO TO 90
NUNJO(IUNIT,1)=MI(ISTA)
NUNJO(IUNIT,2)=MJ(ISTA)
NUNJO(IUNIT,3)=MK(ISTA)
NJOI(IUNIT)=3
GO TO 62
90 IWO(1)=MI(II)
IWO(2)=MJ(II)

```

```

- 200-
      IWO(3)=MK(I1)
      DO 64 IB=1,3
      IN=IWO(IB)
      IKE=NJOI(IUNIT)
      DO 65 IP=1,IKE
      IF(IN.EQ.NUNJO(IUNIT,IP))GO TO 64
65  CONTINUE
      IKE=IKE+1
      NUNJO(IUNIT,IKE)=IN
      NJOI(IUNIT)=IKE
- 210-
64  CONTINUE
62  CONTINUE
61  ISTA=ISTA+NUE(IUNIT)
C
C      TABLE-1 OUT PUT (TEST)
C
      DO 400 I=1,NU
400  WRITE(50,500) (NUNJO(I,J),J=1,30)
      WRITE(50,501) (NJOI(I),I=1,NU)
500  FORMAT(1X,10I4)
- 220-
501  FORMAT(///,1X,4I4,/,1H1)
      NZ=NZ+1
      NWAT(ISTA,NE)=NZ
C
C      TABLE-2 SAKUSEI
C
      WRITE(50,4004)
4004  FORMAT(1H0,24H* NUNJO AND KJOI TABLE *)
      DO 66 I=1,NU
      NX=0
- 230-
      IT=0
      JEND=NJOI(I)
      DO 67 J=1,JEND
      II=NUNJO(I,J)
      IK=ISPU(II)
      IF(IK.NE.4) GO TO 91
      NX=NX+1
      NFIX(I,NX)=II
      NUNJO(I,J)=0
      GO TO 67
- 240-
91  IT=IT+1
      NUNJO(I,IT)=II
67  CONTINUE
      KJOI(I)=IT
66  NFJOI(I)=NX
      DO 401 I=1,NU
401  WRITE(50,500) (NUNJO(I,J),J=1,30)
      WRITE(50,501) (KJOI(I),I=1,NU)
      WRITE(50,4040)
4040  FORMAT(1H0,24H* NFIX AND NFJOI TABLE *)

```

```

- 250-
DO 402 I=1,NU
402 WRITE(50,500)(NFIK(I,J),J=1,25)
WRITE(50,501)(NFJO(I),I=1,NU)
C
C
C
C
C
C
TABLE-2 OUT PUT(TESt)

TABLE-3 SAKUSEI
WRITE(50,4003)
- 260-
4003 FORMAT(1H0,23H* NCLS AND NWJO TABLE *)
DO 68 I=1,NP
68 NWJO(I)=0
DO 69 I=1,NU
DO 69 J=1,4
69 ITBL(I,J)=0
DO 70 IK=1,NU
JP=KJOI(IK)
DO 70 JK=1,JP
II=NUNJO(IK,JK)
- 270-
IF(NWJO(II).NE.0) GO TO 92
NWJO(II)=1
NCLS(II,1)=IK
GO TO 94
92 NWJO(II)=NWJO(II)+1
IKK=NWJO(II)
NCLS(II,IKK)=IK
94 CONTINUE
70 CONTINUE
DO 403 I=1,NP
- 280-
403 WRITE(50,502)(NCLS(I,J),J=1,4),NWJO(I)
502 FORMAT(1X,4I3,5X,I2)
C
C
C
C
C
C
TABLE-3 OUTPUT(TESt)

TABLE-4 SAKUSEI
DO 71 II=1,NP
L=NWJO(II)
- 290-
IF(L.NE.1) GO TO 71
KK=NCLS(II,1)
ITBL(KK,1)=ITBL(KK,1)+1
KE=ITBL(KK,1)
JTBL(KK,KE)=II
71 CONTINUE
DO 41 KI=1,NU
JP=KJOI(KI)
DO 72 JI=1,JP
II=NUNJO(KI,JI)

```

```

JK=NWJO(11)
IF(JK,LE,1) GO TO 72
KAT=0
KMA=0
DO 73 LP=1,JK
KP=NCLS(11,LP)
IF(K1,GT,KP) GO TO 76
IF(K1,EQ,KP) GO TO 73
KAT=1
GO TO 73

76 KMA=1
73 CONTINUE
IF(KAT,NE,1,OR,KMA,NE,1) GO TO 150
IC=3
GO TO 151
150 IF(KAT,NE,1) GO TO 152
IC=4
152 IF(KMA,NE,1) GO TO 151
IC=2
151 ITBL(K1,IC)=ITBL(K1,IC)+1

KAT=(IC-1)*30+ITBL(K1,IC)
JTBL(K1,KAT)=11
72 CONTINUE
WRITE(50,4005)
4005 FORMAT(1H0,17H* JTBL AND ITBL *)
WRITE(50,504)(JTBL(K1,J),J=1,120)
WRITE(50,506)(ITBL(K1,J),J=1,4)
IF(K1,EQ,1) GO TO 41
KKP=K1-1
DO 155 LA=3,4

KEN=ITBL(KKP,LA)
IF(KEN,EQ,0) GO TO 155
LPP=30*(LA-1)
DO 156 LB=1,KEN
KZ=LPP+LB
IN=JTBL(KKP,KZ)
DO 157 LC=2,3
LCN=ITBL(K1,LC)
IF(LCN,EQ,0) GO TO 157
NPA=30*(LC-1)

DO 159 LD=1,LCN
K3=NPA+LD
IF(IN,EQ,JTBL(K1,K3)) GO TO 156
159 CONTINUE
157 CONTINUE
NW=NWJO(IN)
NAE=0
NAT=0
DO 160 K5=1,NW
ME=NCLS(IN,K5)

```

- 300-

- 310-

- 320-

- 330-

- 340-

```

- 350-
      IF(ME.LT.KI) NAE=1
      IF(ME.GT.KI) NAT=1
160  CONTINUE
      IF(NAE.EQ.1) GO TO 910
      III=11
      GO TO 999
910  KF=2
      IF(NAT.EQ.1) KF=3
      ITBL(KI,KF)=ITBL(KI,KF)+1
      NF=ITBL(KI,KF)
- 360-
      N44=30*(KF-1)+NF
      JTBL(KI,N44)=IN
156  CONTINUE
155  CONTINUE
41  CONTINUE
C
C
      DO 161 NA=1,NU
      NTO=0
      DO 162 NB=1,4
- 370-
162  NTO=NTO+ITBL(NA,NB)
      IF(NTO.LE.30) GO TO 161
      III=12
      GO TO 999
161  CONTINUE
C
C
      TABLE=4 OUT PUT(TEST)
C
      WRITE(50,503)
- 380-
503  FORMAT(1H1)
      DO 404 I=1,NU
      WRITE(50,504)(JTBL(I,J),J=1,120)
404  WRITE(50,505)
504  FORMAT(1X,30I4)
505  FORMAT(/)
      DO 405 I=1,NU
405  WRITE(50,506)(ITBL(I,J),J=1,4)
506  FORMAT(1X,4I4)
C
      ***M/T READ-1***
- 390-
      WRITE (81) A10,A20,A30,A40,A50
      WRITE(81) NP,NE,NU,PS,T,NOP,ISWI,MCYCL,PCYCL,OPT1,LINE,EDP2
      WRITE (81) (ISPU(I),I=1,NP)
      WRITE (81) (X(I),I=1,NP)
      WRITE (81) (Y(I),I=1,NP)
      WRITE(81) (NUE(I),I=1,NU)
      WRITE(81) NEE
      DO 601 I=1,NE
      L1=M1(I)
      L2=M1(I)

```

```

L3=MJ(1)
L4=MK(1)
ICON=INOP(1)
G=GAMMA(1)
E=EE(1)
VV=V(1)
WRITE(81) L1,L2,L3,L4,E,VV,G,ICON
601 CONTINUE
WRITE(81) ((JTBL(I,J),I=1,NU),J=1,120)
WRITE(81) ((ITBL(I,J),I=1,NU),J=1,4)

```

- 400-

```

C
C KYOKAIJYOKEN SHORI NO TABLE SAKUSEI
DO 163 KT=1,4
DO 163 LAA=1,2
163 NTABL(KT,LAA)=0
NTABL(2,2)=-1
NTABL(3,1)=-1
DO 164 II=1,NU
DO 165 JK=1,NP
DO 165 LK=1,2

```

- 410-

```

165 NWAT(JK,LK)=0
NZ=0
NZZ=0
DO 166 JJ=1,4
NK=ITBL(II,JJ)
IF(NK.EQ.0) GO TO 166
MM=30*(JJ-1)
DO 167 KK=1,NK
LA=MM+KK
INTA=JTBL(II,LA)

```

- 420-

```

NLP=ISPU(INTA)
DO 168 NE=1,2
KE=NTABL(NLP,NE)
IF(KE.EQ.-1) GO TO 169
NZ=NZ+1
NWAT(INTA*NE)=NZ
NAPAR(NZ)=INTA*10+NE
GO TO 168
169 NZZ=NZZ+1
NNPAR(NZZ)=INTA*10+NE

```

- 430-

```

168 CONTINUE
167 CONTINUE
166 CONTINUE

```

- 440-

C
C
C

```

IF(NFJOI(11).EQ.0) GO TO 170
NJ=NFJOI(11)
DO 171 I=1,NJ
DO 171 J=1,2

```

```

NZZ=NZZ+1
171 NNP(NZZ)=NFI(X(I,I)*10+J
170 NNN=NZ+1
    IF(NZZ.EQ.0) GO TO 172
    DO 173 I=1,NZZ
        NNN=NZ+I
        NA=NNP(I)/10
        NB=NNP(I)-NA*10
        NWAT(NA,NB)=NNN
173 NAPAR(NNN)=NNP(I)
172 CONTINUE

C
C ***M/T READ-2,3***
C WRITE (81) ((NWAT(I,J),I=1,NP),J=1,2)
C WRITE (81) (NAPAR(I),I=1,60)
C
C KYOKAIJYOKEN TABLE OUTPUT(TESt)
C WRITE(50,503)
C WRITE(50,4006)
4006 FORMAT(1H0,32H* NTABL , NWAT AND NAPAR TABLE *)
    WRITE(50,507)((NTABL(I,J),J=1,2),I=1,4)
507 FORMAT(1X,213,/)
    WRITE(50,508)((NWAT(I,J),J=1,2),I=1,NP)
508 FORMAT(1X,214)
    WRITE(50,509)(NAPAR(I),I=1,NNN)
509 FORMAT(1X,2514)
    NMAT(I)=NZ*100+I*ZZ
164 CONTINUE

C
C ***M/T READ-4***
C WRITE (81) (NMAT(I),I=1,NU)
C NNN=0
C READ(40,700) OW,(LUN(I),I=1,NU)
700 FORMAT(13,1514)
    DO 200 I=1,NU
        NN1=LUN(I)/100
        NN2=LUN(I)-NN1*100
        NNN=NNN+NN1*NN2
200 CONTINUE
    IF(NNN.EQ.0) GO TO 300
    READ(40,710) (MFXY(I),XD(I),YD(I),I=1,NNN)
710 FORMAT(4(14,2F8.3))
    WRITE(50,3004)
3004 FORMAT(1H0,5H** 4 )
300 CONTINUE
    IF(LINE.EQ.1) GO TO 5000
    READ(40,720) N1,EDP
    DO 210 I=1,NOP
        KK=2*I-1

```

```

720 FORMAT(15,F10.5)
N10=2*N1
READ(40,730) (X(I),I=1,N10)
DO 211 I=1,N1
  IO=2*I-1
  ALP(I,KK)=X(IO)
  ALP(I,KK+1)=X(IO+1)
211 CONTINUE
730 FORMAT(5(2F8.0))
210 CONTINUE

- 500-

5000 CONTINUE
WRITE(81) OW,NNN,(LUN(I),I=1,NU)
IF(NNN.EQ.0) GO TO 310
WRITE(81) (MFX(I),I=1,NNN)
WRITE(81) (XD(I),I=1,NNN)
WRITE(81) (YD(I),I=1,NNN)
310 CONTINUE
IF(LINE.NE.1) WRITE(81) N1,EDP,ALP
READ(40,556) ICYCL
556 FORMAT(15)

- 510-

WRITE(81) ICYCL
REWIND 81

C
WRITE(50,503)
WRITE(50,4007)
4007 FORMAT(1H0,14H* NMAT TABLE *)
WRITE(50,510) (NMAT(I),I=1,NU)
510 FORMAT(1X,16)
WRITE(50,4010)
4010 FORMAT(1H1////////20X,13(2H**) /20X,13(2H**) ////24X,18HPROGRAM (

- 520-

*A ) END ,////20X,13(2H**)/20X,13(2H**))
WRITE(50,4009)
4009 FORMAT(1H1)
GO TO 1000

C
ERROR MESSAGE OUT
119 FORMAT(10X,5HERROR,12)
999 WRITE(50,119) III
1000 STOP
END

- 530-

```

```

C      BTANAKAI
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
C      ** COMMON /A1/ **
C      NME = KAKU UNIT NO ELEMENT SU
C      NMAT = KAKU UNIT NO JIYUDO SU
C      NWAT = KYOKAI JYOKEN SYORI TABLE
C      LUNI = KAKU UNIT NO KAZU SU
C      ICAB = KAKU UNIT NO JIYUSETTEN SU
C      ICBA = KAKU UNIT NO JIYUSETTEN SU + 1
C      NAPAR = KYOKAI SYORI GO NO INDEX
C
C      ITBL = CLASS BUNRUI TABLE
C      JTBL = CLASS BUNRUI TABLE
C      NEJO = ELEMENT NO I,J,K SETTEN BANGO
C      ** COMMON /A11/ **
C      X = X-ZAHYO
C      Y = Y-ZAHYO
C      WK = ELEMENT STIFFNESS MATRIX
C      USM = KAKU UNIT TANI NO STIFFNESS MATRIX
C      TSM = USM NO WORK AREA
C      FOCE = KAKU UNIT TANI NO KAJUKOH
C
C      WFOCE = FOCE NO WORK AREA
C      ADEF = HENI AND HANRYOKU
C      ** COMMON /A21/ **
C      I10,I20,I30,I40,I50 = KOMZOMBUTU MEISYO
C      NPAGE = PAGE-SU
C      ** COMMON /A3/ **
C      EDP = SYUSOKUCHI
C      WW1 = X-HOKOH NO ZEN KAJU
C      WW2 = Y-HOKOH NO ZEN KAJU
C      NP = SETTEN SU
C
C      NE = ELEMENT SU
C      NU = UNIT SU
C      PS = PLANE STRESS AND STRAIN NO KUBETU
C          PS=0    PLANE STRESS
C          PS=1    PLANE STRAIN
C      T = ITAATU
C      OW = JITYU NO UMU
C      NOP = DANSEI PATTERN SU
C      NNI = NON LINEAR DATA SU
C      ** COMMON /A4/ **
C
C      ALP = NON LINEAR DATA
C      KSW = STOP HANTEI
C      LIKO = SYUSOKU CYCLE SU
C      ** COMMON /A5/ **
C      IISI = SUB.FOW NI OITE ERROR NO BAAI NO STOP HANTEI
C      ** DRUM DIMENSION **
C      DELD = ELEMENT DATA NO KAKUNO
C      DAPAR = NAPAR NO NAIYO O KAKUNO
C      DNWAT = NWAT NO NAIYO O KAKUNO
C      USM2 = USM+FOCE NO NAIYO O KAKUNO

```

- 10-

- 20-

- 30-

- 40-

```

C      DXY = SETTEN ZAHYO KAKUNOH
C      DRDEF = ADEF NO NAIYOH KAKUNOH
C      DDD = N-1 CYCLE NO NAIYOH KAKUNOH
C      DNEP = NEP NO NAIYOH O KAKUNOH
C      ALOAD = SETTEN KAZU O KAKUNOH
COMMON /A1/ NUC(15),NHAT(15),NWAT(300,2),LUNI(15),ICAB(15),
*   ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
*   COMMON /AA1/ X(300),Y(300),WK(6,6),USH(60,60),FOCE(60),TSM(60,60)
*   ,WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
- 50-

COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
COMMON /A5/ IISWI
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
*   ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
*   ,DLOAD(900)
INTEGER OW,PS,PCYCL,OPT1
DOUBLE PRECISION I10,I20,I30,I40,I50
- 60-

ICYCL=1
KSW=0
IISWI=0
NPAGE=0
WRITE (50,1000)
1000 FORMAT(1H1////////,30X,12(3H***),1H*/30X,1H*,35X,1H*/30X,37H* 5
1TRESS AND DEFORMATION ANALYSIS */30X,1H*,35X,1H*/30X,37H* BY
2 FINITE ELEMENT METHOD */30X,1H*,35X,1H*/30X,12(3H***),1H*)
CALL DREAD
IPHAN=PCYCL+ICYCL-1
- 70-

10 CALL STIF
CALL FORW
IF(IISWI.EQ.1) STOP
CALL BACH
CALL REAC
CALL STRESS
ICYCL=ICYCL+1
IF(LINE.EQ.1) GO TO 20
IF(KSW.EQ.0) GO TO 10
20 WRITE(50,4010)
- 80-

4010 FORMAT(1H1////////20X,13(2H**) /20X,13(2H**) ////24X,18HPROGRAM (
*B ) END ,////20X,13(2H**)/20X,13(2H**).)
STOP
END
- 90-

```

```

SUBROUTINE DREAD
BTANAKA2
C
C
TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUE(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,120,130,140,150,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
- 10-

COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DOUBLE PRECISION I10,120,130,140,150
INTEGER OW,PS,PCYCL, OPT1
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* DXY(600),ORDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* DLOAD(900)
DIMENSION NEP(300,12)
EQUIVALENCE (USM,NEP)
REWIND 81
- 20-

REWIND 82
READ (81) I10,120,130,140,150
READ (81) NP,NE,NU,PS,T,NOP,ISWI,MCYCL,PCYCL,OPT1,LINE,EDP2
READ (81) (ISPU(I),I=1,NP)
READ (81) (X(I),I=1,NP)
READ (81) (Y(I),I=1,NP)
READ (81) (NUE(I),I=1,NU)
READ (81) NEP
DRUM WRITE DXY,X,Y
DRUM WRITE DNEP,NEP
- 30-

IF (ISWI.EQ.1) WRITE(50,1060)
1060 FORMAT(1H1)
IF (MCYCL.EQ.0) MCYCL=10
DO 10 I1=1,NE
JT=8*I1-7
READ (81) N1,N2,N3,N4,EE,V,GAMMA,ICON
DRUM WRITE DELD(JT),N1,N2,N3,N4,EE,V,GAMMA,ICON
IF (ISWI.NE.1) GO TO 10
WRITE(50,1031) N1,N2,N3,N4,EE,V,GAMMA,ICON
1031 FORMAT(1H0,4I5,3F10.3,15)
- 40-

10 CONTINUE
READ (81) ((JTBL(J,I),I=1,NU),J=1,120)
READ (81) ((ITBL(I,J),I=1,NU),J=1,4)
IF (ISWI.NE.1) GO TO 90
WRITE(50,1012) (NUE(I),I=1,NU)
WRITE(50,1040) ((JTBL(J,I),I=1,NU),J=1,120)
WRITE(50,1040) ((ITBL(I,J),I=1,NU),J=1,4)
1040 FORMAT(1H ,4I5)
90 CONTINUE
DO 20 I1=1,NU

```

```

110=600*11-599
111=60*11+59
READ (81) ((NWAT(I,J),I=1,NP),J=1,2)
READ (81) (NAPAR(I),I=1,60)
DRUM WRITE DNWAT(110),NWAT
DRUM WRITE DAPAR(111),NAPAR
IF (ISW1.NE.1) GO TO 20
WRITE(50,1050) ((NWAT(I,J),I=1,NP),J=1,2)
WRITE(50,1012) (NAPAR(I),I=1,60)
20 CONTINUE

READ (81) (NMAT(I),I=1,NU)
IF (ISW1.NE.1) GO TO 130
WRITE(50,1012) (NMAT(I),I=1,NU)
130 CONTINUE
1050 FORMAT(1H0,2014)
READ (81) OW,NNN,(LUNI(I),I=1,NU)
IF (NNN.EQ.0) GO TO 40
READ (81) (NAPAR(I),I=1,NNN)
READ (81) (X(I),I=1,NNN)
READ (81) (Y(I),I=1,NNN)

DO 60 III=1,NNN
JT=3*III-2
DRUM WRITE DLOAD(JT),NAPAR(III),X(III),Y(III)
60 CONTINUE
40 CONTINUE
IF (LINE.NE.1) READ(81) NN1,EDP,ALP
READ(81) ICYCL
ICYCL=ICYCL+1
IF (ICYCL.EQ.1) GO TO 150
READ(81) (ADEF(I),I=1,600)

DRUM WRITE DDD,ADEF
150 CONTINUE
NOP2=2*NOP
IF (LINE.EQ.1) GO TO 100
WRITE(50,3000)
GO TO 110
100 WRITE(50,3010)
110 CONTINUE
3000 FORMAT(1H0/38X,23H( NON LINEAR ANALYSIS )//56X,7H(KG.CM))
3010 FORMAT(1H0/38X,23H( LINEAR ANALYSIS )//56X,7H(KG.CM))

WRITE (50,1000)
1000 FORMAT(1H0/3X,16H** INPUT DATA **)
IF (PS.EQ.1) GO TO 30
WRITE (50,1010)
1010 FORMAT(1H0,8X,15H** PLANE STRAIN)
GO TO 35
30 WRITE (50,1011)
1011 FORMAT(1H0,8X,15H** PLANE STRESS)
35 CONTINUE
WRITE (50,1020) T

```

```

1020 FORMAT(1H0.8X,14H** THICKNESS=,F10.2)
      IF (OW.EQ.1) WRITE(50,1030)
1030 FORMAT(1H0.8X,13H** OWN WEIGHT)
      WRITE(50,1001) NP,NE,NU,NOP
1001 FORMAT(1H0//11X,19HNUMBER OF JOINT =,I5//11X,19HNUMBER OF ELEMEN
      *T =,I5//11X,19HNUMBER OF UNIT =,I5//11X,19HNUMBER OF PATTERN =,
      *I5)
1012 FORMAT(1H0,20I5)
      CALL PAGE
      IF(LINE.EQ.1) GO TO 120
- 100-

      IF(NOP.GT.5) NOP2=10
      DO 50 I=1,NN1
      WRITE(50,2020) (ALP(I,J),ALP(I,J+1),J=1,NOP2,2)
2020 FORMAT(1H,5(2F10.5,3X))
      50 CONTINUE
      IF(NOP.LE.5) GO TO 70
      NOP2=2*NOP
      WRITE(50,2030)
2030 FORMAT(1H0)
      DO 80 I=1,NN1
- 110-

      WRITE(50,2020) (ALP(I,J),ALP(I,J+1),J=11,NOP2,2)
      80 CONTINUE
      70 CONTINUE
      120 CONTINUE
      WRITE (50,2040) EDP2
2040 FORMAT(1H0,15HDAMPING FACTOR=,F15.7)
      RETURN
      END
- 120-

```

```

SUBROUTINE STIF
BTANAKA3
C
C TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISW1,LINE
- 10-

COMMON /A6/ ICYCL,HCYCL,PCYCL,OPT1,IPHAN,NNN
INTEGER OW,PS,PCYCL,OPT1
DOUBLE PRECISION I10,I20,I30,I40,I50
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* ,DLOAD(900)
DIMENSION DEF(300,2)
EQUIVALENCE (ADEF,DEF)
DRUM READ DXY,X,Y
REWIND 82
- 20-

REWIND 81
WW1=0.
WW2=0.
ICOU=0
KCOU=0
NEL=0
DO 180 I1=1,2
DO 180 I2=1,NP
DEF(I2,I1)=0.
180 CONTINUE
- 30-

1040 FORMAT(1H0,2X,2HNE,5X,2HIE,2X,2HJE,2X,2HKE,3X,5HGAMMA,3X,8HYOUNG-M
* ,3X,10HPOISSONS-R,4X,8HYOUNG-P ,/)
ICOU1=46
DO 10 K0=1,NU
IF(ICYCL,EQ,1.OR,OPT1,EQ,1) GO TO 301
GO TO 300
301 CONTINUE
CALL PAGE
WRITE(50,1050) K0
WRITE(50,1040)
- 40-

1050 FORMAT(1H0,5H*UNIT,13,2H *)
300 CONTINUE
NEN=NUC(K0)
NW=NMAT(K0)/100
JEN=NMAT(K0)-NW*100
NEN=NW+JEN
DO 20 J1=1,NEN
FOCE(J1)=0.
WFOCE(J1)=0.
20 CONTINUE

```

- 50-

```

I1=K0*600-599
DRUM READ DNWAT(I1)*NWAT
DO 30 K1=1,NEN
DO 30 K2=1,NEN
USM(K1,K2)=0.
30 CONTINUE
DO 40 K3=1,NEND
NEL=NEL+1
JT=B*NEL-7
DRUM READ DELD(JT)*NOEL*NEJO*EE*V*GAMMA*ICON

```

- 60-

```

IF (ICYCL*EQ*1*OR*OPT1*EQ*1) GO TO 311
GO TO 310
311 CONTINUE
WRITE(50,1060) NOEL*NEJO*GAMMA*EE*V*ICON
1060 FORMAT(1H,14,3X,3I4,F8,3,F10,1,F11,3,8X,I2)
310 CONTINUE
I1=NEJO(1)
JJ=NEJO(2)
KK=NEJO(3)
XI=X(I1)

```

- 70-

```

XJ=X(JJ)
XK=X(KK)
YI=Y(I1)
YJ=Y(JJ)
YK=Y(KK)
IF (PS*EQ*1) GO TO 60
C1=EE*(1.0-V)/((1.0+V)*(1.0-2.0*V))
C2=V/(1.0-V)
GO TO 50
60 CONTINUE

```

- 80-

```

C1=EE/(1.0-V*V)
C2=V
50 CONTINUE
F1=(1.0-C2)*0.5
F2=(1.0+C2)*0.5
CO=C1*T/(2.0*((XJ-XI)*YK+(XI-XK)*YJ+(XK-XJ)*YI))
XI=XI-XJ
XJM=XJ-XK
XHI=XK-XI
YIJ=YI-YJ

```

- 90-

```

YJM=YJ-YK
YHI=YK-YI
WK(1,1)=(F1*XJM*XJH+YJM*YJH)*CO
WK(1,2)=(-F2*XJM*YJH)*CO
WK(1,3)=(F1*XJM*XHI+YJM*YHI)*CO
WK(1,4)=(-F1*XJM*YHI+C2*XHI*YJH)*CO
WK(1,5)=(F1*XIJ*XJH+YIJ*YJH)*CO
WK(1,6)=(-F1*XJM*YIJ+C2*XIJ*YJH)*CO
WK(2,2)=(XJM*XJH+F1*YJM*YJH)*CO
WK(2,3)=(-F1*XHI*YJH+C2*XJH*YHI)*CO

```

- 100-

```
WK(2,4)=(XJM*YMI+F1*YJM*YMI)*CO
WK(2,5)=(-F1*XIJ*YJM+C2*XJM*YIJ)*CO
WK(2,6)=(XIJ*XJM+F1*YIJ*YJM)*CO
WK(3,3)=(F1*XMI*YMI+YMI*YMI)*CO
WK(3,4)=(-F2*XMI*YMI)*CO
WK(3,5)=(F1*XIJ*YMI+YIJ*YMI)*CO
WK(3,6)=(-F1*XMI*YIJ+C2*XIJ*YMI)*CO
WK(4,4)=(XMI*YMI+F1*YMI*YMI)*CO
WK(4,5)=(-F1*XIJ*YMI+C2*XMI*YIJ)*CO
WK(4,6)=(XIJ*YMI+F1*YIJ*YMI)*CO
```

- 110-

```
WK(5,5)=(F1*XIJ*XIJ*YIJ*YIJ)*CO
WK(5,6)=(-F2*XIJ*YIJ)*CO
WK(6,6)=(XIJ*XIJ*YIJ*YIJ)*CO
DO 70 K4=1,5
KK4=K4+1
DO 70 K5=KK4,6
WK(K5,K4)=WK(K4,K5)
```

```
70 CONTINUE
C WRITE(50,4010) ((WK(I1,JJ),I1=1,6),JJ=1,6)
4010 FORMAT(1H,6F12.5)
```

- 120-

```
DO 80 K6=1,3
N11=NEJO(K6)
KR=2*K6-2
DO 90 K7=1,3
N33=NEJO(K7)
JR=2*K7-2
DO 100 K8=1,2
KK=NWAT(N11,K8)
KE=KR+K8
DO 110 K9=1,2
```

- 130-

```
KL=NWAT(N33,K9)
NO=JR+K9
USH(KK,KL)=USH(KK,KL)+WK(KE,NO)
110 CONTINUE
100 CONTINUE
90 CONTINUE
80 CONTINUE
IF(OW.EQ.0) GO TO 150
WE=-0.001*GAHMA*T*((XJ-XI)*(YK-YI)-(XK-XI)*(YJ-YI))/6.
DO 120 K10=1,3
```

- 140-

```
N33=NEJO(K10)
IR=NWAT(N33,2)
FOCE(IR)=FOCE(IR)+WE
DEF(N33,2)=DEF(N33,2)+WE
120 CONTINUE
150 CONTINUE
40 CONTINUE
KLU2=LUNI(K0)/100
KLU1=LUNI(K0)-KLU2*100
IF(KLU1.EQ.0) GO TO 130
```

```

DO 140 K11=1,KLU1
ICOU=ICOU+1
JT=3*ICOU-2
DRUM READ DLOAD(JT),LNO,XL,YL
NP1=NWAT(LNO,1)
FOCE(NP1)=FOCE(NP1)+XL
NP1=NWAT(LNO,2)
FOCE(NP1)=FOCE(NP1)+YL
DEF(LNO,1)=DEF(LNO,1)+XL
DEF(LNO,2)=DEF(LNO,2)+YL
- 150-

140 CONTINUE
130 CONTINUE
IF(KLU2,EQ,0) GO TO 190
DO 200 K11=1,KLU2
ICOU=ICOU+1
JT=3*ICOU-2
DRUM READ DLOAD(JT),LNO,XL,YL
NP1=NWAT(LNO,1)
WFOCE(NP1)=WFOCE(NP1)+XL
NP1=NWAT(LNO,2)
WFOCE(NP1)=WFOCE(NP1)+YL
- 160-

200 CONTINUE
CALL INDI(NEN,NW)
190 CONTINUE
IF(ICYCL,EQ,1.OR.OPT1,EQ,1) GO TO 321
GO TO 320
321 CONTINUE
WRITE(50,1090) K0
1090 FORMAT(1H0//5X,10H** LOAD **,5X,6HUNIT =,13//5X,2HNP,10X,5HX(KG),
*10X,5HY(KG)/)
- 170-

320 CONTINUE
DO 170 K11=1,NP
IF(NWAT(K11,1),EQ,0) GO TO 170
KRR=NWAT(K11,1)
KRL=NWAT(K11,2)
W1=FOCE(KRR)
W2=FOCE(KRL)
WW1=WW1+W1
WW2=WW2+W2
IF(ICYCL,EQ,1.OR.OPT1,EQ,1) GO TO 331
- 180-

GO TO 330
331 CONTINUE
WRITE(50,1070) K11,W1,W2
1070 FORMAT(1H ,3X,13,2F15.5)
330 CONTINUE
170 CONTINUE
WRITE(81) ((USM(J1,J2),J1=1,60),J2=1,60)
WRITE(81) (FOCE(J1),J1=1,60)
10 CONTINUE
IF(ICYCL,EQ,1.OR.OPT1,EQ,1) WRITE(50,1080) WW1,WW2
- 190-

```

1080 FORMAT(1H0/7X+6F15.5)
DRUM WRITE ALOAD+ADEF
RETURN
END

- 200-

```

SURROUTINE MTWT
BTANAKAA
C
C
TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISW1,LINE

COMMON /A5/ IISW1
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* DLOAD(900)
INTEGER OW,PS,PCYCL,OPT1
DIMENSION NEP(300,12)
EQUIVALENCE(USH,NEP)
DOUBLE PRECISION I10,I20,I30,I40,I50

IPHAN=IPHAN+PCYCL
REWIND 81
WRITE(81) I10,I20,I30,I40,I50
WRITE(81) NP,NE,NU,PS,T,NOP,ISW1,MCYCL,PCYCL,OPT1,LINE,EDP2
WRITE(81) (ISPU(I),I=1,NP)
DRUM READ DXY,X,Y
DRUM READ DNEP,NEP
WRITE(81) (X(I),I=1,NP)
WRITE(81) (Y(I),I=1,NP)
WRITE(81) (NUC(I),I=1,NU)

WRITE(81) NEP
DO 10 I1=1,NE
JT=8*I1-7
DRUM READ DELD(JT),N1,N2,N3,N4,EE,V,GAMMA,ICON
WRITE(81) N1,N2,N3,N4,EE,V,GAMMA,ICON
10 CONTINUE
WRITE(81) ((JTBL(J,I),I=1,NU),J=1,120)
WRITE(81) ((ITBL(I,J),I=1,NU),J=1,4)
DO 20 I1=1,NU
I10=600*I1-599

I11=60*I1-59
DRUM READ DNWAT(I10),NWAT
DRUM READ DAPAR(I11),NAPAR
WRITE(81) ((NWAT(I,J),I=1,NP),J=1,2)
WRITE(81) (NAPAR(I),I=1,60)
20 CONTINUE
WRITE(81) (NMAT(I),I=1,NU)
WRITE(81) OW,NNN,(LUNI(I),I=1,NU)
IF(NNN.EQ.0) GO TO 40
DO 30 I1=1,NNN

```

```

JT=3*II-2
DRUM READ DLOAD(JT),NAPAR(II),X(II),Y(II)
30 CONTINUE
WRITE(81) (NAPAR(I),I=1,NNN)
WRITE(81) (X(I),I=1,NNN)
WRITE(81) (Y(I),I=1,NNN)
40 CONTINUE
IF(LINE.NE.1) WRITE(81) NN1,EDP,ALP
WRITE(81) ICYCL
DRUM READ DDD,ADEF
WRITE(81) (ADEF(I),I=1,600)
REWIND 81
PAUSE 555
RETURN
END

```

- 50-

- 60-

```

C      SURROUTINE FORM
C      BTANAKAS
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NHAT(15),NWAT(300,2),LUNI(15),ICAB(15),
*      ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
*      ,WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,I,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,TSWI,LINE

COMMON /A5/ IISWI
DIMENSION NWOK(60)
DOUBLE PRECISION I10,I20,I30,I40,I50
INTEGER OW,PS
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
*      ,DXY(600),URDEF(600),DDD(600),DNEP(3600),ALOAD(600)
DIMENSION DEF(300,2)
EQUIVALENCE (ADEF,DEF)
IF (ISWI,NE,1) GO TO 130
DO 8010 I11=1,NU
- 10-

      KO=600*I11-599
      K1=60*I11-59
      DRUM READ DNWAT(KO),NWAT
      DRUM READ DAPAR(K1),NAPAR
      WRITE(50,6010) ((NWAT(LL1,LL2),LL2=1,2),LL1=1,NP)
      WRITE(50,6020) (NAPAR(LL1),LL1=1,60)
8010 CONTINUE
130 CONTINUE
      REWIND 81
      DO 120 I1=1,NU
- 20-

      ICAP(I1)=0
      ICBA(I1)=0
120 CONTINUE
      DO 9999 I1=1,NU
      KO=600*I1-599
7010 FORMAT(1H0,3H50=,13)
      DRUM READ DNWAT(KO),NWAT
      IF (ISWI,NE,1) GO TO 140
      WRITE(50,6010) ((NWAT(LL1,LL2),LL2=1,2),LL1=1,NP)
6010 FORMAT(1H ,2I5)
- 30-

140 CONTINUE
      READ(81) ((USM(J1,J2),J1=1,60),J2=1,60)
      READ(81) (FOCE(J1),J1=1,60)
      J2=NHAT(I1)/100
      IF (I1,EQ,1) GO TO 299
      K11=60*I1-119
6020 FORMAT(1H ,20I5)
      DRUM READ DAPAR(K11),NAPAR
      IF (ISWI,NE,1) GO TO 150
      WRITE(50,6020) (NAPAR(LL1),LL1=1,60)
- 40-

```

150 CONTINUE
 KLL=ICBA(II-1)
 KAI=NMAT(II-1)/100
 IF(KAI.EQ.ICAB(II-1)) GO TO 6134
 DO 2503 NA=KLL,KAI
 K93=NAPAR(NA)
 KJ=K93/10
 KC=K93-KJ*10
 IND=NWAT(KJ,KC)
 NWOK(NA)=IND

2503 CONTINUE
 DO 31 K1=KLL,KAI
 K71=NWOK(K1)
 DO 33 K4=KLL,KAI
 K6=NWOK(K4)
 USM(K71,K6)=USM(K71,K6)+TSM(K1,K4)
 33 CONTINUE
 FOCE(K71)=FOCE(K71)+WFOCE(K1)
 31 CONTINUE
 6134 CONTINUE

299 CONTINUE
 IF(ITBL(II,2).EQ.0) GO TO 63
 NK=30+ITBL(II,2)
 NA=JTBL(NK,II)
 DO 61 K1=1,2
 KP=3-K1
 NI=NWAT(NA,KP)
 IF(NI.GT.0.AND.NI.LE.J2) GO TO 62
 61 CONTINUE
 WRITE(50,2310) II,NI,J2

IISWI=1
 GO TO 8000

63 CONTINUE
 NK=ITBL(II,1)
 NA=JTBL(NK,II)
 DO 71 K1=1,2
 KP=3-K1
 NI=NWAT(NA,KP)
 IF(NI.GT.0.AND.NI.LE.J2) GO TO 62
 71 CONTINUE
 WRITE(50,2310) II,NI,J2
 IISWI=1
 GO TO 8000

62 CONTINUE
 ICAR(II)=NI
 IF(II.EQ.NU) GO TO 2616
 IF(ITBL(II,3).EQ.0) GO TO 73
 NA=JTBL(61,II)
 DO 81 K1=1,2
 NI=NWAT(NA,K1)

```

IF(NI.GT.0.AND.NI.LE.J2) GO TO 83
81 CONTINUE
WRITE(50,2310) I1,NI,J2
IISW1=1
GO TO 8000
73 CONTINUE
NA=JTBL(91,I1)
DO 84 K1=1,2
NI=NWAT(NA,K1)
IF(NI.GT.0.AND.NI.LE.J2) GO TO 83
- 100-

84 CONTINUE
WRITE(50,2310) I1,NI,J2
IISW1=1
GO TO 8000
83 CONTINUE
ICBA(I1)=NI
2616 CONTINUE
NEN=ICAR(I1)
IF(ICBA(I1).EQ.1) GO TO 6771
C *** INVERSE MATRIX KA ***
- 110-
CALL AMINV(NEN)
JIE=J2
JIA=ICBA(I1)
KIE=ICAR(I1)
C *** KA-1*KAR CALCULATION ***
IF(I1.EQ.NU) GO TO 2201
DO 96 KI=1,KIE
DO 93 JI=JIA,JIE
TT=0.
DO 94 NI=1,KIE
- 120-
TT=TT+USH(KI,NI)*USH(NI,JI)
94 CONTINUE
TSM(KI,JI)=TT
93 CONTINUE
96 CONTINUE
2201 CONTINUE
DO 95 NA=1,KIE
WFOCE(NA)=0.
TT=0.
DO 97 IA=1,KIE
- 130-
TT=TT+FOCE(IA)*USH(NA,IA)
97 CONTINUE
WFOCE(NA)=TT
95 CONTINUE
7020 FORMAT(1H0,3H3=,15)
WRITE(82) (#FOCE(I),I=1,60)
WRITE(82) ((TSM(I1,I2),I1=1,60),I2=1,60)
IF(I1.EQ.NU) GO TO 9999
GO TO 6772
6771 CONTINUE
- 140-

```

```

      B1=0.
      WRITE(82) B1
6772 CONTINUE
      DO 101 JJ=JIA,JIE
      DO 101 KK=JIA,JIE
      TSM(JJ,KK)=USM(JJ,KK)
101 CONTINUE
      DO 105 JJ=JIA,JIE
      DO 105 KK=JIA,JIE
      TT=0.

      DO 104 NI=1,KIE
      TT=TT+USM(JJ,NI)*TSM(NI,KK)
104 CONTINUE
      TSM(JJ,KK)=TSM(JJ,KK)-TT
103 CONTINUE
105 CONTINUE
      DO 110 KK=JIA,JIE
      WFOCE(KK)=FOCE(KK)
110 CONTINUE
      DO 112 JT=JIA,JIE

      TT=0.
      DO 113 KT=1,KIE
      TT=TT+USM(JT,KT)*WFOCE(KT)
113 CONTINUE
      WFOCE(JT)=WFOCE(JT)-TT
112 CONTINUE
9999 CONTINUE
2310 FORMAT(1H1,5X,8HERROR 10,315)
8000 CONTINUE
      RETURN

      END

```

- 150-

- 160-

- 170-

- 180-

```

SUBROUTINE AMINV(NP3)
BTANAKA6
C
C TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),AM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
DO 2000 K2000=1,NP3
W=AM(K2000,K2000)
WW=ABS(W)

IF(WW.LE.0.00001) GO TO 2000
2001 DO 1700 K1700=1,NP3
1700 AM(K2000,K1700)=AM(K2000,K1700)/W
DO 2200 K2200=1,NP3
IF(K2200-K2000)1800,2200,1800
1800 DO 2100 K2100=1,NP3
IF(K2100-K2000)2900,2100,2900
2900 AM(K2200,K2100)=AM(K2200,K2100)-AM(K2200,K2000)*AM(K2000,K2100)
2100 CONTINUE
2200 CONTINUE

AM(K2000,K2000)=1./W
DO 1930 K1930=1,NP3
IF(K1930-K2000)3000,1930,3000
3000 AM(K1930,K2000)=-AM(K1930,K2000)/W
1930 CONTINUE
2000 CONTINUE
RETURN
END

```

- 10-

- 20-

```

C      SUBROUTINE BACW
C      BTANAKAT
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
*      ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /A1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
*      WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EUP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
- 10-

DOUBLE PRECISION I10,I20,I30,I40,I50
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
*      DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
INTEGER OW,PS
DIMENSION DEF(300,2),WFOCE1(60)
EQUIVALENCE (ADEF,DEF)
DO 10 I11=1,NU
I1=NU-I11+1
KIE=ICAR(I1)
JIA=ICRA(I1)
- 20-

JZ=NMAT(I1)/100
IZ=NMAT(I1)-JZ*100
JIE=JZ
KZ=IZ+JZ
NTK=60*(I1-1)+1
DRUM READ DAPAR(NTK),NAPAR
IF(ISWI,NE,1) GO TO 130
WRITE(50,1040) (ICAB(L1),L1=1,15), (ICBA(L1),L1=1,15)
1040 FORMAT(1H,1015)
WRITE(50,1000) (NAPAR(L1),L1=1,60)
- 30-

1000 FORMAT(1H,2015)
130 CONTINUE
IF(I1,NE,NU) GO TO 20
BACKSPACE 81
BACKSPACE 81
READ(81) ((TSM(J1,J2),J1=1,60),J2=1,60)
READ(81) (WFOCE(J1),J1=1,60)
BACKSPACE 82
BACKSPACE 82
READ(82) (WFOCE1(I1),I1=1,60)
- 40-

READ(82) ((USM(I1,I2),I1=1,60),I2=1,60)
DO 100 K100=1,KIE
WFOCE(K100)=WFOCE1(K100)
100 CONTINUE
GO TO 30
20 CONTINUE
BACKSPACE 81
BACKSPACE 81
BACKSPACE 81
BACKSPACE 81

```

```

READ(81) ((TSM(J1,J2),J1=1,60),J2=1,60)
READ(81) (WFOCE(J1),J1=1,60)
BACKSPACE 82
BACKSPACE 82
BACKSPACE 82
BACKSPACE 82
READ(82) (WFOCE1(I1),I1=1,60)
READ(82) ((USM(I1,I2),I1=1,60),I2=1,60)
1020 FORMAT(1H,5F10.5)
DO 110 K110=1,K1E

DO 120 K120=J1A,J1E
TSM(K110,K120)=USM(K110,K120)
120 CONTINUE
WFOCE(K110)=WFOCE1(K110)
110 CONTINUE
DRUM READ DRDEF,ADEF
1030 FORMAT(1H,2F15.6)
DO 42 NA=1,K1E
TT=0.00
DO 43 KA=J1A,J1E

N3=NAPAR(KA)/10
NXYZ=NAPAR(KA)-N3*10
TT=TT+TSM(NA,KA)*DEF(N3,NXYZ)
43 CONTINUE
N2=NAPAR(NA)/10
NXYZ=NAPAR(NA)-N2*10
DEF(N2,NXYZ)=WFOCE(NA) -TT
42 CONTINUE
IF(IZ.EQ.0) GO TO 410
NNB=J1E+1

DO 73 NB=NNB,KZ
T1=0.00
T2=0.00
DO 72 NA=1,K1E
N4=NAPAR(NA)/10
IXY=NAPAR(NA)-N4*10
T1=T1+TSM(NB,NA)*DEF(N4,IXY)
72 CONTINUE
DO 74 NC=J1A,J1E
N6=NAPAR(NC)/10

KXY=NAPAR(NC)-N6*10
T2=T2+TSM(NB,NC)*DEF(N6,KXY)
74 CONTINUE
TT=T1+T2
N5=NAPAR(NB)/10
JXY=NAPAR(NB)-N5*10
DEF(N5,JXY)=DEF(N5,JXY)+TT
73 CONTINUE
410 CONTINUE
DRUM WRITE DRDEF,ADEF

```

- 50-

- 60-

- 70-

- 80-

- 90-

- 100-

```

41 CONTINUE
GO TO 10
30 CONTINUE
DO 80 I=1,2
DO 80 J=1,NP
DEF(J,I)=0.
80 CONTINUE
DO 52 NA=1,KIE
N4=NAPAR(NA)/10
IXY=NAPAR(NA)-N4*10
DEF(N4,IXY)=WFOCE(NA)
52 CONTINUE
IF(IZ,EQ,0) GO TO 510
NNB=JIE+1
DO 63 NB=NNB,KZ
TT=0.0
DO 62 NA=1,KIE
N4=NAPAR(NA)/10
IXY=NAPAR(NA)-N4*10
TT=TT+TSM(NB,NA)*DEF(N4,IXY)
62 CONTINUE
N5=NAPAR(NB)/10
JXY=NAPAR(NB)-N5*10
DFF(N5,JXY)=DEF(N5,JXY)+TT
63 CONTINUE
510 CONTINUE
DRUM WRITE DRDEF,ADEF
10 CONTINUE
RETURN
END

```

- 110-

- 120-

- 130-

```

SUBROUTINE REAC
C   BTANAKAB
C   TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
C   ... DISPLACEMENT PRINT
C   ... REACTION PRINT
C   ... CAL OF UMBALANCE CHECK
C   ... NUEL , MEMBERS (UNET)
C   ... NPIN , SUPPORT NO HOKO-SHIKI
COMMON /A1/ NUE(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
*   ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
- 10-
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
*   ,WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNH
DIMENSION NPIN(300),RAT(2),DEF(300,2),DDEF(300,2)
INTEGER OW,PS,OPT1
INTEGER OW,PS
DOUBLE PRECISION I10,I20,I30,I40,I50
- 20-
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
*   ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
EQUIVALENCE (ADEF,DEF), (USM,DDEF)
WRAT1=0.
WRAT2=0.
DRUM READ ALOAD,DDEF
DO 32 K=1,NP
KA=ISPU(K)
N8=0
GO TO(33,45,46,54),KA
- 30-
45 CONTINUE
C   X -ROLLER
N8=2
GO TO 33
46 CONTINUE
C   Y -ROLLER
N8=1
GO TO 33
54 CONTINUE
C   PINROLLER
- 40-
N8=12
33 CONTINUE
NPIN(K)=N8
32 CONTINUE
DRUM READ DRDEF,ADEF
IF(ICYCL,EQ.1.OR.KSW,EQ.1.OR.OPT1,EQ.1) GO TO 200
GO TO 210
200 CONTINUE
CALL PAGE
WRITE(50,1020)

```

```

1020 FORMAT(1H0,5X,14H** REACTION **//3X,2HNP,10X,5HX(KG),10X,5HY(KG)
      *
210 CONTINUE
    DO 100 KE=1,NP
      IF(NPIN(KE).EQ.0) GO TO 100
      DO 110 I=1,2
        RAT(I)=0.
110 CONTINUE
    N55=NPIN(KE)/10
    N22=NPIN(KE)-N55*10

    RAT(N22)=DEF(KE,N22)-DDEF(KE,N22)
    DEF(KE,N22)=0.
    IF(N55.EQ.0) GO TO 120
    RAT(N55)=DEF(KE,N55)-DDEF(KE,N55)
    DEF(KE,N55)=0.
120 CONTINUE
    WRAT1=WRAT1+RAT(1)
    WRAT2=WRAT2+RAT(2)
    IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) WRITE(50,1030) KE,RAT(1),
      * RAT(2)

1030 FORMAT(1H ,14,2F15.5)
100 CONTINUE
    IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 220
    GO TO 230
220 CONTINUE
    WRITE(50,1040) WRAT1,WRAT2
1040 FORMAT(1H0,5HTOTAL,F14.5,F15.5)
    WRITE(50,1050) WW1,WW2
1050 FORMAT(1H0,4HLOAD,2F15.5)
    CALL PAGE

    DRUM READ DXY,X,Y
    KK=46
    WRITE(50,1000)

1000 FORMAT(1H0//5X,18H** DISPLACEMENT **//21X,10HCOORDINATE,25X,
      * 12HDISPLACEMENT//4X,2HNP,13X,5HX(CM),10X,5HY(CM),8X,10HDISP=X(C
      * M),5X,10HDISP=Y(CM),11X,7HX-D(CM),8X,7HY-D(CM)/)
    DO 61 ITT=1,NP
      WF1=X(ITT)+DEF(ITT,1)
      WF2=Y(ITT)+DEF(ITT,2)
      IF(ITT.NE.KK) GO TO 65

    CALL PAGE
    WRITE(50,1000)
    KK=KK+45
65 CONTINUE
    WRITE(50,1010) ITT,X(ITT),Y(ITT),DEF(ITT,1),DEF(ITT,2),WF1,WF2
1010 FORMAT(1H ,15,3(3X,2F15.5))
61 CONTINUE
230 CONTINUE
    DRUM WRITE ORDEF,ADEF
10 CONTINUE

RETURN
END

```

```

SUBROUTINE STRESS
C
BTANAKA9
C
TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTB(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE

COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
INTEGER OW,PS,PCYCL,OPT1
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* ,DLOAD(900)
DIMENSION IWW(4),DEF(300,2),AW1(4),AW2(4),EW(7),NOEL1(7)
DIMENSION G1(500),G2(500),G3(500),G4(500),G5(500),G6(500),G7(500),
1 G8(500),DHAN(600),DWHAN(300,2),G22(500)
DOUBLE PRECISION I10,I20,I30,I40,I50

DIMENSION WT(6),SIGMA(3),NEP(300,12)
EQUIVALENCE (ADEF,DEF),(DHAN,DWHAN)
EQUIVALENCE (USM,G1),(USM(501),G2),(USM(1001),G3),(USM(1501),G4),
* (USM(2001),G5),(USM(2501),G6),(USM(3001),G7)
EQUIVALENCE (NWAT,G8),(TSM,NEP)
IF(ICYCL.EQ.MCYCL) KSW=1
DRUM READ DNEP,NEP
NEL=0
NEL1=0
NEL2=0

LIKO=LIKO+1
IF(ICYCL.EQ.1) GO TO 201
DRUM READ DDD,DHAN
3020 FORMAT(1H0,30X,19H** YOUNG MODULUS **,/1X,5X,7(5H NE,9X,1HE))
AMM1=0,
LL=0
CALL PAGE
WRITE(50,3000)
3000 FORMAT(1H0,30X,46H** DIFFERENCE OF DISPLACEMENT (D(I)-D(I-1)) **)
WRITE(50,3001)

3001 FORMAT(1H0,5X,4(5H NP,3X,7HDIFF(X),3X,7HDIFF(Y))/)
204 CONTINUE
DO 200 II=1,4
KLL=II-1
LL=LL+1
IF(LL.GT.NP) GO TO 202
IWW(II)=LL
AW1(II)=DEF(LL,1)-DWHAN(LL,1)
AW2(II)=DEF(LL,2)-DWHAN(LL,2)
AAW1=ABS(AW1(II))

```

```

AAW2=ABS(AW2(I))
IF(AAW1.GT.AMMX1) AMMX1=AAW1
IF(AAW2.GT.AMMX1) AMMX1=AAW2
200 CONTINUE
WRITE(50,3010) (IWW(KK),AW1(KK),AW2(KK),KK=1,4)
IF(LL.NE.NP) GO TO 204
GO TO 203
202 CONTINUE
WRITE(50,3010) (IWW(KK),AW1(KK),AW2(KK),KK=1,KLL)
203 CONTINUE

IF(AMMX1.LE.EDP) KSW=1
3010 FORMAT(1H,5X,4(15,2F10.4))
WRITE(50,3040) AMMX1
3040 FORMAT(1H0//,10X,10HMAX.DIFF =,F10.4)
201 DRUM WRITE DDD,ADEF
DO 10 I1=1,NU
DRUM READ DRDEF,ADEF

C
C
C   HEADING

IF(CYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 300
GO TO 310
300 CONTINUE
CALL PAGE
WRITE(50,5022)
5022 FORMAT(1H0,30X,23H( ELEMENT INFORMATION ))
WRITE(50,2030) I1
2030 FORMAT(1H0,5X,9H** UNIT =,13,3H **)
WRITE(50,2020)
2020 FORMAT(1H0,3H NE,7X,7HSIGMA-X,7X,7HSIGMA-Y,8X,6HTAU-XY,7X,
*
*       7HTAU-MAX,7X,7HSIGMA-1,7X,7HSIGMA-2,9X,5HTHETA,8X,6HX(X/Y
*))/)
310 CONTINUE
JEND=NUE(I1)
DO 20 K1=1,JEND
NEL=NEL+1
JT=8*NEL-7
DRUM READ DELD(JT),NOEL,NEJO,EE,V,GAMMA,ICON
I1=NEJO(1)
JJ=NEJO(2)

KK=NEJO(3)
XI=X(I1)
XJ=X(JJ)
XK=X(KK)
YI=Y(I1)
YJ=Y(JJ)
YK=Y(KK)
IF(PS.EQ.1) GO TO 30
C1=EE*(1.-V)/(1.+V)*(1.-2.*V)
C2=V/(1.-V)

```

- 100-

```

GO TO 40
30 CONTINUE
C1=EE/(1.-V*V)
C2=V
40 CONTINUE
F1=(1.-C2)/2.
C0=C1/((XJ-XI)*YK+(XI-XK)*YJ+(XK-XJ)*YI)
WK(1,1)=(YK-YJ)*C0
WK(1,2)=C2*(XJ-XK)*C0
WK(1,3)=(YI-YK)*C0
WK(1,4)=(-C2)*(XI-XK)*C0
WK(1,5)=(YI-YJ)*(-C0)
WK(1,6)=C2*(XI-XJ)*C0
WK(2,1)=C2*(YK-YJ)*C0
WK(2,2)=(XJ-XK)*C0
WK(2,3)=C2*(YI-YK)*C0
WK(2,4)=(XI-XK)*(-C0)
WK(2,5)=(-C2)*(YI-YJ)*C0
WK(2,6)=(XI-XJ)*C0
WK(3,1)=F1*(XJ-XK)*C0
WK(3,2)=F1*(YK-YJ)*C0
WK(3,3)=(-F1*(XI-XK)*C0)
WK(3,4)=F1*(YI-YK)*C0
WK(3,5)=F1*(XI-XJ)*C0
WK(3,6)=(-F1*(YI-YJ)*C0)
LLL=0
DO 50 N11=1,3
NO=NEJO(N11)
DO 50 M11=1,2
LLL=LLL+1
WT(LLL)=DEF(NO,M11)
50 CONTINUE
DO 60 IA=1,3
TT=0.0
DO 65 JA=1,6
TT=TT+WK(IA,JA)*WT(JA)
65 CONTINUE
SIGMA(IA)=TT
60 CONTINUE
S1=(SIGMA(1)+SIGMA(2))*0.5
S2=SQR(((SIGMA(1)-SIGMA(2))*0.5)**2+SIGMA(3)**2)
SIG1=S1+S2
SIG2=S1-S2
THETA=28.647889*(ATAN(2.0*SIGMA(3)/(SIGMA(1)-SIGMA(2))))
IF (SIGMA(1)-SIGMA(2).LE.0.0) THETA=THETA+90.
AS2=ABS(SIGMA(2))
GK1=9999.9999
IF (AS2.GE.0.00001) GK1=SIGMA(1)/SIGMA(2)
G1(NOEL)=SIGMA(1)
G2(NOEL)=SIGMA(2)

```

- 110-

- 120-

- 130-

- 140-

```

G3(NOEL)=SIGMA(3)
G4(NOEL)=S2
G5(NOEL)=SIG1
G6(NOEL)=SIG2
G7(NOEL)=THETA
G8(NOEL)=GK1
C
C
C
WRITE
IF((ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1)
- 150-
*WRITE(50,1000) NOEL,SIGMA(1),SIGMA(2),SIGMA(3),S2,SIG1,SIG2,THETA,
1 GK1
1000 FORMAT(1H ,13,8F14.6)
20 CONTINUE
10 CONTINUE
KK=51
IF((ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 320
GO TO 330
320 CONTINUE
CALL PAGE
- 160-
WRITE(50,5021)
5021 FORMAT(1H0,30X,33H( MEAN VALUE AROUND NODAL POINT ))
WRITE(50,2000)
2000 FORMAT(1H0,3H NP,7X,8HSIGMA-X ,6X,8HSIGMA-Y ,6X,8HTAU=XY ,6X,8HTA
1U=MAX ,6X,8HSIGMA-1 ,6X,8HSIGMA-2 ,8X,5HTHETA,7X,7HK (X/Y)/)
330 CONTINUE
DO 100 K100=1,NP
SIGMA(1)=0.
SIGMA(2)=0.
SIGMA(3)=0.
- 170-
S2=0.
SIG1=0.
SIG2=0.
THETA=0.
GK1=0.
JJ=0
DO 110 K110=1,8
KK=NEP(K100,K110)
IF(KK.EQ.0) GO TO 120
JJ=JJ+1
- 180-
SIGMA(1)=SIGMA(1)+G1(KK)
SIGMA(2)=SIGMA(2)+G2(KK)
SIGMA(3)=SIGMA(3)+G3(KK)
S2=S2+G4(KK)
SIG1=SIG1+G5(KK)
SIG2=SIG2+G6(KK)
THETA=THETA+G7(KK)
GK1=GK1+G8(KK)
- 190-
110 CONTINUE
120 CONTINUE

```

```

GK2=FLOAT(JJ)
GK2=1./GK2
SIGMA(1)=SIGMA(1)*GK2
SIGMA(2)=SIGMA(2)*GK2
SIGMA(3)=SIGMA(3)*GK2
S2=S2*GK2
SIG1=SIG1*GK2
SIG2=SIG2*GK2
THETA=THETA*GK2
GK1=GK1*GK2
- 200-

IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 340
GO TO 350
340 CONTINUE
IF(K100.NE.KKK) GO TO 130
CALL PAGE
WRITE(50,5021)
WRITE(50,2000)
KKK=KKK+50
130 CONTINUE
WRITE(50,2010) K100,SIGMA(1),SIGMA(2),SIGMA(3),S2,SIG1,SIG2,THETA,
- 210-

* GK1
2010 FORMAT(1H ,13,6F14.6,F14.3,F14.4)
350 CONTINUE
100 CONTINUE
IF(LINE.EQ.1) GO TO 220
5020 FORMAT(1H ,13)
CALL PAGE
WRITE(50,3020)
DO 140 K140=1,NE
JT=R*K140-7
- 220-

DRUM READ DELD(JT),NOEL,NEJO,EE,V,GAMMA,ICON
NEL1=NEL1+1
NEL2=NEL2+1
NOEL1(NEL1)=NOEL
EW(NEL1)=EE
IF(NEL2.EQ.NE) GO TO 361
IF(NEL1.LT.7) GO TO 360
WRITE(50,3030) (NOEL1(I),EW(I),I=1,7)
NEL1=0
GO TO 360
- 230-

3030 FORMAT(1H ,5X,7(15,F10.2))
361 CONTINUE
WRITE(50,3030) (NOEL1(I),EW(I),I=1,NEL1)
360 CONTINUE
KKK=ICON*2-1
NN2=NN1-1
IF(ICYCL.EQ.1) GO TO 480
G2(NOEL)=EDP2*(G2(NOEL)-G22(NOEL))+G22(NOEL)
480 CONTINUE
IF(G2(NOEL).LT.0.0) GO TO 6666
- 240-

```

- 250-

```

DO 150 K150=1,NN2
  IF (ALP(K150,KKK).LE.G2(NOEL).AND,ALP(K150+1,KKK).GE.G2(NOEL)) GO T
10 160
150 CONTINUE
160 R5=(ALP(K150+1,KKK+1)-ALP(K150,KKK+1))/(ALP(K150+1,KKK)
1 -ALP(K150,KKK))
5010 FORMAT(1H ,5E13.4,15)
R6=R5*G2(NOEL)+(R5*(-ALP(K150,KKK))+ALP(K150,KKK+1))
EE=G2(NOEL)/R6
GO TO 6667

```

- 260-

```

6666 EE=0.
6667 CONTINUE
DRUM WRITE DELD(JT)*NOEL*NEJO*EE*V*GAMMA*ICON
140 CONTINUE
220 CONTINUE
C WRITE(50,6500)
C6500 FORMAT(1H1,3H NE,5X,14HDAMPING-STRESS//)
DO 470 K470=1,NE
G22(K470)=G2(K470)
C WRITE(50,6510)

```

- 270-

```

C6510 FORMAT(1H ,I3,F20.5)
470 CONTINUE
IF(KSW.EQ.0) GO TO 400
DRUM READ DDD,ADEF
CALL PAGE
DRUM READ DXY,X,Y
KK=46
WRITE(50,6000)
6000 FORMAT(1H0//5X,18H** DISPLACEMENT **//21X,10HCOORDINATE,25X,
* 12HDISPLACEMENT//4X,2HNP,13X,5HX(CM),10X,5HY(CM),8X,10HDISP-X(C

```

- 280-

```

*M),5X,10HDISP-Y(CM),11X,7HX-D(CM),8X,7HY-D(CM)//)
DO 461 ITT=1,NP
WF1=X(ITT)+DEF(ITT,1)
WF2=Y(ITT)+DEF(ITT,2)
IF(ITT.NE.KK) GO TO 465
CALL PAGE
WRITE(50,6000)
KK=KK+45
465 CONTINUE
WRITE(50,6010) ITT,X(ITT),Y(ITT),DEF(ITT,1),DEF(ITT,2),WF1,WF2

```

- 290-

```

6010 FORMAT(1H ,I5,3(3X,2F15.5))
461 CONTINUE
400 CONTINUE
IF(ICYCL.EQ.1PHAN) CALL MTWT
RETURN
END

```

```

C      SUBROUTINE PAGE
C      BTANAKA10
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A6/ICYCL
DOUBLE PRECISION I10,I20,I30,I40,I50
NPAGE=NPAGE+1
1000 WRITE(50,1000) NPAGE,I10,I20,I30,I40,I50,ICYCL
      FORMAT(1H1/90X,5HPAGE=,I3,/5X,5A8,10X,6HCYCLE=,I3)
      RETURN
END

```

- 10-

```

SUBROUTINE INDI(NEN,NW)
BTAMAK11
C TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
C COMMON /AA1/ X(300),Y(300),WK(5,6),
1   USM(60,60),FOCE(60),ISM(60,60),
2   WFOCE(60),ADEF(600),ISPU(300)
DO 10 I=1,NW
DO 10 I1=1,NEN
FOCE(I)=FOCE(I)+USM(I,I1)*WFOCE(I1)
10 CONTINUE
RETURN
END

```

- 10-

○
§ 1-5 プ ロ グ ラ ム

(その 2)

○

C ASEKO
 C TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
 C A10, A50 = KO ZO BUTSU NAME
 C NP = SO-SETTENSUU
 C NE = SO-ELEMENTSUU
 C NU = SO-UNITSUU
 C PS = HEIMENO-RYOKU-HIZUMI NO KUBESTU O=HIZUMI,1=O-RYOKU
 C T = BANKOO
 C NOP = DANSEI PATTERNSUU
 C ISPU = SETTENZYOTAI - 10-

 C NUE = KAKU UNIT NO ELEMENTSUU
 C NJOI = JOINT-TABLE SAKUSEI ERIA
 C NUNJO = JOINT-TABLE SAKUSEI ERIA
 C NFIX = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C KJOI = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C NFJOI = UNIT NI FUKUMARERU KOTEI SETTEN
 C NFJOI = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C NWJOI = KAKU SETTEN GA ZOKUSHITEIRU
 C NWJO = KAKU SETTEN GA ZOKUSHITEIRU UNIT BANGOTABLE
 C NCLS = KAKU SETTEN GA ZOKUSHITEIRU - 20-

 C NCLS = KAKU SETTEN GA ZOKUSHITEIRU UNIT BANGOTABLE
 C JTBL = CLASS BUNRUITABLE
 C ITRL = CLASS BUNRUITABLE
 C LUN = KAKU UNIT NO KAZYUSUU
 C INOP = KAKU ELEMENT NO PATERNBANGO
 C M1 = ELEMENT BANGO
 C M1 = ELEMENT NO I,J,K YO-SO
 C M2 = ELEMENT NO I,J,K YO-SO
 C M3 = ELEMENT NO I,J,K YO-SO
 C GAMMA = HIZYU - 30-

 C EE = YOUNG KEISU
 C V = POISPN HI
 C X = X-ZAHYO TI
 C Y = Y-ZAHYO TI
 C NFXI = KOTEI-ROALER SETTENBANGO
 C XD = ZAHYOTI NO WARK-ERIA OYOBI KAJYU YOMIKOMI ERIA
 C YD = ZAHYOTI NO WARK-ERIA OYOBI KAJYU YOMIKOMI ERIA
 C NTRL = SETTEN ZYOTAI TABLE
 C NWAT = KYOKAI ZYOKEN SYORI TABLE
 C NWAT = KAKU UNIT NO ZIYUDOSUU - 40-

 C NAPAR = KYOKAI SYORI GONO INDEX SUU
 C NNPAR = KYOKAI SYORI GONO INDEX SUU
 C PE = KAKU PATERN NO YOUNGKEISUU-POISONHI-HIZYU
 C PV = KAKU PATERN NO YOUNGKEISUU-POISONHI-HIZYU
 C PG = KAKU PATERN NO YOUNGKEISUU-POISONHI-HIZYU
 C ALP = HI SENKEI KYOKUSEN DATA
 C NEE = KAKU SETTEN NO KESTUGO ELEMENT BANGO TABLE
 C NEW = NEE SAKUSEI NOTAHENO WARK ERIA
 C IWO = WARK ERIA
 C NFXY1 = KOTEI SETTEN SIU

```

C      NFXV2 = KOTEI SETTEN SUU
C      NFXV3 = KOTEI SETTEN SUU
      DIMENSION ISPU(300),NUE(15),NJOI(15),NIUNJO(15,30)
      *NFI(15,25),KJOI(15),NFJOI(15),NWJO(300),NCLS(300,4),ITBL(15,4)
      DIMENSION LIUN(15),INOP(500),IWO(3),NEE(300,12),NEW(500)
      *GAMMA(500),EE(500),V(500),X(500),Y(500),NMAT(15)
      *JTBL(15,120),NTABL(4,2),NWAT(300,2),NAPAR(60),NNPAR(60)
      DIMENSION MFX(300),XD(300),YD(300),PE(15),PV(15),PG(15),
      1ALP(20,20),M1(500),M2(500),MJ(500),MK(500)
      DOUBLE PRECISION A10,A20,A30,A40,A50

      INTEGER PS,NW,PCYCL,OPT1
      READ(40,99) A10,A20,A30,A40,A50
      99 FORMAT(5A8)
      READ(40,100) NP,NE,NU,PS,T,NOP,ISW1,MCYCL,PCYCL,OPT1,LINE,EDP2
      WRITE(50,800) NP,NE,NU,PS,T,NOP
      800 FORMAT(1H0,3HNPS=,15,3X,3HNE=,15,3X,3HNU=,15,3X,3HPS=,15
      1,2X,2HT=,F10,0,3X,4HNOP=,15)
      100 FORMAT(415,F10,0,615,F10,1)
      IF(NP.LE.300) GO TO 900
      III=1

      GO TO 999
      900 IF(NE.LE.500) GO TO 901
      III=2
      GO TO 999
      901 IF(NU.LE.15) GO TO 902
      III=3
      GO TO 999
      902 READ(40,101)NFXV1,NFXV2,NFXV3
      REWIND 81
      WRITE(50,801)NFXV1,NFXV2,NFXV3

      801 FORMAT(1H0,6HNFXV1=,13,3X,6HNFXV2=,13,3X,6HNFXV3=,13)
      101 FORMAT(3I3)
      DO 50 I=1,NP
      50 ISPU(I)=1
      IF(NFXV1)10,9,10
      102 FORMAT(2014)
      10 READ(40,102) (MFX(I),I=1,NFXV1)
      WRITE(50,802) (MFX(I),I=1,NFXV1)
      802 FORMAT(2X,22H* KOTEI SETIFN BANGO *,2014)
      DO 51 I=1,NFXV1

      NN=MFX(I)
      ISPU(NN)=4
      51 CONTINUE
      9 IF(NFXV2)12,11,12
      12 READ(40,102) (MFX(I),I=1,NFXV2)
      DO 52 I=1,NFXV2
      NN=MFX(I)
      ISPU(NN)=2
      52 CONTINUE
      WRITE(50,3000)

```

```

3000 FORMAT(1H0,5H** 1 )
11 IF(NFXY3)13,14,13
13 READ(40,102) (MFX Y(I),I=1,NFXY3)
WRITE(50,803) (MFX Y(I),I=1,NFXY3)
803 FORMAT(2X,12H* Y-ROLLER *2014)
WRITE(50,4000)
4000 FORMAT(2X,20H* SETTEN ZAHYO CHI *)
DO 53 I=1,NFXY3
NN=MFX Y(I)
ISPU(NN)=3
- 100-

53 CONTINUE
14 READ(40,103) (MFX Y(I),XD(I),YD(I),I=1,NP)
WRITE(50,804) (MFX Y(I),XD(I),YD(I),I=1,NP)
804 FORMAT(4(14,2F8.3))
103 FORMAT(4(14,2F8.3))
DO 54 I=1,NP
NN=MFX Y(I)
X(NN)=XD(I)
Y(NN)=YD(I)
54 CONTINUE
- 110-

READ(40,104) (PE(I),PV(I),PG(I),I=1,NOP)
WRITE(50,805) (PE(I),PV(I),PG(I),I=1,NOP)
805 FORMAT(2X,17H* DANSEI PATARN *,3(3X,3F7.0),8X)
WRITE(50,3002)
3002 FORMAT(1H0,27H* SETTEN NO SESSYOKU UNIT *)
104 FORMAT(3(3X,3F7.0),8X)
905 NSUM=0
READ(40,106) (NUE(I),I=1,NU)
106 FORMAT(2014)
DO 58 I=1,NU
- 120-

58 NSUM=NSUM+NUE(I)
IF(NSUM.EQ.NE) GO TO 906
I11=7
GO TO 999
906 CONTINUE
DO 621 KK=1,NE
READ(40,107) K1,K2,K3,K4,K5
NEW(K2)=NEW(K2)+1
NEW(K3)=NEW(K3)+1
NEW(K4)=NEW(K4)+1
- 130-

NN1=NEW(K2)
NN2=NEW(K3)
NN3=NEW(K4)
NEE(K2,NN1)=K1
NEE(K3,NN2)=K1
NEE(K4,NN3)=K1
H1(KK)=K1
H1(KK)=K2
HJ(KK)=K3
HK(KK)=K4
- 140-

```

```

INOP(KK)=K5
IF(0.LE.K2.AND.K2.LT.NP) GO TO 621
III=4
GO TO 999
IF(0.LE.K3.AND.K3.LT.NP) GO TO 621
III=5
GO TO 999
IF(0.LE.K4.AND.K4.LT.NP) GO TO 621
III=6
GO TO 999
- 150-

621 CONTINUE
WRITE(50,3510) ((NCE(L1,L2),L2=1,12),L1=1,NP)
3510 FORMAT(1H,12I5)
WRITE(50,4001)
4001 FORMAT(1H0,11H* ELEMENT *)
WRITE(50,107) (MI(I),MJ(I),MK(I),INOP(I),I=1,NE)
WRITE(50,4002)
4002 FORMAT(1H0,24H* NUNJO AND NJOI TABLE *)
107 FORMAT(20I4)
DO 59 I=1,NE
- 160-

NNN=INOP(I)
GAMMA(I)=PG(NNN)
EE(I)=PE(NNN)
V(I)=PV(NNN)
59 CONTINUE
C TABLE-1 SAKUSEI
DO 60 I=1,NU
60 NJOI(I)=0
IEND=0
ISTA=1
- 170-

DO 61 IUNIT=1,NU
IEND=IFND+NUE(IUNIT)
DO 62 II=ISTA,IEND
IF(0.LT.NJOI(IUNIT)) GO TO 90
NUNJO(IUNIT,1)=MI(ISTA)
NUNJO(IUNIT,2)=MJ(ISTA)
NUNJO(IUNIT,3)=MK(ISTA)
NJOI(IUNIT)=3
GO TO 62
90 IWO(1)=MI(II)
- 180-

IWO(2)=MJ(II)
IWO(3)=MK(II)
DO 64 IB=1,3
IN=IWO(IB)
IKE=NJOI(IUNIT)
DO 65 IP=1,IKE
IF(IN.EQ.NUNJO(IUNIT,IP)) GO TO 64
65 CONTINUE
IKE=IKE+1
NUNJO(IUNIT,IKE)=IN
- 190-

```

```

      NJOI(IUNIT)=IKE
64  CONTINUE
62  CONTINUE
61  ISTA=ISTA+NUE(IUNIT)
C   TABLE-1 OUT PUT(TEST)
    DO 400 I=1,NU
400  WRITE(50,500) (NUNJO(I,J),J=1,30)
      WRITE(50,501) (NJOI(I),I=1,NU)
500  FORMAT(1X,10I4)
501  FORMAT(///,1X,4I4,/,1H1)
- 210-

      NZ=NZ+1
      NWAT(INTA*NE)=NZ
C   TABLE-2 SAKUSEI
      WRITE(50,4004)
4004  FORMAT(1H0,24H* NUNJO AND KJOI TABLE *)
      DO 66 I=1,NU
        NX=0
        IT=0
        JEND=NJOI(I)
        DO 67 J=1,JEND
          II=NUNJO(I,J)
          IK=ISPU(II)
          IF(IK.NE.4) GO TO 91
          NX=NX+1
          NFIX(I,NX)=II
          NUNJO(I,J)=0
          GO TO 67
67      IT=IT+1
          NUNJO(I,IT)=II
67      CONTINUE
- 220-

      KJOI(I)=IT
66  NFJOI(I)=NX
      DO 401 I=1,NU
401  WRITE(50,500) (NUNJO(I,J),J=1,30)
      WRITE(50,501) (KJOI(I),I=1,NU)
      WRITE(50,4040)
4040  FORMAT(1H0,24H* NFIX AND NFJOI TABLE *)
      DO 402 I=1,NU
402  WRITE(50,500) (NFIX(I,J),J=1,25)
      WRITE(50,501) (NFJOI(I),I=1,NU)
- 230-

C   TABLE-2 OUT PUT(TEST)
C   TABLE-3 SAKUSEI
      WRITE(50,4003)
4003  FORMAT(1H0,23H* NCLS AND NWJO TABLE *)
      DO 68 I=1,NP
68  NWJO(I)=0
      DO 69 I=1,NU
        DO 69 J=1,4
69  ITRL(I,J)=0
        DO 70 IK=1,NU
- 240-

```

```

- 250-
JP=KJOI(K)
DO 70 JK=1,JP
II=NUNJO(K,JK)
IF(NWJO(II).NE.0) GO TO 92
NWJO(II)=1
NCLS(II,1)=IK
GO TO 94
92 NWJO(II)=NWJO(II)+1
IKK=NWJO(II)
NCLS(II,IKK)=IK

- 260-
94 CONTINUE
70 CONTINUE
DO 403 J=1,NP
403 WRITE(50,502) (NCLS(I,J),J=1,4),NWJO(I)
502 FORMAT(1X,4I3,5X,I2)
C TABLE=3 OUTPUT(TEST)
C TABLE=4 SAKUSEI
DO 71 II=1,NP
L=NWJO(II)
IF(L.NE.1) GO TO 71

- 270-
KK=NCLS(II,1)
ITBL(KK,1)=ITBL(KK,1)+1
KE=ITBL(KK,1)
JTBL(KK,KE)=II
71 CONTINUE
DO 41 KI=1,NU
JP=KJOI(KI)
DO 72 JI=1,JP
II=NUNJO(KI,JI)
JK=NWJO(II)

- 280-
IF(JK.LE.1) GO TO 72
KAT=0
KMA=0
DO 73 LP=1,JK
KP=NCLS(II,LP)
IF(KI.GT.KP) GO TO 76
IF(KI.EQ.KP) GO TO 73
KAT=1
GO TO 73
76 KMA=1

- 290-
73 CONTINUE
IF(KAT.NE.1.OR.KMA.NE.1) GO TO 150
IC=3
GO TO 151
150 IF(KAT.NE.1) GO TO 152
IC=4
152 IF(KMA.NE.1) GO TO 151
IC=2
151 ITBL(KI,IC)=ITBL(KI,IC)+1
KAT=(IC-1)*30+ITBL(KI,IC)

```

```

      JTBL(KI,KAT)=11
72  CONTINUE
      WRITE(50,4005)
4005 FORMAT(1H0,17H* JTBL AND ITBL *)
      WRITE(50,504)(JTBL(KI,J),J=1,120)
      WRITE(50,506)(ITBL(KI,J),J=1,4)
      IF(KI.EQ.1) GO TO 41
      KKP=KI-1
      DO 155 LA=3,4
      KEN=ITBL(KKP,LA)

- 300-

      IF(KEN.EQ.0) GO TO 155
      LPP=30*(LA-1)
      DO 156 LB=1,KEN
      KZ=LPP+LB
      IN=JTBL(KKP,KZ)
      DO 157 LC=2,3
      LCN=ITBL(KI,LC)
      IF(LCN.EQ.0) GO TO 157
      NPA=30*(LC-1)
      DO 159 LD=1,LCN

- 310-

      K3=NPA+LD
      IF(IN.EQ. JTBL(KI,K3)) GO TO 156
159  CONTINUE
157  CONTINUE
      NW=NWJO(IN)
      NAE=0
      NAT=0
      DO 160 K5=1,NW
      ME=NCLS(IN,K5)
      IF(ME.LT.KI) NAE=1

- 320-

      IF(ME.GT.KI) NAT=1
160  CONTINUE
      IF(NAE.EQ.1) GO TO 910
      III=11
      GO TO 999
910  KF=2
      IF(NAT.EQ.1) KF=3
      ITBL(KI,KF)=ITBL(KI,KF)+1
      NF=ITBL(KI,KF)
      N44=30*(KF-1)+NF

- 330-

      JTBL(KI,N44)=IN
156  CONTINUE
155  CONTINUE
41  CONTINUE
      DO 161 NA=1,NU
      NTO=0
      DO 162 NB=1,4
162  NTO=NTO+ITBL(NA,NB)
      IF(NTO.LE.30) GO TO 161
      III=12

- 340-

```

```

- 350-
161 GO TO 999
C CONTINUE
TABLE=4 OUT PUT(TEST)
WRITE(50,503)
503 FORMAT(1H1)
DO 404 I=1,NU
WRITE(50,504)(JTBL(I,J),J=1,120)
404 WRITE(50,505)
504 FORMAT(1X,30I4)
505 FORMAT(/)

- 360-
DO 405 I=1,NU
405 WRITE(50,506)(ITBL(I,J),J=1,4)
506 FORMAT(1X,4I4)
WRITE(81) A10,A20,A30,A40,A50
WRITE(81) NP,NE,NU,PS,T,NOP,ISW1,MCYCL,PCYCL,OPT1,LINE,EPF2
WRITE(81) (ISPU(I),I=1,NP)
WRITE(81) (X(I),I=1,NP)
WRITE(81) (Y(I),I=1,NP)
WRITE(81) (NUE(I),I=1,NU)
WRITE(81) NEE

- 370-
DO 601 I=1,NE
L1=M1(I)
L2=M1(I)
L3=MJ(I)
L4=MK(I)
ICON=INOP(I)
G=GAMMA(I)
E=EE(I)
VV=V(I)
WRITE(81) L1,L2,L3,L4,E,VV,G,ICON

- 380-
601 CONTINUE
WRITE(81) ((JTBL(I,J),I=1,NU),J=1,120)
WRITE(81) ((ITBL(I,J),I=1,NU),J=1,4)
C KYOKAIJYOKEN SHORI NO TABLE SAKUSEI
DO 163 KT=1,4
DO 163 LAA=1,2
163 NTARL(KT,LAA)=0
NTARL(?,2)=-1
NTARL(3,1)=-1
DO 164 II=1,NU

- 390-
DO 165 JK=1,NP
DO 165 LK=1,2
165 NWAT(JK,LK)=0
NZ=0
NZZ=0
DO 166 JJ=1,4
NK=ITBL(II,JJ)
IF(NK.EQ.0) GO TO 166
MM=30-(JJ-1)
DO 167 KK=1,NK

```

```

LA=MM*KK
INTA=JTBL(11,LA)
NLP=ISPU(INTA)
DO 168 NE=1,2
KE=NTABL(NLP,NE)
IF(KE.EQ.-1) GO TO 169
NZ=NZ+1
NWAT(INTA,NE)=NZ
NAPAR(NZ)=INTA*10+NE
GO TO 168
- 400-

169 NZZ=NZZ+1
NNPAR(NZZ)=INTA*10+NE
168 CONTINUE
167 CONTINUE
166 CONTINUE
IF(NFJOI(11),EQ.0) GO TO 170
NJ=NFJOI(11)
DO 171 I=1,NJ
DO 171 J=1,2
NZZ=NZZ+1
- 410-

171 NNPAR(NZZ)=NFIK(11,I)*10+J
170 NNN=NZ+1
IF(NZZ.EQ.0) GO TO 172
DO 173 I=1,NZZ
NNN=NZ+1
NA=NNPAR(I)/10
NB=NNPAR(I)-NA*10
NWAT(NA,NB)=NNN
173 NAPAR(NNN)=NNPAR(I)
172 CONTINUE
- 420-

WRITE(81)((NWAT(I,J),I=1,NNP),J=1,2)
WRITE(81)(NAPAR(I),I=1,60)
C KYOKAIJYOKEN TABLE OUTPUT(TEST)
WRITE(50,503)
WRITE(50,4006)
4006 FORMAT(1H0,32H* NTABL , NWAT AND NAPAR TABLE *)
WRITE(50,507)((NTABL(I,J),J=1,2),I=1,4)
507 FORMAT(1X,2I3,/)
WRITE(50,508)((NWAT(I,J),J=1,2),I=1,NNP)
508 FORMAT(1X,2I4)
- 430-

WRITE(50,509)(NAPAR(I),I=1,NNN)
509 FORMAT(1X,25I4)
NHAT(11)=NZ+100+NZZ
164 CONTINUE
WRITE(81)(NHAT(I),I=1,NU)
NNH=0
READ(40,700)OW,(LUN(I),I=1,NU)
700 FORMAT(13,15I4)
DO 200 I=1,NU
NN1=LUN(I)/100
- 440-

```

```

NN2=LUN(1)-NN1*100
NNN=NNN+NN1*NN2
200 CONTINUE
IF(NNN.EQ.0) GO TO 300
READ(40,710) (MFXY(I),XD(I),YD(I),I=1,NNN)
710 FORMAT(4(14,2F8.3))
WRITE(50,3004)
3004 FORMAT(1H0,5H** 4 )
300 CONTINUE
IF(LINE.EQ.1) GO TO 5000

- 450-

READ(40,720) N1,EDP
DO 210 I=1,NOP
KK=2*I-1
720 FORMAT(15,E10.0)
N10=2*N1
READ(40,730) (X(I),I=1,N10)
DO 211 I=1,N1
I0=2*I-1
ALP(I,KK)= X(I0)
ALP(I,KK+1)= X(I0+1)

- 460-

211 CONTINUE
730 FORMAT(5(2F8.0))
210 CONTINUE
5000 CONTINUE
WRITE(81) OV,NNN,(LUN(I),I=1,NU)
IF(NNN.EQ.0) GO TO 310
WRITE(81) (MFXY(I),I=1,NNN)
WRITE(81) (XD(I),I=1,NNN)
WRITE(81) (YD(I),I=1,NNN)
310 CONTINUE

- 470-

IF(LINE.NE.1) WRITE(81) N1,EDP,ALP
READ(40,556) ICYCL
556 FORMAT(15)
WRITE(81) ICYCL
REWIND 81
WRITE(50,503)
WRITE(50,4007)
4007 FORMAT(1H0,14H* NMAT TABLE *)
WRITE(50,510) (NMAT(I),I=1,NU)
510 FORMAT(1X,16)

- 480-

WRITE(50,4010)
4010 FORMAT(1H1//////20X,13(2H**)/20X,13(2H**)//24X,16HPROGRAM (
*A) END ,////20X,13(2H**)/20X,13(2H**))
WRITE(50,4009)
4009 FORMAT(3H1)
GO TO 1000
C ERROR MESSAGE OUT
119 FORMAT(10X,5HERROR(12)
999 WRITE(50,119) III
1000 STOP

- 490-

- 500-

```

END

```

C      DSEK01
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
C      ** COMMON /A1/ **
C      NUC = KAKU UNIT NO ELEMENT SU
C      NWAT = KYOKAI JYOKEN SYORI TABLE
C      LUNI = KAKU UNIT NO KAZU SU
C      NMAT = KAKU UNIT NO JIYUDO SU
C      ICAB = KAKU UNIT NO JIYUSETTEN SU
C      ICBA = KAKU UNIT NO JIYUSETTEN SU + 1
C      NAPAR = KYOKAI SYORI GO NO INDEX
C
C      ITBL = CLASS BUNRUI TABLE
C      JTBL = CLASS BUNRUI TABLE
C      NEJO = ELEMENT NO I·J·K SETTEN BANGO
C      ** COMMON /A11/ **
C      X = X-ZAHYO
C      Y = Y-ZAHYO
C      WK = ELEMENT·STIFFNESS·MATRIX
C      USM = KAKU UNIT TANI NO STIFFNESS·MATRIX
C      TSM = USM NO WORK AREA
C      FOCE = KAKU UNIT TANI NO KAJUKOH
C
C      WFOCE = FOCE NO WORK AREA
C      ADEF = HENI AND HANRYOKU
C      ** COMMON /A21/ **
C      I10,I20,I30,I40,I50 = KOHZOHBUU MEISYO
C      NPAGE = PAGE-SU
C      ** COMMON /A3/ **
C      EDP = SYUSOKUCHI
C      WW1 = X-HOKOH NO ZEN KAJU
C      WW2 = Y-HOKOH NO ZEN KAJU
C      NP = SETTEN SU
C
C      NE = ELEMENT SU
C      NU = UNIT SU
C      PS = PLANE STRESS AND STRAIN NO KUBETU
C           PS=0    PLANE STRESS
C           PS=1    PLANE STRAIN
C      T = ITAATU
C      OW = JITYU NO UMU
C      NOP = DANSEI PATTERN SU
C      NNI = NON LINEAR DATA SU
C      ** COMMON /A4/ **
C
C      KSW = STOP HANTEI
C      ALP = NON LINEAR DATA
C      LIKO = SYUSOKU CYCLE SU
C      ** COMMON /A5/ **
C      IISI = SUB·FOU NI OITE ERROR NO BAAI NO STOP HANTEI
C      ** DRUM DIMENSION **
C      DAPAR = NAPAR NO NAIYO O KAKUNO
C      DNWAT = NWAT NO NAIYO O KAKUNO
C      DFLD = ELEMENT DATA NO KAKUNO
C      USM2 = USM·FOCE NO NAIYO O KAKUNO

```

- 10-

- 20-

- 30-

- 40-

```

C      DXY = SETTEN ZAHYO KAKUNOH
C      DRDEF = ADEF NO NAIYOH KAKUNOH
C      DDD = N-1 CYCLE NO NAIYOH KAKUNOH
C      DNEP = NEP NO NAIYOH O KAKUNOH
C      ALOAD = SETTEN KAZU O KAKUNOH
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTRL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
- 50-

COMMON /A3/ EDP,WH1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
COMMON /A5/ IISWI
COMMON /A6/ ICYCL,HCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* DLOAD(900)
INTEGER OW,PS,PCYCL,OPT1
DOUBLE PRECISION I10,I20,I30,I40,I50
- 60-

ICYCL=1
KSW=0
IISWI=0
NPAGE=0
WRITE (50,1000)
1000 FORMAT(1H1////////,30X,12(3H***),1H*/30X,1H*,35X,1H*/30X,37H* S
1TRESS AND DEFORMATION ANALYSIS */30X,1H*,35X,1H*/30X,37H* BY
2 FINITE ELEMENT METHOD */30X,1H*,35X,1H*/30X,12(3H***),1H*)
CALL DREAD
IPHAN=PCYCL+ICYCL-1
- 70-

10 CALL STIF
CALL FORW
IF(IISWI.EQ.1) STOP
CALL BACK
CALL REAC
CALL STRESS
ICYCL=ICYCL+1
IF(LINF.EQ.1) GO TO 20
IF(KSW.EQ.0) GO TO 10
20 WRITE(50,4010)
- 80-

4010 FORMAT(1H1////////20X,13(2H**) /20X,13(2H**) ////24X,18HPROGRAM (
*B.) END ,////20X,13(2H**)/20X,13(2H**) )
STOP
END
- 90-

```

```

SUBROUTINE DREAD
BSEK02
C
C TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /A2/ X(300),Y(300),WK(6,6),USH(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NNI
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
- 10-

COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DOUBLE PRECISION I10,I20,I30,I40,I50
INTEGER OW,PS,PCYCL,OPT1
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* DXY(600),DREDF(600),DDD(600),DNEP(3600),ALOAD(600)
* DLOAD(900)
DIMENSION NEP(300,12)
EQUIVALENCE (USH,NEP)
REWIND 81
- 20-

REWIND 82
READ (81) I10,I20,I30,I40,I50
READ(81) NP,NE,NU,PS,T,NOP,ISWI,MCYCL,PCYCL,OPT1,LINE,EDP2
READ (81) (ISPU(I),I=1,NP)
READ (81) (X(I),I=1,NP)
READ (81) (Y(I),I=1,NP)
READ (81) (NUC(I),I=1,NU)
READ (81) NEP
DRUM WRITE DXY,X,Y
DRUM WRITE DNEP,NEP
- 30-

IF (ISWI.EQ.1) WRITE(50,1060)
1060 FORMAT(1H1)
IF (MCYCL.EQ.0) MCYCL=10
DO 10 I=1,NE
JT=8-I-7
READ (81) N1,N2,N3,N4,EE,V,GAMMA,ICON
DRUM WRITE DELD(JT),N1,N2,N3,N4,EE,V,GAMMA,ICON
IF (ISWI.NE.1) GO TO 10
WRITE(50,1031) N1,N2,N3,N4,EE,V,GAMMA,ICON
1031 FORMAT(1H0,4I5,3F10.3,15)
- 40-

10 CONTINUE
READ (81) ((JTBL(J,I),I=1,NU),J=1,120)
READ (81) ((ITBL(I,J),I=1,NU),J=1,4)
IF (ISWI.NE.1) GO TO 90
WRITE(50,1012) (NUC(I),I=1,NU)
WRITE(50,1040) ((JTBL(J,I),I=1,NU),J=1,120)
WRITE(50,1040) ((ITBL(I,J),I=1,NU),J=1,4)
1040 FORMAT(1H ,4I5)
90 CONTINUE
DO 20 I=1,NU

```

```

110=600*11-599
111=60*11-59
READ (81) ((NWAT(I,J),I=1,NP),J=1,2)
READ (81) (NAPAR(I),I=1,60)
DRUM WRITE DNWAT(110),NWAT
DRUM WRITE DAPAR(111),NAPAR
IF (ISW1.NE.1) GO TO 20
WRITE(50,1050) ((NWAT(I,J),I=1,NP),J=1,2)
WRITE(50,1012) (NAPAR(I),I=1,60)
20 CONTINUE

READ (81) (NHAT(I),I=1,NU)
IF (ISW1.NE.1) GO TO 130
WRITE(50,1012) (NHAT(I),I=1,NU)
130 CONTINUE
1050 FORMAT(1H0,2014)
READ (81) OW,NNN,(LUNI(I),I=1,NU)
IF (NNN.EQ.0) GO TO 40
READ (81) (NAPAR(I),I=1,NNN)
READ (81) (X(I),I=1,NNN)
READ (81) (Y(I),I=1,NNN)

DO 60 III=1,NNN
JT=3*III-2
DRUM WRITE DLOAD(JT),NAPAR(III),X(III),Y(III)
60 CONTINUE
40 CONTINUE
IF (LINE.NE.1) READ(81) NN1,EDP,ALP
READ(81) ICYCL
ICYCL=ICYCL+1
IF (ICYCL.EQ.1) GO TO 150
READ(81) (ADEF(I),I=1,600)

DRUM WRITE DDD ,ADEF
150 CONTINUE
NOP2=2*NOP
IF (LINE.EQ.1) GO TO 100
WRITE(50,3000)
GO TO 110
100 WRITE(50,3010)
110 CONTINUE
3000 FORMAT(1H0/38X,23H( NON LINEAR ANALYSIS )//56X,7H(KG,CM))
3010 FORMAT(1H0/38X,23H( LINEAR ANALYSIS )//56X,7H(KG,CM))

WRITE (50,1000)
1000 FORMAT(1H0/5X,16H** INPUT DATA **)
IF (PS.EQ.1) GO TO 30
WRITE (50,1010)
1010 FORMAT(1H0,8X,15H** PLANE STRAIN)
GO TO 35
30 WRITE (50,1011)
1011 FORMAT(1H0,8X,15H** PLANE STRESS)
35 CONTINUE
WRITE (50,1020) T

```

```

1020 FORMAT(1H0,8X,14H** THICKNESS= ,F6.2)
      IF (OW.EQ.1) WRITE(50,1030)
1030 FORMAT(1H0,8X,13H** OWN WEIGHT)
      WRITE(50,1001) NP,NE,NU,NOP
1001 FORMAT(1H0//11X,19HNUMBER OF JOINT =,I5//11X,19HNUMBER OF ELEMEN
      *T =,I5//11X,19HNUMBER OF UNIT =,I5//11X,19HNUMBER OF PATTERN =,
      *I5)
1012 FORMAT(1H0,20I5)
      CALL PAGE
      IF(LINE.EQ.1) GO TO 120
- 100-

      IF(NOP.GT.5) NOP2=10
      DO 50 I=1,NN1
      WRITE(50,2020) (ALP(I,J),ALP(I,J+1),J=1,NOP2,2)
2020 FORMAT(1H ,5(2F10.5,3X))
50 CONTINUE
      IF(NOP.LE.5) GO TO 70
      NOP2=2*NOP
      WRITE(50,2030)
2030 FORMAT(1H0)
      DO 80 I=1,NN1
- 110-

      WRITE(50,2020) (ALP(I,J),ALP(I,J+1),J=11,NOP2,2)
80 CONTINUE
70 CONTINUE
120 CONTINUE
      IF(LINE.EQ.1) GO TO 121
      WRITE(50,2040) EDP2
2040 FORMAT(1H0,15HDAMPING FACTOR=,F15.7)
      WRITE(50,2041) EDP
2041 FORMAT(1H0,17HERROR LIMITATION=,E15.8)
121 CONTINUE
- 120-

      RETURN
      END
- 130-

```

```

C      SUBROUTINE STIF
C      BSEK03
      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
      COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
      * ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
      COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
      * WFOCE(60),ADEF(600),ISPU(300)
      COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
      COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,QW,NOP,NN1
      COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE

      COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
      INTEGER OW,PS,PCYCL,OPT1
      DOUBLE PRECISION I10,I20,I30,I40,I50
      DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
      * ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
      * ,DLOAD(900)
      DIMENSION DEF(300,2)
      EQUIVALENCE (ADEF,DEF)
      DRUM READ DXY,X,Y
      REWIND 82

      REWIND 81
      WW1=0.
      WW2=0.
      ICOU=0
      KCOU=0
      NEL=0
      DO 180 I1=1,2
      DO 180 I2=1,NP
      DEF(I2,I1)=0.
180 CONTINUE

1040 FORMAT(1H0,2X,2HNE,5X,2HIE,2X,2HJE,2X,2HKE,3X,5HGAMMA,3X,8HYOUNG-M
      * ,3X,10HPOISSONS-R,4X,8HYOUNG-P ,/)
      ICOU1=46
      DO 10 K0=1,NU
      IF(ICYCL.EQ.1.OR.OPT1.EQ.1) GO TO 301
      GO TO 300
301 CONTINUE
      CALL PAGE
      WRITE(50,1050) K0
      WRITE(50,1040)

1050 FORMAT(1H0,5H*UNIT,13,2H *)
300 CONTINUE
      NEND=NUE(K0)
      NW=NMAT(K0)/100
      JEN=NMAT(K0)-NW*100
      NEN=NW+JEN
      DO 20 J1=1,NEN
      FOCE(J1)=0.
      WFOCE(J1)=0.
20 CONTINUE

```

```

Ii=x0*600-599
DRUM READ DNWAT(Ii),NWAT
DO 30 F1=1,NEN
DO 30 K2=1,NEN
USM(K1,K2)=0.
30 CONTINUE
DO 40 K3=1,NEND
NEL=NEL+1
JT=R*NEL-7
DRUM RFAD DELO(JT),NOEL,NEJO,EE,V,GAMMA,ICON
IF(ICYCL,E0,1-OR,OPT1,E0,1) GO TO 311
GO TO 310
311 CONTINUE
WRITE(50,1060)NOEL,NEJO,GAMMA,EE,V,ICON
1060 FORMAT(1H ,I4,3X,3I4,F8.3,F10.1,F11.3,8X,12)
310 CONTINUE
Ii=NEJO(1)
JJ=NEJO(2)
KK=NEJO(3)
XI=X(Ii)
XJ=X(JJ)
XK=X(KK)
YI=Y(Ii)
YJ=Y(JJ)
YK=Y(KK)
IF(PS,E0,1) GO TO 60
C1=EE*(1.0-V)/((1.0+V)*(1.0-2.0*V))
C2=V/(1.0-V)
GO TO 50
60 CONTINUE
C1=EE/(1.0-V *V)
C2=V
50 CONTINUE
F1=(1.0-C2)*0.5
F2=(1.0+C2)*0.5
C0=C1*T/(2.0*(C1-XI)*YK+(X1-XK)*YJ+(XK-XJ)*YI)
XIJ=XI-XJ
XJM=XJ-XK
XMI=XK-XI
YIJ=YI-YJ
YJM=YJ-YK
YMI=YK-YI
WK(1,1)=(F1*XJM*XJH+YJM*YJH)*C0
WK(1,2)=(-F2*XJM*YJH)*C0
WK(1,3)=(F1*XJM*XMI+YJM*YMI)*C0
WK(1,4)=(-F1*XJM*YMI+C2*XMI*YJH)*C0
WK(1,5)=(F1*XIJ*XJH+YIJ*YJH)*C0
WK(1,6)=(-F1*XJM*YIJ+C2*XIJ*YJH)*C0
WK(2,2)=(XJM*XJH+F1*YJM*YJH)*C0
WK(2,3)=(-F1*XMI*YJH+C2*XJM*YMI)*C0

```

- 50-

- 60-

- 70-

- 80-

- 90-

```

WK(2,4)=(XJM*XMI+F1*YJM*YMI)*C0
WK(2,5)=(-F1*XIJ*YJM+C2*XJM*YIJ)*C0
WK(2,6)=(XIJ*XJM+F1*YIJ*YJM)*C0
WK(3,3)=(F1*XMI*XMI+YMI*YMI)*C0
WK(3,4)=(-F2*XMI*YMI)*C0
WK(3,5)=(F1*XIJ*XMI+YIJ*YMI)*C0
WK(3,6)=(-F1*XMI*YIJ+C2*XIJ*YMI)*C0
WK(4,4)=(XMI*XMI+F1*YMI*YMI)*C0
WK(4,5)=(-F1*XIJ*YMI+C2*XMI*YIJ)*C0
WK(4,6)=(XIJ*XMI+F1*YIJ*YMI)*C0
- 100-

WK(5,5)=(F1*XIJ*XIJ+YIJ*YIJ)*C0
WK(5,6)=(-F2*XIJ*YIJ)*C0
WK(6,6)=(XIJ*XIJ+F1*YIJ*YIJ)*C0
DO 70 K4=1,5
KK4=K4+1
DO 70 K5=KK4,6
WK(K5,K4)=WK(K4,K5)
- 110-

70 CONTINUE
C WRITE(50,4010) ((WK(I,JJ),I=1,6),JJ=1,6)
4010 FORMAT(1H,6F12.5)

DO 80 K6=1,3
N11=NEJO(K6)
KR=2*K6-2
DO 90 K7=1,3
N33=NEJO(K7)
JR=2*K7-2
DO 100 K8=1,2
KK=NWAT(N11,K8)
KE=KR+K8
DO 110 K9=1,2

KL=NWAT(N33,K9)
NO=JR+K9
USM(KK,KL)=USM(KK,KL)+WK(KE,NO)
- 120-

110 CONTINUE
100 CONTINUE
90 CONTINUE
80 CONTINUE
IF(OW.EQ.0) GO TO 150
WE=-0.001*GAMMA*T*((XJ-XI)*(YK-YI)-(XK-XI)*(YJ-YI))/6.
DO 120 K10=1,3
- 130-

N33=NEJO(K10)
IR=NWAT(N33,2)
FOCE(IR)=FOCE(IR)+WE
DEF(N33,2)=DEF(N33,2)+WE
- 140-

120 CONTINUE
150 CONTINUE
40 CONTINUE
KLU2=LUNI(K0)/100
KLU1=LUNI(K0)-KLU2*100
IF(KLU1.EQ.0) GO TO 130

```

- 150-

```

DO 140 K11=1,KLU1
ICOU=ICOU+1
JT=3*ICOU-2
DRUM READ DLOAD(JT),LNO,XL,YL
NP1=NWAT(LNO,1)
FOCE(NP1)=FOCE(NP1)+XL
NP1=NWAT(LNO,2)
FOCE(NP1)=FOCE(NP1)+YL
DEF(LNO,1)=DEF(LNO,1)+XL
DEF(LNO,2)=DEF(LNO,2)+YL

140 CONTINUE
130 CONTINUE
IF(KLU2.EQ.0) GO TO 190
DO 200 K11=1,KLU2
ICOU=ICOU+1
JT=3*ICOU-2
DRUM READ DLOAD(JT),LNO,XL,YL
NP1=NWAT(LNO,1)
WFOCE(NP1)=WFOCE(NP1)+XL
NP1=NWAT(LNO,2)
WFOCE(NP1)=WFOCE(NP1)+YL

200 CONTINUE
CALL INDI(NEN,NW)
190 CONTINUE
IF(ICYCL.EQ.1.OR.OPT1.EQ.1) GO TO 321
GO TO 320
321 CONTINUE
WRITE(50,1090) K0
1090 FORMAT(1H0//5X,10H** LOAD **,5X,6HUNIT =,13//5X,2HNP,10X,5HX(KG),
*10X,5HY(KG)//)

320 CONTINUE
DO 170 K11=1,NP
IF(NWAT(K11,1).EQ.0) GO TO 170
KRR=NWAT(K11,1)
KRL=NWAT(K11,2)
W1=FOCE(KRR)
W2=FOCE(KRL)
WW1=WW1+W1
WW2=WW2+W2
IF(ICYCL.EQ.1.OR.OPT1.EQ.1) GO TO 331
GO TO 330
331 CONTINUE
WRITE(50,1070) K11,W1,W2
1070 FORMAT(1H ,3X,13,2F15.5)
330 CONTINUE
170 CONTINUE
WRITE(81) ((USH(J1,J2),J1=1,60),J2=1,60)
WRITE(81) (FOCE(J1),J1=1,60)
10 CONTINUE
IF(ICYCL.EQ.1.OR.OPT1.EQ.1) WRITE(50,1080) WW1,WW2

```

- 160-

- 170-

- 180-

- 190-

1080 FORMAT(1H0/7X.6F15.5)
DRUM WRITE ALOAD,ADEF
RETURN
END

- 200-

```

SUBROUTINE MTWT
BSEK04
C
C TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NHAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USH(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISW1,LINE
- 10-
COMMON /A5/ IISW1
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* ,DLOAD(900)
INTEGER OW,PS,PCYCL,OPT1
DIMENSION NEP(300,12)
EQUIVALENCE (USH,NEP)
DOUBLE PRECISION I10,I20,I30,I40,I50
- 20-
IPHAN=IPHAN+PCYCL
REWIND 81
WRITE(81) I10,I20,I30,I40,I50
WRITE(81) NP,NE,NU,PS,T,NOP,ISW1,MCYCL,PCYCL,OPT1,LINE,EDP2
WRITE(81) (ISPU(I),I=1,NP)
DRUM READ DXY,X,Y
DRUM READ DNEP,NEP
WRITE(81) (X(I),I=1,NP)
WRITE(81) (Y(I),I=1,NP)
WRITE(81) (NUC(I),I=1,NU)
- 30-
WRITE(81) NEP
DO 10 I1=1,NE
JT=8*I1-7
DRUM READ DELD(JT),N1,N2,N3,N4,EE,V,GAMMA,ICON
WRITE(81) N1,N2,N3,N4,EE,V,GAMMA,ICON
10 CONTINUE
WRITE(81) ((JTBL(J,I),I=1,NU),J=1,120)
WRITE(81) ((ITBL(I,J),I=1,NU),J=1,4)
DO 20 I1=1,NU
I10=600*I1-599
- 40-
I11=60*I1-59
DRUM READ DNWAT(I10),NWAT
DRUM READ DAPAR(I11),NAPAR
WRITE(81) ((NWAT(I,J),I=1,NP),J=1,2)
WRITE(81) (NAPAR(I),I=1,60)
20 CONTINUE
WRITE(81) (NHAT(I),I=1,NU)
WRITE(81) OW,NNN,((LUNI(I),I=1,NU)
IF(NNN.EQ.0) GO TO 40
DO 30 I1=1,NNN

```

- 50-

```
JT=3*II-2
DRUM READ DLOAD(JT),NAPAR(II),X(II),Y(II)
30 CONTINUE
WRITE(81) (NAPAR(I),I=1,NNN)
WRITE(81) (X(I),I=1,NNN)
WRITE(81) (Y(I),I=1,NNN)
40 CONTINUE
IF(LINE.NE.1) WRITE(81) NN1,EDP,ALP
WRITE(81) ICYCL
DRUM READ DDD,ADEF
```

- 60-

```
WRITE(81) (ADEF(I),I=1,600)
REWIND 81
PAUSE 55555
RETURN
END
```

```

SUBROUTINE FORM
BSEK05
C
C
TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
10-
COMMON /A5/ IISWI
DIMENSION NWOK(60)
DOUBLE PRECISION I10,I20,I30,I40,I50
INTEGER OW,PS,PCYCL,OPT1
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000),
* DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
DIMENSION DEF(300,2)
EQUIVALENCE (ADEF,DEF)
IF(ISWI,NE,1) GO TO 130
DO 8010 I11=1,NU
20-
K0=600*I11-599
K1=60*I11-59
DRUM READ DNWAT(K0),NWAT
DRUM READ DAPAR(K1),NAPAR
WRITE(50,6010) ((NWAT(LL1,LL2),LL2=1,2),LL1=1,NP)
WRITE(50,6020) (NAPAR(LL1),LL1=1,60)
8010 CONTINUE
130 CONTINUE
REWIND 81
DO 120 I1=1,NU
30-
ICAB(I1)=0
ICBA(I1)=0
120 CONTINUE
DO 9999 I1=1,NU
K0=600*I1-599
7010 FORMAT(1H0,3H50=,I3)
DRUM READ DNWAT(K0),NWAT
IF(ISWI,NE,1) GO TO 140
WRITE(50,6010) ((NWAT(LL1,LL2),LL2=1,2),LL1=1,NP)
6010 FORMAT(1H ,215)
40-
140 CONTINUE
READ(81) ((USM(J1,J2),J1=1,60),J2=1,60)
READ(81) (FOCE(J1),J1=1,60)
J2=NMAT(I1)/100
IF(I1,EO,1) GO TO 299
K11=60*I1-119
6020 FORMAT(1H ,2015)
DRUM READ DAPAR(K11),NAPAR
IF(ISWI,NE,1) GO TO 150
WRITE(50,6020) (NAPAR(LL1),LL1=1,60)

```

150 CONTINUE - 50-
 KLL=ICBA(II-1)
 KAI=NMAT(II-1)/100
 IF(KAI.EQ.ICAB(II-1)) GO TO 6134
 DO 2503 NA=KLL,KAI
 K93=NAPAR(NA)
 KJ=K93/10
 KC=K93-KJ*10
 IND=NWAT(KJ,KC)
 NWOK(NA)=IND

2503 CONTINUE - 60-
 DO 31 K1=KLL,KAI
 K71=NWOK(K1)
 DO 33 K4=KLL,KAI
 K6=NWOK(K4)
 USM(K71,K6)=USM(K71,K6)+TSH(K1,K4)

33 CONTINUE
 FOCE(K71)=FOCE(K71)+WFOCE(K1)

31 CONTINUE
 6134 CONTINUE - 70-

299 CONTINUE
 IF(ITBL(II,2).EQ.0) GO TO 63
 NK=30+ITBL(II,2)
 NA=JTBL(NK,II)
 DO 61 K1=1,2
 KP=3-K1
 NI=NWAT(NA,KP)
 IF(NI.GT.0.AND.NI.LE.J2) GO TO 62

61 CONTINUE
 WRITE(50,2310) II,NI,J2 - 80-

ITSWI=1
 GO TO 8000

63 CONTINUE
 NK=ITBL(II,1)
 NA=JTBL(NK,II)
 DO 71 K1=1,2
 KP=3-K1
 NI=NWAT(NA,KP)
 IF(NI.GT.0.AND.NI.LE.J2) GO TO 62

71 CONTINUE - 90-
 WRITE(50,2310) II,NI,J2
 ITSWI=1
 GO TO 8000

62 CONTINUE
 ICAR(II)=NI
 IF(II.EQ.NU) GO TO 2616
 IF(ITBL(II,3).EQ.0) GO TO 73
 NA=JTBL(61,II)
 DO 81 K1=1,2
 NI=NWAT(NA,K1)

```

      IF(NI.GT.0.AND.NI.LE.JZ) GO TO 83
81  CONTINUE
      WRITE(50,2310) II,NI,JZ
      IISW=1
      GO TO 8000
73  CONTINUE
      NA=JTBL(91,II)
      DO 84 K1=1,2
      NI=NWAT(NA,K1)
      IF(NI.GT.0.AND.NI.LE.JZ) GO TO 83

84  CONTINUE
      WRITE(50,2310) II,NI,JZ
      IISW=1
      GO TO 8000
83  CONTINUE
      ICBA(II)=NI
2616 CONTINUE
      NEN=ICAB(II)
      IF(ICBA(II).EQ.1) GO TO 6771
C *** INVERSE MATRIX KA ***
      CALL AMINV(NEN)
      JIE=JZ
      JIA=ICBA(II)
      KIE=ICAB(II)
C *** KA-1*KAB CALCULATION ***
      IF(II.EQ.NU) GO TO 2201
      DO 96 KI=1,KIE
      DO 93 JI=JIA,JIE
      TT=0.
      DO 94 NI=1,KIE
      TT=TT+USM(KI,NI)*USM(NI,JI)
94  CONTINUE
      TSM(KI,JI)=TT
93  CONTINUE
96  CONTINUE
2201 CONTINUE
      DO 95 NA=1,KIE
      WFOCE(NA)=0.
      TT=0.
      DO 97 IA=1,KIE
      TT=TT+FOCE(IA)*USM(NA,IA)
97  CONTINUE
      WFOCE(NA)=TT
95  CONTINUE
7020 FORMAT(1H0,3HK3=,15)
      WRITE(82) (WFOCE(I),I=1,60)
      WRITE(82) ((TSM(I1,I2),I1=1,60),I2=1,60)
      IF(II.EQ.NU) GO TO 9999
      GO TO 6772
6771 CONTINUE

```

- 100-

- 110-

- 120-

- 130-

- 140-

```

B1=0.
WRITE(A2) B1
6772 CONTINUE
DO 101 JJ=JIA,JIE
DO 101 KK=JIA,JIE
TSM(JJ,KK)=USM(JJ,KK)
101 CONTINUE
DO 105 JJ=JIA,JIE
DO 103 KK=JIA,JIE
TT=0.
DO 104 NI=1,KIE
TT=TT+USM(JJ,NI)*TSM(NI,KK)
104 CONTINUE
TSM(JJ,KK)=TSM(JJ,KK)-TT
103 CONTINUE
105 CONTINUE
DO 110 KK=JIA,JIE
WFOCE(KK)=FOCE(KK)
110 CONTINUE
DO 112 JT=JIA,JIE
TT=0.
DO 113 KT=1,KIE
TT=TT+USM(JT,KT)*WFOCE(KT)
113 CONTINUE
WFOCE(JT)=WFOCE(JT)-TT
112 CONTINUE
9999 CONTINUE
2310 FORMAT(1H1,5X,8HERROR 10,3I5)
8000 CONTINUE
RETURN
END

```

- 150-

- 160-

- 170-

- 180-

```

C      SUBROUTINE AMINV(NP3)
C      BSEK06
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),HWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),AM (60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
DO 2000 K2000=1,NP3
W=AM(K2000,K2000)
WW=ABS(W)
      IF(WW.LE.0.00001) GO TO 2000
2001 DO 1700 K1700=1,NP3
1700 AM(K2000,K1700)=AM(K2000,K1700)/W
      DO 2200 K2200=1,NP3
      IF(K2200-K2000)1800,2200,1800
1800 DO 2100 K2100=1,NP3
      IF(K2100-K2000)2900,2100,2900
2900 AM(K2200,K2100)=AM(K2200,K2100)-AM(K2200,K2000)*AM(K2000,K2100)
2100 CONTINUE
2200 CONTINUE
      AM(K2000,K2000)=1./W
      DO 1930 K1930=1,NP3
      IF(K1930-K2000)3000,1930,3000
3000 AM(K1930,K2000)=-AM(K1930,K2000)/W
1930 CONTINUE
2000 RETURN
      END

```

- 10-

- 20-

```

SUBROUTINE BACH
C
C BSEKOT
TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NHAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /A1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
- 10-
INTEGER OW,PS
DOUBLE PRECISION I10,I20,I30,I40,I50
DRUM DIMENSION DELD(4000),NAPAR(900),DNWAT(9000)
* DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
DIMENSION DEF(300,2),WFOCE1(60)
EQUIVALENCE (ADEF,DEF)
DO 10 I11=1,NU
I1=NU-I11+1
KIE=ICAB(I1)
JIA=ICBA(I1)
- 20-
JZ=NHAT(I1)/100
IZ=NHAT(I1)-JZ*100
JIE=JZ
KZ=IZ+JZ
NTK=60*(I1-1)+1
DRUM READ DAPAR(NTK),NAPAR
IF(ISWI,NE,1) GO TO 130
WRITE(50,1040) (ICAB(L1),L1=1,15),(ICBA(L1),L1=1,15)
1040 FORMAT(1H ,1015)
WRITE(50,1000) (NAPAR(L1),L1=1,60)
- 30-
1000 FORMAT(1H ,2015)
130 CONTINUE
IF(I1,NE,NU) GO TO 20
BACKSPACE 81
BACKSPACE 81
READ(81) ((TSM(J1,J2),J1=1,60),J2=1,60)
READ(81) (WFOCE(J1),J1=1,60)
BACKSPACE 82
BACKSPACE 82
READ(82) (WFOCE1(I1),I1=1,60)
- 40-
READ(82) ((USM(I1,I2),I1=1,60),I2=1,60)
DO 100 K100=1,KIE
WFOCE(K100)=WFOCE1(K100)
100 CONTINUE
GO TO 30
20 CONTINUE
BACKSPACE 81
BACKSPACE 81
BACKSPACE 81
BACKSPACE 81

```

```

READ(81) ((TSM(J1,J2),J1=1,60),J2=1,60)
READ(81) (WFOCE(J1),J1=1,60)
BACKSPACE 82
BACKSPACE 82
BACKSPACE 82
BACKSPACE 82
READ(82) (WFOCE1(I1),I1=1,60)
READ(82) ((USM(I1,I2),I1=1,60),I2=1,60)
1020 FORMAT(1H,5F10.5)
DO 110 K110=1,K1E

DO 120 K120=J1A,J1E
TSM(K110,K120)=USM(K110,K120)
120 CONTINUE
WFOCE(K110)=WFOCE1(K110)
110 CONTINUE
DRUM READ DRDEF,ADEF
1030 FORMAT(1H,2F15.6)
DO 42 NA=1,K1E
T1=0.00
DO 43 KA=J1A,J1E

N3=NAPAR(KA)/10
NXYZ=NAPAR(KA)-N3*10
TT=TT+TSM(NA,KA)*DEF(N3,NXYZ)
43 CONTINUE
N2=NAPAR(NA)/10
NXYZ=NAPAR(NA)-N2*10
DEF(N2,NXYZ)=WFOCE(NA) -TT
42 CONTINUE
IF(IZ,E0,0) GO TO 410
NNB=J1E+1

DO 73 NB=NNB,KZ
T1=0.00
T2=0.00
DO 72 NA=1,K1E
N4=NAPAR(NA)/10
IXY=NAPAR(NA)-N4*10
T1=T1+TSM(NB,NA)*DEF(N4,IXY)
72 CONTINUE
DO 74 NC=J1A,J1E
N6=NAPAR(NC)/10

KXY=NAPAR(NC)-N6*10
T2=T2+TSM(N3,NC)*DEF(N6,KXY)
74 CONTINUE
TT=T1+T2
N5=NAPAR(NB)/10
JXY=NAPAR(NB)-N5*10
DEF(N5,JXY)=DEF(N5,JXY)+TT
73 CONTINUE
410 CONTINUE
DRUM WRITE DRDEF,ADEF

```

- 50-

- 60-

- 70-

- 80-

- 90-

41 CONTINUE

GO TO 10

30 CONTINUE

DO 80 I=1,2

DO 80 J=1,NP

DEF(J,I)=0.

80 CONTINUE

DO 52 NA=1,KIE

NA=NAPAR(NA)/10

IXY=NAPAR(NA)-NA*10

DEF(NA,IXY)=WFOCE(NA)

52 CONTINUE

IF(IZ.FQ.0) GO TO 510

NNR=JIE+1

DO 63 NR=NNB,KZ

TT=0.0

DO 62 NA=1,KIE

NA=NAPAR(NA)/10

IXY=NAPAR(NA)-NA*10

TT=TT+TSM(NB,NA)*DEF(NA,IXY)

62 CONTINUE

N5=NAPAR(NB)/10

JXY=NAPAR(NB)-N5*10

DEF(N5,JXY)=DEF(N5,JXY)+TT

63 CONTINUE

510 CONTINUE

DRUM WRITE DRDEF,ADEF

10 CONTINUE

RETURN

END

- 100-

- 110-

- 120-

- 130-

```

SUBROUTINE REAC
C   BSEK08
C   TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
C   ... REACTION PRINT
C   ... DISPLACEMENT PRINT
C   ... CAL OF UMBALANCE CHECK
C   ... NJEL : MEMBERS (UNET)
C   ... NPIN : SUPPORT NO HOKO-SHIKI
COMMON /A1/ NUE(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
*   ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJU(3)
*   COMMON /A1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
*   ,WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
INTEGER OW,PS,OPT1
DIMENSION NPIN(300),RAT(2),DEF(300,2),DDEF(300,2)
DOUBLE PRECISION I10,I20,I30,I40,I50
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
*   ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
EQUIVALENCE (ADEF,DEF), (USM,DDEF)
WRAT1=0,
WRAT2=0,
DRUM READ ALOAD,DDEF
DO 32 K=1,NP
KA=ISPU(K)
N8=0
GO TO(33,45,46,54),KA
45 CONTINUE
*   X -ROLLER
N8=2
GO TO 33
46 CONTINUE
*   Y -ROLLER
N8=1
GO TO 33
54 CONTINUE
*   PINROLLER
N8=12
33 CONTINUE
NPIN(K)=N8
32 CONTINUE
DRUM READ DRDEF,ADEF
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 200
GO TO 210
200 CONTINUE
CALL PAGE
WRITE(50,1020)
1020 FORMAT(1H0,5X,14H** REACTION **//3X,2HNP,10X,5HX(KG),10X,5HY(KG)

```

```

*      /)
210 CONTINUE
DO 100 KE=1,NP
IF(NPIN(KE).EQ.0) GO TO 100
DO 110 I=1,2
RAT(I)=0.
110 CONTINUE
N55=NPIN(KE)/10
N22=NPIN(KE)-N55*10
RAT(N22)=DEF(KE,N22)-DDEF(KE,N22)
DEF(KE,N22)=0.
IF(N55.EQ.0) GO TO 120
RAT(N55)=DEF(KE,N55)-DDEF(KE,N55)
DEF(KE,N55)=0.
120 CONTINUE
WRAT1=WRAT1+RAT(1)
WRAT2=WRAT2+RAT(2)
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) WRITE(50,1030) KE,RAT(1),
* RAT(2)
1030 FORMAT(1H ,14,2F15.5)
100 CONTINUE
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 220
GO TO 230
220 CONTINUE
WRITE(50,1040) WRAT1,WRAT2
1040 FORMAT(1H0,5HTOTAL,F14.5,F15.5)
WRITE(50,1050) WW1,WW2
1050 FORMAT(1H0,4HLOAD,2F15.5)
CALL PAGE
DRUM READ DXY,X,Y
KK=46
WRITE(50,1000)
1000 FORMAT(1H0//5X,18H** DISPLACEMENT **//21X,10HCOORDINATE,25X,
* 12HDISPLACEMENT//4X,2HNP,13X,5HX(CM),10X,5HY(CM),8X,10HDISP-X(C
*M),5X,10HDISP-Y(CM),11X,7HX-D(CM),8X,7HY-D(CM)/)
DO 61 ITT=1,NP
WF1=X(ITT)+DEF(ITT,1)
WF2=Y(ITT)+DEF(ITT,2)
IF(ITT.NE.KK) GO TO 65
CALL PAGE
WRITE(50,1000)
KK=KK+45
65 CONTINUE
WRITE(50,1010) ITT,X(ITT),Y(ITT),DEF(ITT,1),DEF(ITT,2),WF1,WF2
1010 FORMAT(1H ,15,3(3X,2F15.5))
61 CONTINUE
230 CONTINUE
DRUM WRITE DRDEF,ADDEF
10 CONTINUE
RETURN
END

```

-91-

```

AMMX1=AMMX1+AW1(I1)**2+AW2(I1)**2
200 CONTINUE
WRITE(50,3010) (IWW(KK),AW1(KK),AW2(KK),KK=1,4)
IF(ILL.NE.NP) GO TO 204
GO TO 203
202 CONTINUE
WRITE(50,3010) (IWW(KK),AW1(KK),AW2(KK),KK=1,KLL)
203 CONTINUE
AMMX1=AMMX1/FLOAT(2*NP)
IF(AMMX1.LE.EDP) KSW=1

3010 FORMAT(1H,5X,4(15,2F10.4))
WRITE(50,3040) AMMX1
3040 FORMAT(1H0//,10X,17HERROR LIMITATION=,E15.4)
C WRITE(50,6500)
6500 FORMAT(1H1,3H NE,5X,14HDAMPIND-STRESS//)
201 DRUM WRITE DDD,ADEF
DO 10 I1=1,NU
DRUM READ DRDEF,ADEF
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 300
GO TO 310

300 CONTINUE
CALL PAGE
WRITE(50,5022)
5022 FORMAT(1H0,30X,23H( ELEMENT INFORMATION ))
WRITE(50,2030) I1
2030 FORMAT(1H0,5X,9H** UNIT =,13,3H **)
WRITE(50,2020)
2020 FORMAT(1H0,3H NE,7X,7HSIGMA-X,7X,7HSIGMA-Y,8X,6HTAU-XY,7X,
* 7HTAU-MAX,7X,7HSIGMA-1,7X,7HSIGMA-2,9X,5HTHETA,8X,6HK(X/Y
*)//)

310 CONTINUE
JEND=NUE(I1)
DO 20 K1=1,JEND
NEL=NEL+1
LT=8*NEL-7
DRUM READ DELD(JT),NOEL,NEJO,EE,V,GAMMA,ICON
I1=NEJO(1)
JJ=NEJO(2)
KK=NEJO(3)
X1=X(I1)

XJ=X(JJ)
XK=X(KK)
Y1=Y(I1)
YJ=Y(JJ)
YK=Y(KK)
IF(PS.EQ.1) GO TO 30
C1=EE*(1.-V)/((1.+V)*(1.-2.*V))
C2=V/(1.-V)
GO TO 40
30 CONTINUE

```

- 100-

```

C1=EE/(1.-V*V)
C2=V
40 CONTINUE
F1=(1.-C2)/2.
C0=C1/((XJ-XI)*YK+(XI-XK)*YJ+(XK-XJ)*YI)
WK(1,1)=(YK-YJ)*C0
WK(1,2)=C2*(XJ-XK)*C0
WK(1,3)=(YI-YK)*C0
WK(1,4)=(-C2)*(XI-XK)*C0
WK(1,5)=(YI-YJ)*(-C0)

```

- 110-

```

WK(1,6)=C2*(XI-XJ)*C0
WK(2,1)=C2*(YK-YJ)*C0
WK(2,2)=(XJ-XK)*C0
WK(2,3)=C2*(YI-YK)*C0
WK(2,4)=(XI-XK)*(-C0)
WK(2,5)=(-C2)*(YI-YJ)*C0
WK(2,6)=(XI-XJ)*C0
WK(3,1)=F1*(XJ-XK)*C0
WK(3,2)=F1*(YK-YJ)*C0
WK(3,3)=(-F1*(XI-XK)*C0)

```

- 120-

```

WK(3,4)=F1*(YI-YK)*C0
WK(3,5)=F1*(XI-XJ)*C0
WK(3,6)=(-F1*(YI-YJ)*C0)
LLL=0
DO 50 N11=1,3
NO=NEJO(N11)
DO 50 M11=1,2
LLL=LLL+1
WT(LLL)=DEF(NO,M11)
50 CONTINUE

```

- 130-

```

DO 60 JA=1,3
TT=0.0
DO 65 JA=1,6
TT=TT+WK(JA,JA)*WT(JA)
65 CONTINUE
SIGMA(JA)=TT
60 CONTINUE
S1=(SIGMA(1)+SIGMA(2))*0.5
S2=SQRT(((SIGMA(1)-SIGMA(2))*0.5)**2+SIGMA(3)**2)
SIG1=S1+S2

```

- 140-

```

SIG2=S1-S2
THETA=2A.647889*(ATAN(2.0*SIGMA(3)/(SIGMA(1)-SIGMA(2))))
IF(SIGMA(1)-SIGMA(2).LE.0.0)THETA=THETA+90.
AS2=ABS(SIGMA(2))
GX1=9999.9999
IF(AS2.GE.0.00001) GX1=SIGMA(1)/SIGMA(2)
G1(NOEL)=SIGMA(1)
G2(NOEL)=SIGMA(2)
G3(NOEL)=SIGMA(3)
G4(NOEL)=S2

```

```

- 150-
G5(NOEL)=SIG1
G6(NOEL)=SIG2
G7(NOEL)=THETA
G8(NOEL)=GK1
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1)
  *WRITE(50,1000) NOEL,SIGMA(1),SIGMA(2),SIGMA(3),S2,SIG1,SIG2,THETA,
1    GK1
1000 FORMAT(1H ,I3,8F14.6)
20 CONTINUE
10 CONTINUE

- 160-
KKK=51
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 320
GO TO 330
320 CONTINUE
CALL PAGE
WRITE(50,9021)
5021 FORMAT(1H0,30X,33H( MEAN VALUE AROUND NODAL POINT ))
WRITE(50,2000)
2000 FORMAT(1H0,3H NP,7X,8HSIGMA-X ,6X,8HSIGMA-Y ,6X,8HTAU-XY ,6X,8HTA
1U-MAX ,6X,8HSIGMA-1 ,6X,8HSIGMA-2 ,8X,5HTHETA,7X,7HK (X/Y)/)

- 170-
330 CONTINUE
DO 100 K100=1,NP
  SIGMA(1)=0.
  SIGMA(2)=0.
  SIGMA(3)=0.
  S2=0.
  SIG1=0.
  SIG2=0.
  THETA=0.
  GK1=0.

- 180-
JJ=0
DO 110 K110=1,8
  KK=NEP(K100,K110)
  IF(KK.EQ.0) GO TO 120
  JJ=JJ+1
  SIGMA(1)=SIGMA(1)+G1(KK)
  SIGMA(2)=SIGMA(2)+G2(KK)
  SIGMA(3)=SIGMA(3)+G3(KK)
  S2=S2+G4(KK)
  SIG1=SIG1+G5(KK)

- 190-
SIG2=SIG2+G6(KK)
THETA=THETA+G7(KK)
GK1=GK1+G8(KK)
110 CONTINUE
120 CONTINUE
GK2=FLOAT(JJ)
GK2=1./GK2
SIGMA(1)=SIGMA(1)*GK2
SIGMA(2)=SIGMA(2)*GK2
SIGMA(3)=SIGMA(3)*GK2

```

- 200-

```

S2=S2*GK2
SIG1=SIG1*GK2
SIG2=SIG2*GK2
THETA=THETA*GK2
GK1=GK1*GK2
IF((CYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 340
GO TO 350
340 CONTINUE
IF(K100.NE.KKK) GO TO 130
CALL PAGE

WRITE(50,5021)
WRITE(50,2000)
KKK=KKK*50
130 CONTINUE
WRITE(50,2010) K100,SIGMA(1),SIGMA(2),SIGMA(3),S2,SIG1,SIG2,THETA,
* GK1
2010 FORMAT(1H ,I3,6F14.6,6F14.3,F14.4)
350 CONTINUE
100 CONTINUE
IF(LINF.EQ.1) GO TO 220

```

- 210-

- 220-

```

5020 FORMAT(1H ,I5)
CALL PAGE
WRITE(50,3020)
DO 140 K140=1,NE
JT=R*K140-7
DRUM READ DELD(JT),NOEL,NFJO,EE,V,GAMMA,ICON
NEL1=NEL1+1
NEL2=NEL2+1
NOEL1(NEL1)=NOEL
EW(NEL1)=EE

```

- 230-

```

IF(NEL2.EQ.NE) GO TO 361
IF(NEL1.LT.7) GO TO 360
WRITE(50,3030) (NOEL1(I),EW(I)),I=1,7)
NEL1=0
GO TO 360
3030 FORMAT(1H ,5X,7(I5,F10.2))
361 CONTINUE
WRITE(50,3030) (NOEL1(I),EW(I)),I=1,NEL1)
360 CONTINUE
KKK=ICON*2-1

```

- 240-

```

NN2=NN1-1
IF((CYCL.EQ.1) GO TO 480
G2(NOEL)=EDP2*(G2(NOEL)-G22(NOEL))+G22(NOEL)
480 CONTINUE
DO 150 K150=1,NN2
IF(ALP(K150,KKK).LE.G2(NOEL).AND.ALP(K150+1,KKK).GE.G2(NOEL)) GO T
10 160
150 CONTINUE
160 R3=(ALP(K150+1,KKK+1)-ALP(K150,KKK+1))/(ALP(K150+1,KKK)
1 -ALP(K150,KKK))

```

- 250-

```

5010 FORMAT(1H,5E13.4,15)
      R6=R5*G2(NOEL)+(R5*(-ALP(K150,KKK))+ALP(K150,KKK+1))
      EE=G2(NOEL)/R6
      DRUM WRITE DELD(JT),NOEL,NEJO,EE,V,GAMMA,ICON
140  CONTINUE
220  CONTINUE
      DO 470 K470=1,NE
          G22(K470)=G2(K470)
C      WRITE(50,6510) K470,G2(K470)
6510  FORMAT(1H,13,F20.5)
470  CONTINUE

```

- 260-

```

      IF(KSW.E0.0) GO TO 400
      DRUM READ DDD,ADEF
      CALL PAGE
      DRUM READ DXY,X,Y
      KK=46
      WRITE(50,6000)
6000  FORMAT(1H0//5X,18H** DISPLACEMENT **//21X,10HCOORDINATE,25X,
*      12HDISPLACEMENT//4X,2HNP,13X,5HX(CH),10X,5HY(CM),8X,10HDISP-X(C
*H),5X,10HDISP-Y(CH),11X,7HX-D(CH),8X,7HY-D(CM)//
      DO 461 ITT=1,NP

```

- 270-

```

      WF1=X(ITT)+DEF(ITT,1)
      WF2=Y(ITT)+DEF(ITT,2)
      IF(ITT.NE.KK) GO TO 465
      CALL PAGE
      WRITE(50,6000)
      KK=KK+45
465  CONTINUE
      WRITE(50,6010) ITT,X(ITT),Y(ITT),DEF(ITT,1),DEF(ITT,2),WF1,WF2
6010  FORMAT(1H,15,3(3X,2F15.5))
461  CONTINUE

```

- 280-

```

400  CONTINUE
      DO 444 ITT=1,NP
          X(ITT)=X(ITT)+DEF(ITT,1)
          Y(ITT)=Y(ITT)+DEF(ITT,2)
444  CONTINUE
      DRUM WRITE DXY,X,Y
      IF(ICYCL.E0,IPHAN) CALL MTWT
      RETURN
      END

```

```

C      SUBROUTINE PAGE
C      BSEK010
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A6/ICYCL
DOUBLE PRECISION I10,I20,I30,I40,I50
NPAGE=NPAGE+1
WRITE(50,1000) NPAGE,I10,I20,I30,I40,I50,ICYCL
1000 FORMAT(1H1/90X,5HPAGE=,I3,/5X,5A8,10X,6HCYCLE=,I3)
RETURN
END

```

- 10-

```

C      SUBROUTINE INDI(NEN,NW)
C      TWO-DIMENSIONAL ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
      BSEK011
      COMMON /AA1/ X(300),Y(300),WK(6,6),
1          USM(60,60),FOCE(60),TSM(60,60),
2          WFOCE(60),ADEF(600),ISPU(300)
      DO 10 I=1,NW
      FOCE(I)=FOCE(I)+USM(1,11)*WFOCE(I1)
10  CONTINUE
      RETURN
      END

```

- 10-

第2章 軸対称問題解析用プログラム

§ 2-1 概 要

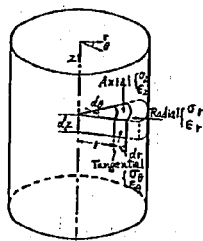
軸対称問題では円筒座標系を用いると記述が簡単である。円筒座標系では変位とひずみの関係式は次式で表わされる。

$$\left. \begin{aligned} \epsilon_r &= \frac{\partial u}{\partial r}, & \epsilon_\theta &= \frac{u}{r} + \frac{\partial v}{r \partial \theta} \\ \epsilon_z &= \frac{\partial w}{\partial z} \\ \gamma_{rz} &= \frac{\partial u}{\partial z} + \frac{\partial w}{\partial r} \\ \gamma_{z\theta} &= \frac{\partial v}{\partial z} + \frac{\partial w}{r \partial \theta} \\ \gamma_{r\theta} &= \frac{\partial u}{r \partial \theta} + \frac{\partial v}{\partial r} - \frac{v}{r} \end{aligned} \right\} \dots\dots\dots (2-1)$$

ここに u, v, w はそれぞれ半径方向、回転方向および軸方向の変位を表わす。

いまねじり成分を考えなければ回転方向には無関係となり回転方向変位は発生しない。そして上式は次のように書き換える事が出来る。

$$\left. \begin{aligned} \epsilon_r &= \frac{\partial u}{\partial r}, & \epsilon_\theta &= \frac{u}{r} \\ \epsilon_z &= \frac{\partial w}{\partial z} \\ \gamma_{rz} &= \frac{\partial u}{\partial z} + \frac{\partial w}{\partial r} \\ \gamma_{z\theta} &= 0, & \gamma_{r\theta} &= 0 \end{aligned} \right\} \dots\dots\dots (2-2)$$



このように変位としては軸方向変位と半径方向変位のみを考えればよいことになる。そしてこの問題は二次元状態であるから平面解析に用いた変位関数および要素がそのまま使えることになる。

しかし軸対称状態では式(2-2)から分るように半径方向変位に供なって自動的に回転方向のひずみが生じることになる。そしてこれに関連した回転方向の応力成分が発生するのでこの成分を考慮した剛性計算を行なう必要がある。

その剛性マトリクス K は式(2-4)で与えられる。

計算に用いた主な計算式を以下に示す。

等方性体の応力とひずみの関係式

$$\{\sigma\} = \begin{Bmatrix} \sigma_z \\ \sigma_r \\ \sigma_\theta \\ \tau_{rz} \end{Bmatrix} = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \left[\begin{array}{ccc|ccc} 1 & & & & & \\ \nu & 1 & & & & \\ \nu & \nu & 1 & & & \\ 0 & 0 & 0 & 1 & & \\ & & & & \frac{1-2\nu}{2(1-\nu)} & \end{array} \right] \begin{Bmatrix} \epsilon_z \\ \epsilon_r \\ \epsilon_\theta \\ \gamma_{rz} \end{Bmatrix} \quad (2-3)$$

または

$$\{\sigma\} = \begin{bmatrix} C_{22} & C_{12} & C_{12} & 0 \\ C_{12} & C_{11} & C_X & 0 \\ C_{12} & C_X & C_{11} & 0 \\ 0 & 0 & 0 & C_{33} \end{bmatrix} \begin{Bmatrix} \varepsilon_z \\ \varepsilon_r \\ \varepsilon_\theta \\ r_{rz} \end{Bmatrix}$$

と表わす。

剛性マトリクス

$$K = 2\pi * [A^{-1}]^T [k] [A^{-1}] \dots\dots\dots (2-4)$$

ここに

$$[k] = \begin{bmatrix} C_{11} I_4 & & & & & \\ (CX+C_{11}) I_2 & 2(C_{11}+CX) I_2 & & & & \\ C_{11} I_5 & (C_{11}+CX) I_3 & C_{11} I_6 + C_{33} I_2 & & & \\ 0 & 0 & 0 & 0 & & \\ 0 & 0 & C_{33} I_2 & 0 & C_{33} I_2 & \\ C_{12} I_1 & 2C_{12} I_2 & C_{12} I_3 & 0 & 0 & C_{22} I_2 \end{bmatrix} \quad \text{対 称}$$

$$[A^{-1}] = \frac{1}{2I_1} * \begin{bmatrix} r_j z_k - r_i z_j & 0 & r_k z_i - r_i z_k & 0 & r_i z_j - r_j z_i & 0 \\ z_{jk} & 0 & z_{ki} & 0 & z_{ij} & 0 \\ r_{kj} & 0 & r_{ik} & 0 & r_{ji} & 0 \\ 0 & r_j z_k - r_k z_j & 0 & r_k z_i - r_i z_k & 0 & r_i z_j - r_j z_i \\ 0 & z_{jk} & 0 & z_{ki} & 0 & z_{ij} \\ 0 & r_{kj} & 0 & r_{ik} & 0 & r_{ji} \end{bmatrix}$$

$$I_1 = \iint dr dz = \frac{1}{2} (r_{ji} * z_{ki} - r_{ki} * z_{ji})$$

$$I_2 = \iint r dr dz = (r_i + r_j + r_k) * I_1 / 3$$

$$I_3 = \iint z dr dz = (z_i + z_j + z_k) * I_1 / 3$$

$$I_4 = \iint \frac{1}{r} dr dz = \sum_{\substack{\alpha=i,j,k \\ \beta=j,k,i \\ r=k,i,j}} (A_{\alpha r} - A_{r \beta}) \log_e r_r$$

$$I_5 = \iint \frac{z}{r} dr dz = \sum_{\substack{\alpha=i,j,k \\ \beta=j,k,i \\ r=k,i,j}} \left\{ \frac{1}{2} (A_{\alpha r}^2 - A_{r \beta}^2) \log_e r_r + A_{r \beta} B_{r \beta} (r_\beta - r_r) + \frac{1}{4} B_{r \beta}^2 (r_\beta^2 - r_r^2) \right\}$$

$$I_6 = \iint \frac{z^2}{r} d\tau dz = \sum_{\substack{\alpha=ijk \\ \beta=jk i \\ \gamma=ki j}} \left\{ \frac{1}{3} (A_{\alpha\gamma}^3 - A_{\gamma\beta}^3) \log_e r_\gamma + A_{\gamma\beta}^2 B_{\gamma\beta} (r_\beta - r_\gamma) \right. \\ \left. + \frac{1}{2} A_{\gamma\beta} B_{\gamma\beta}^2 (r_\beta^2 - r_\gamma^2) + \frac{1}{9} B_{\gamma\beta}^3 (r_\beta^3 - r_\gamma^3) \right\}$$

$$A_{\alpha\gamma} = \frac{r_\gamma z_\alpha - r_\alpha z_\gamma}{r_\gamma - r_\alpha}, \quad B_{\gamma\beta} = \frac{z_\beta - z_\gamma}{r_\beta - r_\gamma}$$

$$r_\beta = r_\gamma \text{ なるとき } A_{\gamma\beta} = B_{\gamma\beta} = 0$$

$$r_{ij} = r_i - r_j, \quad z_{ij} = z_i - z_j \text{ etc.}$$

ひずみマトリクス

$$\{\epsilon\} = \begin{Bmatrix} \epsilon_z \\ \epsilon_r \\ \epsilon_\theta \\ r_{rz} \end{Bmatrix} = \frac{1}{2I_1} \begin{Bmatrix} 0 & r_{kj} & 0 & r_{ik} & 0 \\ z_{jk} & 0 & z_{ki} & 0 & z_{ij} \\ \frac{a_i^1}{r} + z_{jk} + r_{kj} \frac{z}{r} & 0 & \frac{a_j^1}{r} + z_{ki} + r_{ik} \frac{z}{r} & 0 & \frac{a_k^1}{r} + z_{ij} + r_{ji} \frac{z}{r} \\ r_{kj} & z_{jk} & r_{ik} & z_{ki} & r_{ji} \end{Bmatrix} \begin{Bmatrix} u_i \\ v_i \\ u_j \\ v_j \\ u_k \\ v_k \end{Bmatrix} \quad (2-5)$$

$$\text{ここに } a_i^1 = r_j z_k - r_k z_j$$

$$a_j^1 = r_k z_i - r_i z_k$$

$$a_k^1 = r_i z_j - r_j z_i$$

応力の計算

$$\{\sigma\} = [D] \{\epsilon\} \quad (2-6)$$

ひずみおよび応力の計算で注意しなければならないことは式(2-5)から分かるように要素内でひずみが変わるので応力成分も要素内で変化するということである。

物体力を考慮する時は各節点に等価節点力をふりわけると。

$$\{F_i\} = \{F_j\} = \{F_k\} = -2\pi \{R\} \bar{r} I_1 / 3$$

ここに Z, R はそれぞれ軸方向, 半径方向の単位体積当りの物体力, $\bar{r} = (r_i + r_j + r_k) / 3$

以上の計算式を基にしたプログラムを次に示す。

このプログラムは二次元解析用プログラムと同様に線形，非線形計算に利用できる。

§ 2-2 入力データ作成の説明

絶対称用入力データシート

No.																				
1	計 算 名																			
2	NP	NE	NU	NOP	0	NOCL	FOCL	OPT1	LINE	EDP2										
3	STAIN STAY																			
4	NEXY				NFXV1															
5	NEXY				NFXV2															
6	NEXY				NFXV3															
7	NP	RD	ZD	NP	RD	ZD														
8	N	PE	PV	PG	N	PE	PV	PG												
9	NUE				NU															
10	NE	I	J	K	NOP															
11	OW	LUN			NU															
12	NP	LOADR	LOADZ	NP	LOADR	LOADZ														
13	N1		EDP																	
14	a	b	c	d	N1組															
15	ブランクカード1枚加える																			

§ 2 - 3 軸対称用プログラム

C AAXIFEM
 C AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
 C A10, A50 = KO ZO BUTSU NAME
 C NP = SO-SETTENSUU
 C NE = SO-ELEMENTSUU
 C NU = SO-UNITSUU
 C NDP = DANSEI PATTERN SUU
 C ISPN = SETTENZYOTAI
 C NUE = KAKU UNIT NO ELEMENTSUU
 C NJOI = JOINT-TABLE SAKUSEI ERIA

- 10-

C NUNJO = JOINT-TABLE SAKUSEI ERIA
 C NFIX = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C KJOI = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C NFJOI = UNIT NI FUKUMARERU KOTEI SETTEN
 C NFJOI = UNIT NI FUKUMARERU KOTEI SETTEN TABLE ERIA
 C NWJOI = KAKU SETTEN GA ZOKUSHITEIRU
 C NWJO = KAKU SETTEN GA ZOKUSHITEIRU UNIT BANGOTABLE
 C NCLS = KAKU SETTEN GA ZOKUSHITEIRU
 C NCLS = KAKU SETTEN GA ZOKUSHITEIRU UNIT BANGOTABLE
 C JTBL = CLASS BUNRUITABLE

- 20-

C ITBL = CLASS BUNRUITABLE
 C LUN = KAKU UNIT NO KAZYUSUU
 C INOP = KAKU ELEMENT NO PATERNBANGO
 C MI = ELEMENT BANGO
 C MI = ELEMENT NO I.J.K YO-SO
 C MJ = ELEMENT NO I.J.K YO-SO
 C MK = ELEMENT NO I.J.K YO-SO
 C GAMMA = HIZYU
 C EE = YOUNG KEISU
 C V = POISPN HI

- 30-

C X = R-ZAHYO TI
 C Y = Z-ZAHYO TI
 C NFXV = KOTEI-ROALER SETTENBANGO
 C XD = ZAHYOTI NO WARK-ERIA OYOBI KAJYU YOMIKOMI ERIA
 C YD = ZAHYOTI NO WARK-ERIA OYOBI KAJYU YOMIKOMI ERIA
 C NTBL = SETTEN ZYOTAI TABLE
 C NWAT = KYOKAI ZYOKEN SYORI TABLE
 C NWAT = KAKU UNIT NO ZIYUDOSUU
 C NPAR = KYOKAI SYORI GONO INDEX SUU
 C NPAR = KYOKAI SYORI GONO INDEX SUU

- 40-

C PE = KAKU PATERN NO YOUNGKEISUU*POISONHI*HIZYU
 C PV = KAKU PATERN NO YOUNGKEISUU*POISONHI*HIZYU
 C PG = KAKU PATERN NO YOUNGKEISUU*POISONHI*HIZYU
 C ALP = HI SENKEI KYOKUSEN DATA
 C NEE = KAKU SETTEN NO KESTUGO ELEMENT BANGO TABLE
 C NEW = NEE SAKUSEI NOTAMENO WARK ERIA
 C IYO = WARK ERIA
 C NFXV1 = KOTEI SETTEN SUU
 C NFXV2 = KOTEI SETTEN SUU
 C NFXV3 = KOTEI SETTEN SUU

```

DIMENSION ISPU(300),NUE(15),NJOL(15),NUNJO(15,30)
*,NFI(15,25),KJOL(15),NFJOL(15),NWJO(300),NCLS(300,4),ITBL(15,4)
DIMENSION LUN(15),INOP(500),IWO(3),NEE(300,12),NEW(500)
*,GAMMA(500),EE(500),V(500),X(500),Y(500),NMAT(15)
*,JTBL(15,120),HTABL(4,2),NWAT(300,2),NAPAR(60),NNPAR(60)
DIMENSION MFX(300),XD(300),YD(300),PE(15),PV(15),PG(15),
1ALP(20,20),MI(500),MJ(500),MK(500)
DOUBLE PRECISION A10,A20,A30,A40,A50
INTEGER PS,OW,PCYCL,OPT1
READ(40,99)A10,A20,A30,A40,A50
- 50-

99 FORMAT(5A8)
READ(40,100) NP,NE,NU,NOP,ISW1,MCYCL,PCYCL,OPT1,LINE,EDP2
WRITE(50,800) NP,NE,NU,NOP
800 FORMAT(1H0,3HNP=,15,3X,3HNE=,15,3X,3HNU=,15,3X,
14HNOP=,15)
100 FORMAT(9I5,F10,1)
IF(NP.LE.300) GO TO 900
111=1
GO TO 999
900 IF(NE.LE.500) GO TO 901
- 60-

111=2
GO TO 999
901 IF(NU.LE.15) GO TO 902
111=3
GO TO 999
902 READ(40,101)MFX1,MFX2,MFX3
REWIND 81
WRITE(50,801)MFX1,MFX2,MFX3
801 FORMAT(1H0,6HMFXY1=,13,3X,6HMFXY2=,13,3X,6HMFXY3=,13)
101 FORMAT(3I3)
- 70-

DO 50 I=1,NP
50 ISPU(I)=1
IF(MFX1(I).NE.1)
102 FORMAT(20I4)
10 READ(40,102) (MFX(I),I=1,MFX1)
WRITE(50,802) (MFX(I),I=1,MFX1)
802 FORMAT(2X,22H* KOTEI SETTEN BANGO *.20I4)
DO 51 I=1,MFX1
51 NN=MFX(I)
ISPU(NN)=4
- 80-

51 CONTINUE
9 IF(MFX2)12,11,12
12 READ(40,102) (MFX(I),I=1,MFX2)
DO 52 I=1,MFX2
52 NN=MFX(I)
ISPU(NN)=2
51 CONTINUE
WRITE(50,3000)
3000 FORMAT(1H0,5H** 1 )
11 IF(MFX3)13,14,13
- 90-

```

```

13 READ(40,102) (MFX(Y(I),I=1,NFXY3)
WRITE(50,803) (MFX(Y(I),I=1,NFXY3)
803 FORMAT(2X,12H* Y-ROLLER *2014)
WRITE(50,4000)
4000 FORMAT(2X,20H* SETTEN ZAHYO CHI *)
DO 53 I=1,NFXY3
NN=MFX(Y(I)
ISPU(NN)=3
53 CONTINUE
14 READ(40,103) (MFX(Y(I),XD(I),YD(I),I=1,NP)
WRITE(50,804) (MFX(Y(I),XD(I),YD(I),I=1,NP)
804 FORMAT(4(14,2F8.3))
103 FORMAT(4(14,2F8.3))
DO 54 I=1,NP
NN=MFX(Y(I)
X(NN)=XD(I)
Y(NN)=YD(I)
54 CONTINUE
READ(40,104) (PE(I),PV(I),PG(I),I=1,NOP)
WRITE(50,805) (PE(I),PV(I),PG(I),I=1,NOP)
805 FORMAT(2X,17H* DANSEI PATARN *,3(3X,3F7.0),8X)
WRITE(50,3002)
3002 FORMAT(1H0,27H* SETTEN NO SESSYOKU UNIT *)
104 FORMAT(3(3X,3F7.0),8X)
905 NSUM=0
READ(40,106) (NUE(I),I=1,NU)
106 FORMAT(2014)
DO 58 I=1,NU
58 NSUM=NSUM+NUE(I)
IF(NSUM,EQ,NE) GO TO 906

111=7
GO TO 999
906 CONTINUE
DO 621 KK=1,NE
READ(40,107) K1,K2,K3,K4,K5
NEW(K2)=NEW(K2)+1
NEW(K3)=NEW(K3)+1
NEW(K4)=NEW(K4)+1
NN1=NEW(K2)
NN2=NEW(K3)
NN3=NEW(K4)
NEE(K2,NN1)=K1
NEE(K3,NN2)=K1
NEE(K4,NN3)=K1
H1(KK)=K1
H2(KK)=K2
H3(KK)=K3
H4(KK)=K4
INOP(KK)=K5
IF(0.LE,K2.AND,K2.LT.NP) GO TO 621

```

- 100-

- 110-

- 120-

- 130-

- 140-

```

111=4
GO TO 999
IF(0.LE.K3.AND.K3.LT.NP) GO TO 621
111=5
GO TO 999
IF(0.LE.K4.AND.K4.LT.NP) GO TO 621
111=6
GO TO 999
621 CONTINUE
WRITE(50,3510) ((NEE(L1,L2),L2=1,12),L1=1,NP)
- 150-

3510 FORMAT(1H ,12I5)
WRITE(50,4001)
- 160-
4001 FORMAT(1H0,11H* ELEMENT *)
WRITE(50,107) (M1(I),M1(I),MJ(I),MK(I),INOP(I),I=1,NE)
WRITE(50,4002)
4002 FORMAT(1H0,24H* NUNJO AND NJOI TABLE *)
107 FORMAT(20I4)
DO 59 I=1,NE
NNN=INOP(I)
GAMMA(I)=PG(NNN)
- 170-

EE(I)=PE(NNN)
V(I)=PV(NNN)
59 CONTINUE
TABLE=1 SAKUSEI
DO 60 I=1,NU
C 60 NJOI(I)=0
IEND=0
ISTA=1
DO 61 IUNIT=1,NU
IEND=IEND+IUE(IUNIT)
- 180-

DO 62 II=ISTA,IEND
IF(0.LT.NJOI(IUNIT)) GO TO 90
NUNJO(IUNIT,1)=M1(ISTA)
NUNJO(IUNIT,2)=MJ(ISTA)
NUNJO(IUNIT,3)=MK(ISTA)
NJOI(IUNIT)=3
GO TO 62
90 IWO(1)=M1(II)
IWO(2)=MJ(II)
IWO(3)=MK(II)
- 190-

DO 64 IB=1,3
IN=IWO(IB)
IKE=NJOI(IUNIT)
DO 65 IP=1,IKE
IF(IN.EQ.NUNJO(IUNIT,IP)) GO TO 64
65 CONTINUE
IKE=IKE+1
NUNJO(IUNIT,IKE)=IN
NJOI(IUNIT)=IKE
64 CONTINUE

```

```

62 CONTINUE
61 ISTA=ISTA+NUE(IUNIT)
C   TABLE-1 OUT PUT(TEST)
DO 400 I=1,NU
400 WRITE(50,500)(NUNJO(I,J),J=1,30)
WRITE(50,501)(NJOI(I),I=1,NU)
500 FORMAT(1X,10I4)
501 FORMAT(///,1X,4I4,/,1H1)
NZ=NZ+1
NWAT(ISTA*NE)=NZ
- 210-

C   TABLE-2 SAKUSEI
WRITE(50,4004)
4004 FORMAT(1H0,24H* NUNJO AND KJOI TABLE *)
DO 66 I=1,NU
NX=0
IT=0
JEND=NJOI(I)
DO 67 J=1,JEND
II=NUNJO(I,J)
IK=ISPU(II)

IF(II*NE.4) GO TO 91
NX=NX+1
NFIX(I,NX)=II
NUNJO(I,J)=0
GO TO 67
91 IT=IT+1
NUNJO(I,IT)=II
67 CONTINUE
KJOI(I)=IT
66 NFJOI(I)=NX
- 220-

DO 401 I=1,NU
401 WRITE(50,500)(NUNJO(I,J),J=1,30)
WRITE(50,501)(KJOI(I),I=1,NU)
WRITE(50,4040)
4040 FORMAT(1H0,24H* NFIX AND NFJOI TABLE *)
DO 402 I=1,NU
402 WRITE(50,500)(NFIX(I,J),J=1,25)
WRITE(50,501)(NFJOI(I),I=1,NU)
C   TABLE-2 OUT PUT(TEST)
C   TABLE-3 SAKUSEI
- 230-

WRITE(50,4003)
4003 FORMAT(1H0,23H* NCLS AND NWJO TABLE *)
DO 68 I=1,NP
68 NWJO(I)=0
DO 69 I=1,NU
DO 69 J=1,4
69 ITRL(I,J)=0
DO 70 IK=1,NU
JP=KJOI(IK)
DO 70 JK=1,JP
- 240-

```

```

- 250-
      II=NUNJO(IK,JK)
      IF(NWJO(II).NE.0) GO TO 92
      NWJO(II)=1
      NCLS(II,1)=IK
      GO TO 94
92  NWJO(II)=NWJO(II)+1
      IKK=NWJO(II)
      NCLS(II,IKK)=IK
94  CONTINUE
70  CONTINUE

- 260-
      DO 403 I=1,NP
403  WRITE(50,502)(NCLS(I,J),J=1,4),NWJO(I)
502  FORMAT(1X,4I3,5X,I2)
C    TABLE=3 OUTPUT(TEST)
C    TABLE=4 SAYUSE1
      DO 71 II=1,NP
      L=NWJO(II)
      IF(L.NE.1) GO TO 71
      KK=NCLS(II,1)
      ITBL(KK,1)=ITBL(KK,1)+1

- 270-
      KE=ITBL(KK,1)
      JTBL(KK,KE)=II
71  CONTINUE
      DO 41 KI=1,NU
      JP=KJOI(KI)
      DO 72 JI=1,JP
      II=NUNJO(KI,JI)
      JK=NWJO(II)
      IF(JK.LE.1) GO TO 72
      KAT=0

- 280-
      KMA=0
      DO 73 LP=1,JK
      KP=NCLS(II,LP)
      IF(KI.GT.KP) GO TO 76
      IF(KI.EQ.KP) GO TO 73
      KAT=1
      GO TO 73
76  KMA=1
73  CONTINUE
      IF(KAT.NE.1.OR.KMA.NE.1) GO TO 150

- 290-
      IC=3
      GO TO 151
150  IF(KAT.NE.1) GO TO 152
      IC=4
152  IF(KMA.NE.1) GO TO 151
      IC=2
151  ITBL(KI,IC)=ITBL(KI,IC)+1
      KAT=(IC-1)*30+ITBL(KI,IC)
      JTBL(KI,KAT)=II
72  CONTINUE

```

4005 WRITE(50,4005)
 FORMAT(1H0,17H* JTBL AND ITBL *)
 WRITE(50,504)(JTBL(KI,J),J=1,120)
 WRITE(50,506)(ITBL(KI,J),J=1,4)
 IF(KI.EQ.1) GO TO 41
 KKP=KI-1
 DO 155 LA=3,4
 KEN=ITBL(KKP,LA)
 IF(KEN.EQ.0) GO TO 155
 LPP=30*(LA-1)

- 310-

DO 156 LB=1,KEN
 KZ=LPP+LB
 IN=JTBL(KKP,KZ)
 DO 157 LC=2,3
 LCN=ITBL(KI,LC)
 IF(LCN.EQ.0) GO TO 157
 NPA=30*(LC-1)
 DO 159 LU=1,LCN
 K3=NPA+LU
 IF(IN.EQ.JTBL(KI,K3)) GO TO 156

- 310-

159 CONTINUE
 157 CONTINUE
 NW=NWJO(IN)
 NAE=0
 NAT=0
 DO 160 K5=1,NW
 ME=NCLS(IN,K5)
 IF(ME.LT.KI) NAE=1
 IF(ME.GT.KI) NAT=1
 160 CONTINUE

- 320-

IF(NAE.EQ.1) GO TO 910
 III=11
 GO TO 999
 910 KF=2
 IF(NAT.EQ.1) KF=3
 ITBL(KI,KF)=ITBL(KI,KF)+1
 NF=ITBL(KI,KF)
 N44=30*(KF-1)+NF
 JTBL(KI,N44)=IN
 156 CONTINUE

- 330-

155 CONTINUE
 41 CONTINUE
 DO 161 NA=1,NU
 NTO=0
 DO 162 NB=1,4
 NTO=NT0+ITBL(NA,NB)
 IF(NT0.LE.30) GO TO 161
 III=12
 GO TO 999
 161 CONTINUE

- 340-

```

C      TABLE-4 OUT PUT(TEST)
      WRITE(50,503)
503  FORMAT(1H1)
      DO 404 I=1,NU
      WRITE(50,504) (JTBL(I,J),J=1,120)
404  WRITE(50,505)
504  FORMAT(1X,30I4)
505  FORMAT(/)
      DO 405 I=1,NU
405  WRITE(50,506) (ITBL(I,J),J=1,4)
                                     - 350-

506  FORMAT(1X,4I4)
C      ***M/T READ-1***
      WRITE (81) A10,A20,A30,A40,A50
      WRITE(81) NP,NE,NU,NOP,ISW,MCYCL,PCYCL,OPT1,LINE,EDP2
      WRITE (81) (ISPU(I),I=1,NP)
      WRITE (81) (X(I),I=1,NP)
      WRITE (81) (Y(I),I=1,NP)
      WRITE(81) (NUE(I),I=1,NU)
      WRITE(81) NEE
      DO 601 I=1,NE
                                     - 360-

      L1=M1(I)
      L2=M1(I)
      L3=MJ(I)
      L4=MK(I)
      ICON=INOP(I)
      G=GAMMA(I)
      E=EE(I)
      VV=V(I)
      WRITE(81) L1,L2,L3,L4,E,VV,G,ICON
      601 CONTINUE
                                     - 370-

      WRITE (81) ((JTBL(I,J),I=1,NU),J=1,120)
      WRITE (81) ((ITBL(I,J),I=1,NU),J=1,4)
C      KYOKAIJYOKEN SHORI NO TABLE SAKUSEI
      DO 163 KT=1,4
      DO 163 LAA=1,2
163  NTABL(KT,LAA)=0
      NTABL(2,2)=-1
      NTABL(3,1)=-1
      DO 164 II=1,NU
      DO 165 JK=1,NP
                                     - 380-

      DO 165 LK=1,2
165  NWAT(JK,LK)=0
      NZ=0
      NZZ=0
      DO 166 JJ=1,4
      NK=ITBL(II,JJ)
      IF(NK,EQ,0) GO TO 166
      MH=30*(JJ-1)
      DO 167 KK=1,NK
      LA=MH+KK
                                     - 390-

```

```

- 400-
      INTA=JTBL(I1,LA)
      NLP=ISPU(INTA)
      DO 168 NE=1,2
      KE=NTABL(NLP,NE)
      IF(KE.EQ.-1) GO TO 169
      NZ=NZ+1
      NWAT(INTA,NE)=NZ
      NAPAR(NZ)=INTA*10+NE
      GO TO 168
169  NZZ=NZ+1
      NNPAP(NZZ)=INTA*10+NE
168  CONTINUE
167  CONTINUE
166  CONTINUE
      IF(NFJO1(I1).EQ.0) GO TO 170
      NJ=NFJO1(I1)
      DO 171 I=1,NJ
      DO 171 J=1,2
      NZZ=NZZ+1
171  NNPAP(NZZ)=NF[X(I1),J]*10+J
      NNN=NZ+1
      IF(NZZ.EQ.0) GO TO 172
      DO 173 I=1,NZZ
      NNN=NZ+1
      NA=NNPAP(I)/10
      NB=NNPAP(I)-NA*10
      NWAT(NA,NB)=NNN
173  NAPAR(NNN)=NNPAP(I)
172  CONTINUE
      WRITE (81) ((NWAT(I,J),I=1,NP),J=1,2)
      WRITE (81) (NAPAR(I),I=1,60)
C      KYOKAIJYOKEN TABLE OUTPUT(TEST)
      WRITE(50,503)
      WRITE(50,4006)
4006  FORMAT(1H0,32H* NTABL , NWAT AND NAPAR TABLE *)
      WRITE(50,507) ((NTABL(I,J),J=1,2),I=1,4)
507  FORMAT(1X,213,/)
      WRITE(50,508) ((NWAT(I,J),J=1,2),I=1,NP)
508  FORMAT(1X,214)
      WRITE(50,509) (NAPAR(I),I=1,NNN)
- 440-
509  FORMAT(1X,2514)
      NMAT(I1)=NZ*100+NZZ
164  CONTINUE
      WRITE (81) (NMAT(I),I=1,NU)
      NNN=0
      READ(40,700) OW,(LUN(I),I=1,NU)
700  FORMAT(13,1514)
      DO 200 I=1,NU
      NN1=LUN(I)/100
      NN2=LUN(I)-NN1*100

```

```

NNN=NNN+NN1+NN2
200 CONTINUE
IF(NNN.EQ.0) GO TO 300
READ(40,710) (MFXY(I),XD(I),YD(I),I=1,NNN)
710 FORMAT(4(14,2F8.3))
WRITE(50,3004)
3004 FORMAT(1H0,5H** 4 )
300 CONTINUE
IF(LINE.EQ.1) GO TO 5000
READ(40,720) N1,EDP

DO 210 I=1,NOP
KK=2*I-1
720 FORMAT(15,E10.0)
N10=2*N1
READ(40,730) (X(I),I=1,N10)
DO 211 I=1,N1
I0=2*I-1
ALP(I,KK)= X(I0)
ALP(I,KK+1)= X(I0+1)
211 CONTINUE

730 FORMAT(5(2F8.0))
210 CONTINUE
5000 CONTINUE
WRITE(81) OW,NNN,(LUN(I),I=1,NU)
IF(NNN.EQ.0) GO TO 310
WRITE(81) (MFXY(I),I=1,NNN)
WRITE(81) (XD(I),I=1,NNN)
WRITE(81) (YD(I),I=1,NNN)
310 CONTINUE
IF(LINE.NE.1) WRITE(81) N1,EDP,ALP

READ(40,556) ICYCL
556 FORMAT(I5)
WRITE(81) ICYCL
REWIND 81
WRITE(50,503)
WRITE(50,4007)
4007 FORMAT(1H0,14H* NMAT TABLE *)
WRITE(50,510) (NMAT(I),I=1,NU)
510 FORMAT(1X,I6)
WRITE(50,4010)

4010 FORMAT(1H1////////20X,13(2H**) /20X,13(2H**) ////24X,18HPROGRAM (
*A ) END ,////20X,13(2H**)/20X,13(2H**) )
WRITE(50,4009)
4009 FORMAT(1H1)
GO TO 1000
C ERROR MESSAGE OUT
119 FORMAT(10X,5HERROR,I2)
999 WRITE(50,119) I11
1000 STOP
END

```

```

C BAXIFEM)
C / AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
C ** COMMON /A1/ **
C NUC = KAKU UNIT NO ELEMENT SU
C NMAT = KAKU UNIT NO JIYUDO SU
C NWAT = KYOKAI JYOKEN SYORI TABLE
C LUNI = KAKU UNIT NO KAZU SU
C ICAB = KAKU UNIT NO JIYUSETTEN SU
C ICBA = KAKU UNIT NO JIYUSETTEN SU + 1
C NAPAR = KYOKAI SYORI GO NO INDEX
C
C ITBL = CLASS BUNRUI TABLE
C JTBL = CLASS BUNRUI TABLE
C NEJO = ELEMENT NO I,J,K SETTEN BANGO
C ** COMMON /A11/ **
C X=R-ZAHYO
C Y=Z-ZAHYO
C WK = ELEMENT STIFFNESS MATRIX
C USM = KAKU UNIT TANI NO STIFFNESS MATRIX
C TSM = USM NO WORK AREA
C FOCE = KAKU UNIT TANI NO KAJUKOH
C
C WFOCE = FOCE NO WORK AREA
C ADEF = HENI AND HANRYOKU
C ** COMMON /A21/ **
C I10,I20,I30,I40,I50 = KOHZOHBTU MEISYO
C NPAGE = PAGE-SU
C ** COMMON /A3/ **
C EDP = SYUSOKUCHI
C WW1 = X-HOKOH NO ZEN KAJU
C WW2 = Y-HOKOH NO ZEN KAJU
C NP = SETTEN SU
C
C NE = ELEMENT SU
C NU = UNIT SU
C OW = JITYU NO UMU
C NOP = DANSEI PATTERN SU
C NN1 = NON LINEAR DATA SU
C ** COMMON /A4/ **
C ALP = NON LINEAR DATA
C KSW = STOP HANTEI
C LIKO = SYUSOKU CYCLE SU
C ** COMMON /A5/ **
C
C IISI = SUB.FOW NI OITE ERROR NO BAAI NO STOP HANTEI
C ** DRUM DIMENSION **
C DELD = ELEMENT DATA NO KAKUNO
C DAPAR = NAPAR NO NAIYO O KAKUNO
C DNUAT = NWAT NO NAIYO O KAKUNO
C USM2 = USM,FOCE NO NAIYO O KAKUNO
C DXY = SETTEN ZAHYO KAKUNOH
C DRDEF = ADEF NO NAIYOH KAKUNOH
C DDD = N-1 CYCLE NO NAIYOH KAKUNOH
C DNEP = NEP NO NAIYOH O KAKUNOH

```

- 10-

- 20-

- 30-

- 40-

```

C      ALOAD = SETTEN KAZU O KAKUNOH:
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUHI(15),ICAB(15),
*      ICBA(15),NAPAR(60),ITBL(15,4),JTRL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USH(60,60),FOCE(60),TSM(60,60)
*      WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
COMMON /A5/ IISWI
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN

COMMON /A7/ EDP2
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
*      ,DXY(600),URDEF(600),DDD(600),DNEP(3600),ALOAD(600)
*      ,DLOAD(900)
INTEGER OW,PS,PCYCL,OPT1
DOUBLE PRECISION I10,I20,I30,I40,I50
ICYCL=1
KSW=0
IISWI=0
NPAGE=0

WRITE (50,1000)
1000 FORMAT(1H1////////,30X,12(3H***),1H*/30X,1H*,35X,1H*/30X,37H* S
1TRFSS AND DEFORMATION ANALYSIS */30X,1H*,35X,1H*/30X,37H* BY
2 FINITE ELEMENT METHOD */30X,1H*,35X,1H*/30X,12(3H***),1H*)
CALL DREAD
IPHAN=PCYCL+ICYCL-1
10 CALL STIF
CALL FORM
IF(IISWI.EQ.1) STOP
CALL BACW

CALL REAC
CALL STRESS
ICYCL=ICYCL+1
IF(LINE.EQ.1) GO TO 20
IF(KSW.EQ.0) GO TO 10
20 WRITE(50,4010)
4010 FORMAT(1H1////////20X,13(2H**)/20X,13(2H**) ////24X,18HPROGRAM (
*B ) END ,////20X,13(2H**)/20X,13(2H**))
STOP
END

```

```

SURROUTINE DREAD
BAXIFEM2
C
C
AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),JTBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USH(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,120,130,140,150,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
- 10-

COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DOUBLE PRECISION I10,120,130,140,150
INTEGER OX,PS,PCYCL,OPT1
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* ,DLOAD(900)
DIMENSION NEP(300,12)
EQUIVALENCE (USM,NEP)
REWIND 81
- 20-

REWIND 82
READ (81) I10,120,130,140,150
READ(81) NP,NE,NU,NOP,ISWI,MCYCL,PCYCL,OPT1,LINE,EDP2
READ (81) (ISPU(I),I=1,NP)
READ (81) (X(I),I=1,NP)
READ (81) (Y(I),I=1,NP)
READ (81) (NUE(I),I=1,NU)
READ (81) NEP
DRUM WRITE DXY,X,Y
DRUM WRITE DNEP,NEP
- 30-

IF(ISWI.EQ.1) WRITE(50,1060)
1060 FORMAT(1H1)
IF(MCYCL.EQ.0) MCYCL=10
DO 10 I=1,NE
JT=8*I-7
READ (81) N1,N2,N3,N4,EE,V,GAMMA,ICON
DRUM WRITE DELD(JT),N1,N2,N3,N4,EE,V,GAMMA,ICON
IF(ISWI.NE.1) GO TO 10
WRITE(50,1031) N1,N2,N3,N4,EE,V,GAMMA,ICON
1031 FORMAT(1H0,4I5,3F10.3,15)
- 40-

10 CONTINUE
READ (81) ((JTBL(J,I),I=1,NU),J=1,120)
READ (81) ((ITBL(I,J),I=1,NU),J=1,4)
IF(ISWI.NE.1) GO TO 90
WRITE(50,1012) (NUE(I),I=1,NU)
WRITE(50,1040) ((JTBL(J,I),I=1,NU),J=1,120)
WRITE(50,1040) ((ITBL(I,J),I=1,NU),J=1,4)
1040 FORMAT(1H ,4I5)
90 CONTINUE
DO 20 I=1,NU

```

```

110=600*11-599
111=60*11-59
READ (81) ((NWAT(I,J),I=1,NP),J=1,2)
READ (81) (NAPAR(I),I=1,60)
DRUM WRITE DNWAT(110),NWAT
DRUM WRITE DAPAR(111),NAPAR
IF(ISW1.NE.1) GO TO 20
WRITE(50,1050) ((NWAT(I,J),I=1,NP),J=1,2)
WRITE(50,1012) (NAPAR(I),I=1,60)
20 CONTINUE

READ (81) (NMAT(I),I=1,NU)
IF(ISW1.NE.1) GO TO 130
WRITE(50,1012) (NMAT(I),I=1,NU)
130 CONTINUE
1050 FORMAT(1H0,2014)
READ (81) OW,NNN,(LUNI(I),I=1,NU)
IF(NNN.EQ.0) GO TO 40
READ (81) (NAPAR(I),I=1,NNN)
READ (81) (X(I),I=1,NNN)
READ (81) (Y(I),I=1,NNN)

DO 60 111=1,NNN
JT=3*111-2
DRUM WRITE DLOAD(JT),NAPAR(111),X(111),Y(111)
60 CONTINUE
40 CONTINUE
IF(LINE.NE.1) READ(81) NN1,EDP,ALP
READ(81) ICYCL
ICYCL=ICYCL+1
IF(ICYCL.EQ.1) GO TO 150
READ(81) (ADEF(I),I=1,600)

DRUM WRITE DDD ,ADEF
150 CONTINUE
NOP2=2*NOP
IF(LINE.EQ.1) GO TO 100
WRITE(50,3000)
GO TO 110
100 WRITE(50,3010)
110 CONTINUE
3000 FORMAT(1H0/38X,23H( NON LINEAR ANALYSIS )//56X,7H(KG,CM))
3010 FORMAT(1H0/38X,23H( LINEAR ANALYSIS )//56X,7H(KG,CM))

WRITE (50,1000)
1000 FORMAT(1H0/5X,16H** INPUT DATA **)
WRITE (50,1010)
1010 FORMAT(1H0,8X,24H** AXISYMMETRIC ANALYSIS)
WRITE (50,1020)
1020 FORMAT(1H0,8X,22H** THICKNESS= 360.0 DO)
IF (OW.EQ.1) WRITE(50,1030)
1030 FORMAT(1H0,8X,13H** OWN WEIGHT)
WRITE(50,1001) NP,NE,NU,NOP
1001 FORMAT(1H0//11X,19HNUMBER OF JOINT =,15//11X,19HNUMBER OF ELEMEN

```

```

      *T =,15//11X,19HNUMBER OF UNIT      =,15//11X,19HNUMBER OF PATTERN =:
      *15)
1012 FORMAT(1H0,2015)
      CALL PAGE
      IF(LINE.EQ.1) GO TO 120
      IF(NOP.GT.5) NOP2=10
      DO 50 I=1,NN1
      WRITE(50,2020) (ALP(I,J),ALP(I,J+1),J=1,NOP2,2)
2020 FORMAT(1H ,5(2F10,5,3X))
      50 CONTINUE
      IF(NOP.LE.5) GO TO 70
      NOP2=2*NOP
      WRITE(50,2030)
2030 FORMAT(1H0)
      DO 80 I=1,NN1
      WRITE(50,2020) (ALP(I,J),ALP(I,J+1),J=11,NOP2,2)
      80 CONTINUE
      70 CONTINUE
      120 CONTINUE
      IF(LINE.EQ.1) GO TO 121
      WRITE(50,2040) EDP2
2040 FORMAT(1H0,15HDAMPING FACTOR=,F15.7)
      WRITE(50,2041) EDP
2041 FORMAT(1H0,17HERROR LIMITATION=,E15.8)
      121 CONTINUE
      RETURN
      END

```

```

SUBROUTINE STIF
C
C BAXIFEM3
C AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NHAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE

COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNH
INTEGER OW,PS,PCYCL,OPT1
DOUBLE PRECISION I10,I20,I30,I40,I50
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* ,DLOAD(900)
DIMENSION DEF(300,2)
DIMENSION A(6,6),DAWK(6,6),ADAW(6,6)
EQUIVALENCE (ADEF,DEF)
DRUM READ DXY,X,Y

REWIND 82
REWIND 81
WW1=0.
WW2=0.
ICOU=0
KCOU=0
NEL=0
DO 180 I1=1,2
DO 180 I2=1,NP
DEF(I2,I1)=0.

180 CONTINUE
1040 FORMAT(1H0,2X,2HNE,5X,2HIE,2X,2HJE,2X,2HKE,3X,5HGAMMA,3X,8HYOUNG-M
* ,3X,10HPOISSONS-R,4X,8HYOUNG-P,/)
ICOU1=46
DO 10 K0=1,NU
IF(ICYCL,EQ.1.OR.OPT1,EQ.1) GO TO 301
GO TO 300
301 CONTINUE
CALL PAGE
WRITE(50,1050) K0

WRITE(50,1040)
1050 FORMAT(1H0,5H*UNIT,13,2H *)
300 CONTINUE
NEND=NUC(K0)
NN=NHAT(K0)/100
JEN=NHAT(K0)-NN*100
NEN=NY+JEN
DO 20 J1=1,NEN
FOCE(J1)=0.
WFOCE(J1)=0.

```

- 50-

```

20 CONTINUE
  IJ=K0*600-599
  DRUM READ DNWAT(IJ),NWAT
  DO 30 K1=1,NEN
  DO 30 K2=1,NEN
    USM(K1,K2)=0.
30 CONTINUE
  DO 40 K3=1,NEND
    NEL=NEL+1
    JT=8*NEL-7

```

- 60-

```

  DRUM READ DELD(JT),NOEL,NEJO,EE,V,GAMMA,ICON
  IF (ICYCL.E0.1.OR.OPT1.E0.1) GO TO 311
  GO TO 310
311 CONTINUE
  WRITE(30,1060)NOEL,NEJO,GAMMA,EE,V,ICON
1060 FORMAT(1H ,14,3X,314,F8.3,F10.1,F11.3,8X,12)
310 CONTINUE
  IJ=NEJO(1)
  JJ=NEJO(2)
  KK=NEJO(3)

```

- 70-

```

  XI=X(IJ)
  XJ=X(JJ)
  XK=X(KK)
  XM=XK
  YI=Y(IJ)
  YJ=Y(JJ)
  YK=Y(KK)
  YM=YK
  XM1=XK-XI
  YM1=YK-YI

```

- 80-

```

  XJ1=XJ-XI
  YJ1=YJ-YI
  C=EE*(1.-V)/(1.+V)/(1.0-2.0*V)
  C11=C
  C22=C
  C12=C*(V/(1.0-V))
  CX=C12
  C33=C*(1.0-2.0*V)/(2.0*(1.0-V))
  IF (XI.E0.XM) GO TO 61
  AIM=(XM*YI-XI*YM)/(XM-XI)

```

- 90-

```

  BIM=(YM-YI)/(XM-XI)
  GO TO 62
61 CONTINUE
  AIM=0.0
  BIM=0.0
62 CONTINUE
  IF (XJ.E0.XM) GO TO 63
  AMJ=(XJ*YM-XM*YJ)/(XJ-XM)
  BMJ=(YJ-YM)/(XJ-XM)
  GO TO 64

```

- 100-

```

63 CONTINUE
  AMJ=0.0
  BMJ=0.0
64 CONTINUE
  IF (X1.EQ.XJ) GO TO 65
  AJI=(X1*YJ-XJ*YI)/(X1-XJ)
  BJI=(YI-YJ)/(X1-XJ)
  GO TO 66
65 CONTINUE
  AJI=0.0
  BJI=0.0
66 CONTINUE
  T111=(XJ1+YH1-XH1+YJ1)*0.50
  T112=(X1+XJ+XH)*T111*(1.0/3.0)
  T113=(Y1+YJ+YH)*T111*(1.0/3.0)
  T114=3.0*T111/(X1+XH+XJ)
  T115=0.5*(AJI**2-AIM**2)*ALOG(X1)+
1    (AMJ**2-AJI**2)*ALOG(XJ)+
2    (AIM**2-AMJ**2)*ALOG(XH))
3*AIM*BIH*(XH-X1)+AMJ*BMJ*(XJ-XH)
4*AJI*BJI*(X1-XJ)+0.25*(BIH**2*(XH**2-XI**2)
5*BMJ**2*(XJ**2-XH**2)+BJI**2*(X1**2-XJ**2))
  T116=(1.0/3.0)*(AJI**3-AIH**3)*ALOG(X1)+
1  (AMJ**3-AJI**3)*ALOG(XJ)+(AIM**3-AMJ**3)*ALOG(XH))+
2  AIH**2*BIH*(XH-X1)+AMJ**2*BMJ*(XJ-XH)+AJI**2*BJI*
3  (X1-XJ)+0.5*(AIM*BIH**2*(XH**2-XI**2)+AMJ*BMJ**2*
4  (XJ**2-XH**2)+AJI*BJI**2*(X1**2-XJ**2))+
5  (1.0/9.0)*(BIH**3*(XH**3-XI**3)+BMJ**3*(XJ**3-XH**3)+
6  BJI**3*(X1**3-XJ**3))
  T117=(X1**2+XJ**2+XH**2+(X1+XJ+XH)**2)*T111/12.0
  T118=(X1*Y1+XJ*YJ+XH*YH+(X1+XH+XJ)*(Y1+YJ+YH))*T111/12.0
  WK(1,1)=C11*T114
  WK(1,2)=(CX+C11)*T111
  WK(1,3)=C11*T115
  WK(1,4)=0.
  WK(1,5)=0.
  WK(1,6)=C12*T111
  WK(2,2)=2.0*T112*(C11+CX)
  WK(2,3)=T13*(C11+CX)
  WK(2,4)=0.
  WK(2,5)=0.
  WK(2,6)=2.0*T112*C12
  WK(3,3)=C11*T116+C33*T112
  WK(3,4)=0.
  WK(3,5)=C33*T112
  WK(3,6)=C12*T13
  WK(4,4)=0.
  WK(4,5)=0.
  WK(4,6)=0.
  WK(5,5)=C33*T112

```

- 110-

- 120-

- 130-

- 140-

```

WK(5,6)=0.
WK(6,6)=C22*T112
DO 70 K4=1,5
KK4=K4+1
DO 70 K5=KK4,6
WK(K5,K4)=WK(K4,K5)
70 CONTINUE
DO 71 K4=1,6
DO 71 K5=1,6
A(K4,K5)=0.

- 150-

71 CONTINUE
A(1,1)=(XJ*YM-XM*YJ)/(2.0*T111)
A(2,1)=(YJ-YM)/(2.0*T111)
A(3,1)=(XM-XJ)/(2.0*T111)
A(1,3)=(XM*YI-XI*YM)/(2.0*T111)
A(2,3)=(YM-YI)/(2.0*T111)
A(3,3)=(XI-XM)/(2.0*T111)
A(1,5)=(XI*YJ-XJ*YI)/(2.0*T111)
A(2,5)=(YI-YJ)/(2.0*T111)
A(3,5)=(XJ-XI)/(2.0*T111)

- 160-

DO 72 K4=1,3
KK4=K4+3
DO 72 K5=1,5,2
KK5=K5+1
A(KK4,KK5)=A(K4,K5)
72 CONTINUE
C WRITE(50,4010) ((A(II,JJ),II=1,6),JJ=1,6)
4010 FORMAT(1H,6F12.5)
DO 73 K4=1,6
DO 73 K5=1,6

- 170-

TT=0.0
DO 74 K6=1,6
74 TT=TT+WK(K4,K6)*A(K6,K5)
73 DAWK(K4,K5)=TT
75 CONTINUE
DO 76 K4=1,6
DO 76 K5=1,6
TT=0.0
DO 77 K6=1,6
77 TT=TT+A(K6,K4)*DAWK(K6,K5)

- 180-

76 ADAW(K4,K5)=6.28318531*TT
78 CONTINUE
DO 82 K4=1,6
DO 82 K5=1,6
82 WK(K4,K5)=ADAW(K4,K5)
C WRITE(50,4010) ((WK(II,JJ),II=1,6),JJ=1,6)
DO 80 K6=1,3
N11=NEJO(K6)
KR=2*K6-2
DO 90 K7=1,3

- 190-

```

```

N33=NEJO(K7)
JR=2*K7-2
DO 100 K8=1,2
KK=NWAT(N11,K8)
KE=KR+K8
DO 110 K9=1,2
KL=NWAT(N33,K9)
NO=JR+K9
USM(KK,KL)=USM(KK,KL)+WK(KE,NO)
110 CONTINUE
- 200-

100 CONTINUE
90 CONTINUE
80 CONTINUE
IF(OW.EQ.0) GO TO 150
WE=-0.001*GAMMA*(XI+XJ+XM)*T111*0.1111111
DO 120 K10=1,3
N33=NEJO(K10)
IR=NWAT(N33,2)
FOCE(IR)=FOCE(IR)+WE
DEF(N33,2)=DEF(N33,2)+WE
- 210-

120 CONTINUE
150 CONTINUE
40 CONTINUE
KLU2=LUNI(K0)/100
KLU1=LUNI(K0)-KLU2*100
IF(KLU1.EQ.0) GO TO 130
DO 140 K11=1,KLU1
ICOU=ICOU+1
JT=3*ICOU-2
DRUM READ DLOAD(JT),LNO,XL,YL
- 220-

NP1=NWAT(LNO,1)
FOCE(NP1)=FOCE(NP1)+XL
NP1=NWAT(LNO,2)
FOCE(NP1)=FOCE(NP1)+YL
DEF(LNO,1)=DEF(LNO,1)+XL
DEF(LNO,2)=DEF(LNO,2)+YL
- 230-

140 CONTINUE
130 CONTINUE
IF(KLU2.EQ.0) GO TO 190
DO 200 K11=1,KLU2
ICOU=ICOU+1
JT=3*ICOU-2
DRUM READ DLOAD(JT),LNO,XL,YL
NP1=NWAT(LNO,1)
WFOCE(NP1)=WFOCE(NP1)+XL
NP1=NWAT(LNO,2)
WFOCE(NP1)=WFOCE(NP1)+YL
- 240-

200 CONTINUE
CALL INDI(NEN,NW)
190 CONTINUE

```

```

IF(ICYCL.EQ.1.OR.OPT1.EQ.1) GO TO 321
GO TO 320
321 CONTINUE
WRITE(50,1090) K0
1090 FORMAT(1H0//5X,10H** LOAD **5X,6HUNIT =,13//5X,2HNP,10X,5HR(KG),
*10X,5HZ(KG)/)
320 CONTINUE
DO 170 K11=1,NP
IF(NWAT(K11,1).EQ.0) GO TO 170
KRR=NWAT(K11,1)
KRL=NWAT(K11,2)
W1=FOCE(KRR)
W2=FOCE(KRL)
WW1=WW1+W1
WW2=WW2+W2
IF(ICYCL.EQ.1.OR.OPT1.EQ.1) GO TO 331
GO TO 330
331 CONTINUE
WRITE(50,1070) K11,W1,W2
1070 FORMAT(1H,3X,13,2F15.5)
330 CONTINUE
170 CONTINUE
WRITE(81) ((USM(J1,J2),J1=1,60),J2=1,60)
WRITE(81) (FOCE(J1),J1=1,60)
10 CONTINUE
IF(ICYCL.EQ.1.OR.OPT1.EQ.1) WRITE(50,1080) WW1,WW2
1080 FORMAT(1H0/7X,6F15.5)
DRUM WRITE ALOAD,ADEF
RETURN
END

```

```

C
C
SUBROUTINE MTWT
BAXIFEMA
AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUE(15),NHAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USH(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
- 10-

COMMON /A5/ IISWI
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* DXY(600),ORDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* DLOAD(900)
INTEGER OW,PS,PCYCL,OPT1
DIMENSION NEP(300,12)
EQUIVALENCE (USH,NEP)
DOUBLE PRECISION I10,I20,I30,I40,I50
- 20-

IPHAN=IPHAN+PCYCL
REWIND 81
WRITE(81) I10,I20,I30,I40,I50
WRITE(81) NP,NE,NU,NOP,ISWI,MCYCL,PCYCL,OPT1,LINE,EDP2
WRITE(81) (ISPU(I),I=1,NP)
DRUM READ DXY,X,Y
DRUM READ DNEP,NEP
WRITE(81) (X(I),I=1,NP)
WRITE(81) (Y(I),I=1,NP)
WRITE(81) (NUE(I),I=1,NU)
- 30-

WRITE(81) NEP
DO 10 I1=1,NE
JT=8*I1-7
DRUM READ DELD(JT),N1,N2,N3,N4,EE,V,GAMMA,ICON
WRITE(81) N1,N2,N3,N4,EE,V,GAMMA,ICON
10 CONTINUE
WRITE(81) ((JTBL(J,I),I=1,NU),J=1,120)
WRITE(81) ((ITBL(I,J),I=1,NU),J=1,4)
DO 20 I1=1,NU
I10=600*I1-599
- 40-

I11=60*I1-59
DRUM READ DNWAT(I10),NWAT
DRUM READ DAPAR(I11),NAPAR
WRITE(81) (NWAT(I,J),I=1,NP),J=1,2)
WRITE(81) (NAPAR(I),I=1,60)
20 CONTINUE
WRITE(81) (NHAT(I),I=1,NU)
WRITE(81) OW,NNN,(LUNI(I),I=1,NU)
IF(NNN,EG,0) GO TO 40
DO 30 I1=1,NNN

```

- 50-

```
JT=3*II-2
DRUM READ DLOAD(JT),NAPAR(II),X(II),Y(II)
30 CONTINUE
WRITE(81) (NAPAR(I),I=1,NNN)
WRITE(81) (X(I),I=1,NNN)
WRITE(81) (Y(I),I=1,NNN)
40 CONTINUE
IF(LINE.NE.1) WRITE(81) NN1,EDP,ALP
WRITE(81) ICYCL
DRUM READ DDD,ADEF
```

- 60-

```
WRITE(81) (ADEF(I),I=1,600)
REWIND 81
PAUSE 55555
RETURN
END
```

```

SUBROUTINE FORW
C BAXIFEM5
C AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3),
COMMON /A11/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE

COMMON /A5/ IISWI
DIMENSION NWOK(60)
DOUBLE PRECISION I10,I20,I30,I40,I50
INTEGER OW,PS,PCYCL,OPT1
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
DIMENSION DEF(300,2)
EQUIVALENCE (ADEF,DEF)
IF (ISWI.NE.1) GO TO 130
DO 8010 I11=1,NU

K0=600*I11-599
K1=60*I11-59
DRUM READ DNWAT(K0),NWAT
DRUM READ DAPAR(K1),NAPAR
WRITE(50,6010) ((NWAT(L1,L2),L2=1,2),L1=1,NP)
WRITE(50,6020) (NAPAR(LL1),LL1=1,60)
8010 CONTINUE
130 CONTINUE
REWIND 81
DO 120 I1=1,NU

ICAB(I1)=0
ICBA(I1)=0
120 CONTINUE
DO 9999 I1=1,NU
K0=600*I1-599
7010 FORMAT(1H0,3H50=.13)
DRUM READ DNWAT(K0),NWAT
IF (ISWI.NE.1) GO TO 140
WRITE(50,6010) ((NWAT(LL1,LL2),LL2=1,2),LL1=1,NP)
6010 FORMAT(1H ,2I5)

140 CONTINUE
READ(81) ((USM(J1,J2),J1=1,60),J2=1,60)
READ(81) (FOCE(J1),J1=1,60)
JZ=NMAT(I1)/100
IF (I1.EQ.1) GO TO 299
K11=60*I1-119
6020 FORMAT(1H ,20I5)
DRUM READ DAPAR(K11),NAPAR
IF (ISWI.NE.1) GO TO 150
WRITE(50,6020) (NAPAR(LL1),LL1=1,60)

```

```

150 CONTINUE
    KLL=ICBA(I1-1)
    KAI=NMAT(I1-1)/100
    IF(KAI.EQ.ICAB(I1-1)) GO TO 6134
    DO 2503 NA=KLL,KAI
    K93=NAPAR(NA)
    KJ=K93/10
    KC=K93-KJ*10
    IND=NWAT(KJ,KC)
    NWOK(NA)=IND

2503 CONTINUE
    DO 31 K1=KLL,KAI
    K71=NWOK(K1)
    DO 33 K4=KLL,KAI
    K6=NWOK(K4)
    USM(K71,K6)=USM(K71,K6)+TSM(K1,K4)
33 CONTINUE
    FOCE(K71)=FOCE(K71)+WFOCE(K1)
31 CONTINUE
6134 CONTINUE

299 CONTINUE
    IF(ITBL(I1,2).EQ.0) GO TO 63
    NK=30+ITBL(I1,2)
    NA=JTBL(NK,I1)
    DO 61 K1=1,2
    KP=3-K1
    NI=NWAT(NA,KP)
    IF(NI.GT.0.AND.NI.LE.J2) GO TO 62
61 CONTINUE
    WRITE(50,2310) I1,NI,J2

    IISW1=1
    GO TO 8000
63 CONTINUE
    NK=ITBL(I1,1)
    NA=JTBL(NK,I1)
    DO 71 K1=1,2
    KP=3-K1
    NI=NWAT(NA,KP)
    IF(NI.GT.0.AND.NI.LE.J2) GO TO 62
71 CONTINUE

    WRITE(50,2310) I1,NI,J2
    IISW1=1
    GO TO 8000
62 CONTINUE
    ICAR(I1)=NI
    IF(I1.EQ.NU) GO TO 2616
    IF(ITHL(I1,3).EQ.0) GO TO 73
    NA=JTBL(61,I1)
    DO 81 K1=1,2
    NI=NWAT(NA,K1)

```

- 50-

- 60-

- 70-

- 80-

- 90-

```

      IF(NI.GT.O.AND.NI.LE.JZ) GO TO 83
81  CONTINUE
    WRITE(50,2310) II,NI,JZ
    IISW1=1
    GO TO 8000
73  CONTINUE
    NA=JTBL(91,II)
    DO 84 K1=1,2
    NI=NWAT(NA,K1)
    IF(NI.GT.O.AND.NI.LE.JZ) GO TO 83

84  CONTINUE
    WRITE(50,2310) II,NI,JZ
    IISW1=1
    GO TO 8000
83  CONTINUE
    ICBA(II)=NI
2616 CONTINUE
    NEN=ICAB(II)
    IF(ICBA(II).EQ.1) GO TO 6771
C *** INVERSE MATRIX KA ***
    CALL AMINV(NEN)
    JIE=JZ
    JIA=ICBA(II)
    KIE=ICAB(II)
C *** KA=1*KAB CALCULATION ***
    IF(II.EQ.NU) GO TO 2201
    DO 96 KI=1,KIE
    DO 93 JI=JIA,JIE
    TT=0.
    DO 94 NI=1,KIE
    TT=TT+USM(KI,NI)*USM(NI,JI)
94  CONTINUE
    TSM(KI,JI)=TT
93  CONTINUE
96  CONTINUE
2201 CONTINUE
    DO 95 NA=1,KIE
    WFOCE(NA)=0.
    TT=0.
    DO 97 IA=1,KIE
    TT=TT+WFOCE(IA)*USM(II,IA)
97  CONTINUE
    WFOCE(II)=TT
95  CONTINUE
7020 FORMAT(1H0,3HK3=,15)
    WRITE(82) (WFOCE(I),I=1,60)
    WRITE(82) ((TSM(I1,I2),I1=1,60),I2=1,60)
    IF(II.EQ.NU) GO TO 9999
    GO TO 6772
6771 CONTINUE

```

- 100-

- 110-

- 120-

- 130-

- 140-

```

B1=0.
WRITE(A2) B1
6772 CONTINUE
DO 101 JJ=J1A,J1E
DO 101 KK=J1A,J1E
TSM(JJ,KK)=USM(JJ,KK)
101 CONTINUE
DO 105 JJ=J1A,J1E
DO 103 KK=J1A,J1E
TT=0.

DO 104 NI=1,K1E
TT=TT+USM(JJ,NI)*TSM(NI,KK)
104 CONTINUE
TSM(JJ,KK)=TSM(JJ,KK)-TT
103 CONTINUE
105 CONTINUE
DO 110 KK=J1A,J1E
WFOCE(KK)=FOCE(KK)
110 CONTINUE
DO 112 JT=J1A,J1E

TT=0.
DO 113 KT=1,K1E
TT=TT+USM(JT,KT)*WFOCE(KT)
113 CONTINUE
WFOCE(JT)=WFOCE(JT)-TT
112 CONTINUE
9999 CONTINUE
2310 FORMAT(1H1.5X,8HERROR 10,315)
8000 CONTINUE
RETURN

END

```

- 150-

- 160-

- 170-

- 180-

```

C      SUBROUTINE AMINV(NP3)
C      BAXIFEM6
C      AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ HUE(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
*   ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),AM (60,60),FOCE(60),TSM(60,60)
*   WFOCE(60),ADEF(600),ISPU(300)
DO 2000 K2000=1,NP3
W=AM(K2000,K2000)
WW=ABS(W)

```

```

IF(WW.LE.0.00001) GO TO 2000
2001 DO 1700 K1700=1,NP3
1700 AM(K2000,K1700)=AM(K2000,K1700)/W
DO 2200 K2200=1,NP3
IF(K2200-K2000)1800,2200,1800
1800 DO 2100 K2100=1,NP3
IF(K2100-K2000)2900,2100,2900
2900 AM(K2200,K2100)=AM(K2200,K2100)-AM(K2200,K2000)*AM(K2000,K2100)
2100 CONTINUE
2200 CONTINUE

```

```

AM(K2000,K2000)=1./W
DO 1930 K1930=1,NP3
IF(K1930-K2000)3000,1930,3000
3000 AM(K1930,K2000)=-AM(K1930,K2000)/W
1930 CONTINUE
2000 CONTINUE
RETURN
END

```

```

SUBROUTINE BACW
C
C BAXIFEM7
AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUE(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /A11/ X(300),Y(300),WK(6,6),USH(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISW1,LINE

INTEGER OW,PS
DOUBLE PRECISION I10,I20,I30,I40,I50
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DKY(600),DRDEF(600),DDO(600),DNEP(3600),ALOAD(600)
DIMENSION DEF(300,2),WFOCE1(60)
EQUIVALENCE (ADEF,DEF)
DO 10 I1=1,NU
I1=NU-I1+1
KIE=ICAB(I1)
JIA=ICBA(I1)

J2=NMAT(I1)/100
I2=NMAT(I1)-J2*100
JIE=J2
KZ=I2+J2
NTK=60*(I1-1)+1
DRUM READ DAPAR(NTK),NAPAR
IF(ISW1.NE.1) GO TO 130
WRITE(50,1040) (ICAB(L1),L1=1,15), (ICBA(L1),L1=1,15)
1040 FORMAT(1H ,10I5)
WRITE(50,1000) (NAPAR(L1),L1=1,60)

1000 FORMAT(1H ,20I5)
130 CONTINUE
IF(I1.NE.NU) GO TO 20
BACKSPACE 81
BACKSPACE 81
READ(81) ((TSM(J1,J2),J1=1,60),J2=1,60)
READ(81) (WFOCE(J1),J1=1,60)
BACKSPACE 82
BACKSPACE 82
READ(82) (WFOCE1(I1),I1=1,60)

READ(82) ((USH(I1,I2),I1=1,60),I2=1,60)
DO 100 K100=1,KIE
WFOCE(K100)=WFOCE1(K100)
100 CONTINUE
GO TO 30
20 CONTINUE
BACKSPACE 81
BACKSPACE 81
BACKSPACE 81
BACKSPACE 81

```

```

READ(81) ((TSM(J1,J2),J1=1,60),J2=1,60)
READ(81) (WFOCE(J1),J1=1,60)
BACKSPACE 82
BACKSPACE 82
BACKSPACE 82
BACKSPACE 82
READ(82) (WFOCE1(I1),I1=1,60)
READ(82) ((USM(I1,I2),I1=1,60),I2=1,60)
1020 FORMAT(1H,5F10.5)
DO 110 K110=1,K1E

DO 120 K120=J1A,J1E
TSM(K110,K120)=USM(K110,K120)
120 CONTINUE
WFOCE(K110)=WFOCE1(K110)
110 CONTINUE
DRUM READ DRDEF,ADEF
1030 FORMAT(1H,2F15.6)
DO 42 NA=1,K1E
TT=0.00
DO 43 KA=J1A,J1E

N3=NAPAR(KA)/10
NXYZ=NAPAR(KA)-N3*10
TT=TT+TSM(NA,KA)*DEF(N3,NXYZ)
43 CONTINUE
N2=NAPAR(NA)/10
NXYZ=NAPAR(NA)-N2*10
DEF(N2,NXYZ)=WFOCE(NA) -TT
42 CONTINUE
IF (IZ.EQ.0) GO TO 410
NNB=J1E+1

DO 73 NB=NNB,KZ
T1=0.00
T2=0.00
DO 72 NA=1,K1E
N4=NAPAR(NA)/10
IXY=NAPAR(NA)-N4*10
T1=T1+TSM(NB,NA)*DEF(N4,IXY)
72 CONTINUE
DO 74 NC=J1A,J1E
N6=NAPAR(NC)/10

KXY=NAPAR(NC)-N6*10
T2=T2+TSM(NB,NC)*DEF(N6,KXY)
74 CONTINUE
TT=T1+T2
N5=NAPAR(NB)/10
JXY=NAPAR(NB)-N5*10
DEF(N5,JXY)=DEF(N5,JXY)+TT
73 CONTINUE
410 CONTINUE
DRUM WRITE DRDEF,ADEF

```

41 CONTINUE
 GO TO 10
 30 CONTINUE
 DO 80 I=1,2
 DO 80 J=1,NP
 DEF(J,I)=0,
 80 CONTINUE
 DO 52 NA=1,KIE
 N4=NAPAR(NA)/10
 IXY=NAPAR(NA)-N4*10

- 100-

DEF(N4,IXY)=WFOCE(NA)
 52 CONTINUE
 IF(Z.EQ.U) GO TO 510
 NNB=JIE+1
 DO 63 NB=NNB,KZ
 TT=0.0
 DO 62 NA=1,KIE
 N4=NAPAR(NA)/10
 IXY=NAPAR(NA)-N4*10
 TT=TT+TSM(NB,NA)*DEF(N4,IXY)

- 110-

62 CONTINUE
 N5=NAPAR(NB)/10
 JXY=NAPAR(NB)-N5*10
 DEF(N5,JXY)=DEF(N5,JXY)+TT
 63 CONTINUE
 510 CONTINUE
 DRUM WRITE DRDEF, ADEF
 10 CONTINUE
 RETURN
 END

- 120-

- 130-

```

SUBROUTINE REAC
C   BAXIFEM8
C   AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
C   ... DISPLACEMENT PRINT
C   ... REACTION PRINT
C   ... CAL OF UMBALANCE CHECK
C   ... NUEL = MEMBERS (UNET)
C   ... NPIN = SUPPORT NO HOKO-SHIKI
COMMON /A1/ NUE(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
*   ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJD(3)
*   COMMON /A1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
*   WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
INTEGER OW,PS,OPT1
DIMENSION NPIN(300),RAT(2),DEF(300,2),DDEF(300,2)
DOUBLE PRECISION I10,I20,I30,I40,I50
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
*   ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
EQUIVALENCE (ADEF,DEF), (USM,DDEF)
WRAT1=0,
WRAT2=0,
DRUM READ ALOAD,DDEF
DO 32 K=1,NP
KA=ISPU(K)
N8=0
GO TO(33,45,46,54),KA
45 CONTINUE
C   X -ROLLER
N8=2
GO TO 33
46 CONTINUE
C   Y -ROLLER
N8=1
GO TO 33
54 CONTINUE
C   PINROLLER
N8=12
33 CONTINUE
NPIN(K)=N8
32 CONTINUE
DRUM READ DRDEF,ADEF
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 200
GO TO 210
200 CONTINUE
CALL PAGE
WRITE(50,1020)
1020 FORMAT(1H0,5X,14H** REACTION **//3X,2HNP,10X,5HX(KG),10X,5HY(KG)

```

```

*      /)
210 CONTINUE
DO 100 KE=1,NP
IF(NPIN(KE).EQ.0) GO TO 100
DO 110 I=1,2
RAT(I)=0.
110 CONTINUE
N55=NPIN(KE)/10
N22=NPIN(KE)-N55*10
RAT(N22)=DEF(KE,N22)-DDEF(KE,N22)

DEF(KE,N22)=0.
IF(N55.EQ.0) GO TO 120
RAT(N55)=DEF(KE,N55)-DDEF(KE,N55)
DEF(KE,N55)=0.
120 CONTINUE
WRAT1=WRAT1+RAT(1)
WRAT2=WRAT2+RAT(2)
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) WRITE(50,1030) KE,RAT(1),
* RAT(2)
1030 FORMAT(1H ,14,2F15.5)

100 CONTINUE
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 220
GO TO 230
220 CONTINUE
WRITE(50,1040) WRAT1,WRAT2
1040 FORMAT(1H0,5H1OTAL,F14.5,F15.5)
WRITE(50,1050) WW1,WW2
1050 FORMAT(1H0,4HLOAD,2F15.5)
CALL PAGE
DRUM READ DXY,X,Y

KK=46
WRITE(50,1000)
1000 FORMAT(1H0//5X,18H** DISPLACEMENT **//21X,10HCOORDINATE,25X,
* 12HDISPLACEMENT//4X,2HNP,13X,5HX(CM),10X,5HY(CM),8X,10HDISP-X(C
*M),5X,10HDISP-Y(CM),11X,7HX-D(CM),8X,7HY-D(CM)/)
DO 61 ITT=1,NP
WF1=X(ITT)+DEF(ITT,1)
WF2=Y(ITT)+DEF(ITT,2)
IF(ITT.NE.KK) GO TO 65
CALL PAGE

WRITE(50,1000)
KK=KK+45
65 CONTINUE
WRITE(50,1010) ITT,X(ITT),Y(ITT),DEF(ITT,1),DEF(ITT,2),WF1,WF2
1010 FORMAT(1H ,15,3(3X,2F15.5))
61 CONTINUE
230 CONTINUE
DRUM WRITE DRDEF,ADEF
10 CONTINUE
RETURN

END

```

```

SUBROUTINE STRESS
C
C BAXIFEM9
C AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A1/ NUC(15),NMAT(15),NWAT(300,2),LUNI(15),ICAB(15),
* ICBA(15),NAPAR(60),ITBL(15,4),JTBL(120,15),NEJO(3)
COMMON /AA1/ X(300),Y(300),WK(6,6),USM(60,60),FOCE(60),TSM(60,60)
* WFOCE(60),ADEF(600),ISPU(300)
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A3/ EDP,WW1,WW2,NP,NE,NU,PS,T,OW,NOP,NN1
COMMON /A4/ ALP(20,20),KSW,LIKO,ISWI,LINE
- 10-
COMMON /A6/ ICYCL,MCYCL,PCYCL,OPT1,IPHAN,NNN
COMMON /A7/ EDP2
INTEGER OW,PS,PCYCL,OPT1
DRUM DIMENSION DELD(4000),DAPAR(900),DNWAT(9000)
* ,DXY(600),DRDEF(600),DDD(600),DNEP(3600),ALOAD(600)
* ,DLOAD(900)
DIMENSION IWW(4),DEF(300,2),AW1(4),AW2(4),EW(7),NOEL1(7)
DIMENSION G1(500),G2(500),G3(500),G4(500),G5(500),G6(500),GT(500),
1 G8(500),DHAN(600),DWHAN(300,2),G22(500)
DOUBLE PRECISION I10,I20,I30,I40,I50
- 20-
DIMENSION DT(4,4),HIZ(4,6),GG3(300)
DIMENSION WT(6),SIGMA(4),NEP(300,12)
EQUIVALENCE (ADEF,DEF), (DHAN,DWHAN)
EQUIVALENCE (USM,G1), (USM(501),G2), (USM(1001),G3), (USM(1501),G4),
* (USM(2001),G5), (USM(2501),G6), (USM(3001),G7)
EQUIVALENCE (NWAT,G8), (TSM,NEP)
IF (ICYCL.EQ.MCYCL) KSW=1
DRUM READ DNEP,NEP
NEL=0
NEL1=0
- 30-
NEL2=0
LIKO=LIKO+1
IF (ICYCL.EQ.1) GO TO 201
DRUM READ DDD,DHAN
3020 FORMAT(1H0,30X,19H** YOUNG MODULUS **,/1X,5X,7(5H NE,9X,1HE))
AMWX1=0,
LL=0
CALL PAGE
WRITE(50,3000)
3000 FORMAT(1H0,30X,46H**DIFFERENCE OF DISPLACEMENT ( D(I)-D(I-1)) **)
- 40-
WRITE(50,3001)
3001 FORMAT(1H0,5X,4(5H NP,3X,7HDIFF(R),3X,7HDIFF(Z))/)
204 CONTINUE
DO 200 I1=1,4
KLL=I1-1
LL=LL+1
IF (LL.GT.NP) GO TO 202
IWW(I1)=LL
AW1(I1)=DEF(LL,1)-DWHAN(LL,1)
AW2(I1)=DEF(LL,2)-DWHAN(LL,2)

```

```

      AMMX1=AMMX1+AW1(I1)**2+AW2(I1)**2
200 CONTINUE
      WRITE(50,3010) (1W(KK),AW1(KK),AW2(KK),KK=1,4)
      IF(LL.NE.NP) GO TO 204
      GO TO 203
202 CONTINUE
      WRITE(50,3010) (1W(KK),AW1(KK),AW2(KK),KK=1,KLL)
203 CONTINUE
      AMMX1=AMMX1/FLOAT(2*NP)
      IF(AMMX1.LE.EUP) KSW=1
- 50-

3010 FORMAT(1H,5X,4(15,2F10.4))
      WRITE(50,3040) AMMX1
C3040 FORMAT(1H0//,10X,10HMAX.DIFF=,F15.8)
3040 FORMAT(1H0//,10X,17HERROR LIMITATION=,E15.8)
C      WRITE(50,6500)
C6500 FORMAT(1H1,3H NE,5X,14HDAMPIND-STRESS//)
201 DRUM WRITE DDD,ADEF
      DO 10 I1=1,NU
      DRUM READ DRDEF,ADEF
      IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 300
- 60-

      GO TO 310
300 CONTINUE
      CALL PAGE
      WRITE(50,5022)
5022 FORMAT(1H0,30X,23H( ELEMENT INFORMATION ))
      WRITE(50,2030) I1
2030 FORMAT(1H0,5X,9H** UNIT =,13,3H **)
      WRITE(50,2020)
2020 FORMAT(1H0,3H NE,7X,7HSIGMA-Z,7X,7HSIGMA-R,7X,7HSIGM-TH,7X,
*6HTAU-RZ,8X,7HTAU-MAX,7X,7HSIGMA-1,7X,7HSIGMA-2,8X,5HTHETA,
- 70-

*6X,7HK (R/Z)//
310 CONTINUE
      JEND=NUE(I1)
      DO 20 K1=1,JEND
      NEL=NEL+1
      JT=A*NEL-7
      DRUM READ DELD(JT),NOEL,NFJO,EE,V,GAMMA,ICON
      I1=NEJO(1)
      JJ=NEJO(2)
      KK=NEJO(3)
- 80-

      XI=X(I1)
      XJ=X(JJ)
      XK=X(KK)
      YI=Y(I1)
      YJ=Y(JJ)
      YK=Y(KK)
      XM=XK
      YM=YK
      XB=(XI+XJ+XM)/3.0
      YB=(YI+YJ+YM)/3.0
- 90-

```

```

XJI=XJ-XI
YJI=YJ-YI
YMI=YM-YI
XMI=XM-XI
T111=(XJI*YMI-XMI*YJI)*0.5
C=EF*(1,-V)/(1.+V)/(1,-2.*V)
C11=C
C22=C
C12=C*(V/(1.-V))
CX=C12

```

- 100-

```

C33=C*(1.-2.*V)/(2.*(1.-V))
DT(1,1)=C22
DT(1,2)=C12
DT(1,3)=C12
DT(1,4)=0.
DT(2,2)=C11
DT(2,3)=CX
DT(2,4)=0.
DT(3,3)=C11
DT(3,4)=0.

```

- 110-

```

DT(4,4)=C33
DT(2,1)=DT(1,2)
DT(3,1)=DT(1,3)
DT(4,1)=0.
DT(3,2)=CX
DT(4,2)=0.
DT(4,3)=0.
DO 40 K4=1,4
DO 40 K5=1,6
HIZ(K4,K5)=0.

```

- 120-

40 CONTINUE

```

HIZ(1,2)=(XM-XJ)/(2.0*T111)
HIZ(1,4)=(XI-XM)/(2.0*T111)
HIZ(1,6)=(XJ-XI)/(2.0*T111)
HIZ(2,1)=(YJ-YM)/(2.0*T111)
HIZ(2,3)=(YM-YI)/(2.0*T111)
HIZ(2,5)=(YI-YJ)/(2.0*T111)
HIZ(3,1)=(CXJ*YM-XM*YJ)/XB+(YJ-YM)*(XM-XJ)*(YB/XB)/(2.0*T111)
HIZ(3,3)=(CXH*YI-XI*YH)/XB+(YM-YI)*(XI-XM)*(YB/XB)/(2.0*T111)
HIZ(3,5)=(CXI*YJ-XJ*YI)/XB+(YI-YJ)*(XJ-XI)*(YB/XB)/(2.0*T111)

```

- 130-

```

HIZ(4,1)=(XM-XJ)/(2.0*T111)
HIZ(4,2)=(YJ-YM)/(2.0*T111)
HIZ(4,3)=(XI-XM)/(2.0*T111)
HIZ(4,4)=(YM-YI)/(2.0*T111)
HIZ(4,5)=(XJ-XI)/(2.0*T111)
HIZ(4,6)=(YI-YJ)/(2.0*T111)
DO 49 K4=1,4
DO 49 K5=1,6
TT=0.0
DO 41 KK4=1,4

```

- 140-

```

41 TT=TT+DT(K4,KK4)*H12(KK4,K5)
   WK(K4,K5)=TT
49 CONTINUE
   LLL=0
   DO 50 M11=1,3
     NO=NEJO(M11)
     DO 50 M12=1,2
       LLL=LLL+1
   WT(LLL)=DEF(NO,M11)
50 CONTINUE

   DO 60 JA=1,4
     TT=0.0
     DO 65 JA=1,6
       TT=TT+WK(JA,JA)*WT(JA)
65 CONTINUE
   SIGMA(JA)=TT
60 CONTINUE
   S1=(SIGMA(1)+SIGMA(2))*0.5
   S2=SORT(((SIGMA(2)-SIGMA(1))*0.5)**2+SIGMA(4)**2)
   SIG1=S1+S2
   SIG2=S1-S2
   THETA=28.647889*(ATAN(2.0*SIGMA(3)/(SIGMA(1)-SIGMA(2))))
   IF(SIGMA(1)-SIGMA(2).LE.0.0)THETA=THETA+90.
   AS2=ABS(SIGMA(2))
   GK1=9999.9999
   IF(AS2.GE.0.00001) GK1=SIGMA(1)/SIGMA(2)
   G1(NOEL)=SIGMA(2)
   G2(NOEL)=SIGMA(1)
   G3(NOEL)=SIGMA(4)
   G4(NOEL)=S2

   G63(NOEL)=SIGMA(3)
   G5(NOEL)=SIG1
   G6(NOEL)=SIG2
   G7(NOEL)=THETA
   G8(NOEL)=GK1
   IF((CYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1)
*WRITE(50,1000) NOEL,SIGMA(1),SIGMA(2),SIGMA(3),SIGMA(4),S2,SIG1,
*SIG2,THETA,GK1
1000 FORMAT(1H,13,7F14.6,F13.6,F13.6)
20 CONTINUE

10 CONTINUE
   KKK=51
   IF((CYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 320
   GO TO 330
320 CONTINUE
   CALL PAGE
   WRITE(50,5021)
5021 FORMAT(1H,30X,33H( MEAN VALUE AROUND NODAL POINT ))
   WRITE(50,2000)
2000 FORMAT(1H,3H NP,7X,7H5IGMA-2,7X,7H5IGMA-R,7X,7H5IGMA-TH,8X,6HTAU-R

```

```

*Z,7X,7HTAU-MAX,7X,7HSIGMA-1,7X,7HSIGMA-2,7X,5HTheta,5X,7HK (Z/R)/) - 200-
330 CONTINUE
DO 100 K100=1,NP
SIGMA(1)=0.
SIGMA(2)=0.
SIGMA(3)=0.
SIGMA(4)=0.
S2=0.
SIG1=0.
SIG2=0.

THETA=0.
GK1=0.
JJ=0
DO 110 K110=1,8
KK=NEP(K100,K110)
IF(KK.EQ.0) GO TO 120
JJ=JJ+1
SIGMA(1)=SIGMA(1)+G2(KK)
SIGMA(2)=SIGMA(2)+G1(KK)
SIGMA(3)=SIGMA(3)+G3(KK)

SIGMA(4)=SIGMA(4)+G3(KK)
S2=S2+G4(KK)
SIG1=SIG1+G5(KK)
SIG2=SIG2+G6(KK)
THETA=THETA+G7(KK)
GK1=GK1+G8(KK)
110 CONTINUE
120 CONTINUE
GK2=FLOAT(JJ)
GK2=1./GK2

SIGMA(1)=SIGMA(1)+GK2
SIGMA(2)=SIGMA(2)+GK2
SIGMA(3)=SIGMA(3)+GK2
SIGMA(4)=SIGMA(4)+GK2
S2=S2+GK2
SIG1=SIG1+GK2
SIG2=SIG2+GK2
THETA=THETA+GK2
GK1=GK1+GK2
IF(ICYCL.EQ.1.OR.KSW.EQ.1.OR.OPT1.EQ.1) GO TO 340

GO TO 240-
340 CONTINUE
IF(K100.NE.KKK) GO TO 130
CALL PAGE
WRITE(50,5021)
WRITE(50,2000)
KKK=KKK+50
130 CONTINUE
WRITE(50,2010) K100,SIGMA(1),SIGMA(2),SIGMA(3),SIGMA(4),S2,SIG1,
*SIG2,THETA,GK1

```

```

2010 FORMAT(1H ,I3,7F14.6,F13.6,F13.6)
350 CONTINUE
100 CONTINUE
IF(LINE.EQ.1) GO TO 220
5020 FORMAT(1H ,I5)
CALL PAGE
WRITE(50,3020)
DO 140 K140=1,NE
JT=8*K140-7
DRUM READ DELD(JT),NOEL,NEJO,EE,V,GAMMA,ICON
- 250-

NEL1=NEL1+1
NEL2=NEL2+1
NOEL1(NEL1)=NOEL
EW(NEL1)=EE
IF(NEL2.EQ.NE) GO TO 361
IF(NEL1.LT.7) GO TO 360
WRITE(50,3030) (NOEL1(I),EW(I),I=1,7)
NEL1=0
GO TO 360
3030 FORMAT(1H ,5X,7(I5,F10.2))
- 260-

361 CONTINUE
WRITE(50,3030) (NOEL1(I),EW(I),I=1,NEL1)
360 CONTINUE
KKK=ICON*2-1
NN2=NN1-1
IF(ICYCL.EQ.1) GO TO 480
G2(NOEL)=EDP2*(G2(NOEL)-G22(NOEL))+G22(NOEL)
480 CONTINUE
DO 150 K150=1,NN2
IF(ALP(K150,KKK).LE.G2(NOEL).AND.ALP(K150+1,KKK).GE.G2(NOEL)) GO T
- 280-

10 160
150 CONTINUE
160 R5=(ALP(K150+1,KKK+1)-ALP(K150,KKK+1))/(ALP(K150+1,KKK)
-ALP(K150,KKK))
1 R6=R5*G2(NOEL)+(R5*(-ALP(K150,KKK))+ALP(K150,KKK+1))
3010 FORMAT(1H ,5E13.4,I5)
R6=G2(NOEL)/R6
DRUM WRITE DELD(JT),NOEL,NEJO,EE,V,GAMMA,ICON
140 CONTINUE
220 CONTINUE
- 290-

DO 470 K470=1,NE
G22(K470)=G2(K470)
WRITE(50,6510) K470,G2(K470)
C6510 FORMAT(1H ,I3,F20.5)
470 CONTINUE
IF(KSW.EQ.0) GO TO 400
DRUM READ DDD,ADEF
CALL PAGE
DRUM READ DXY,X,Y
KK=46

```

- 300-

```

WRITE(50,6000)
6000 FORMAT(1H0//5X,18H** DISPLACEMENT **//21X,10HCOORDINATE+25X,
* 12H0ISPLACEMENT//4X,2HNP,13X,5HR(CM),10X,5HZ(CM),8X,10HDISP-R(C
*M),5X,10HDISP-Z(CM),11X,7HR-D(CM),8X,7HZ-D(CM)/)
DO 461 ITT=1,NP
WF1=X(ITT)+DEF(ITT,1)
WF2=Y(ITT)+DEF(ITT,2)
IF(ITT.NE.KK) GO TO 465
CALL PAGE
WRITE(50,6000)

```

- 310-

```

KK=KK+45
465 CONTINUE
WRITE(50,6010) ITT,X(ITT),Y(ITT),DEF(ITT,1),DEF(ITT,2),WF1,WF2
6010 FORMAT(1H ,15,3(3X,2F15.5))
461 CONTINUE
400 CONTINUE
IF(1CYCL.E0.1PHAN) CALL MTWT
RETURN
END

```

```

SUBROUTINE PAGE
C BAXIFEM10
C AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /A2/ I10,I20,I30,I40,I50,NPAGE
COMMON /A6/ICYCL
DOUBLE PRECISION I10,I20,I30,I40,I50
NPAGE=NPAGE+1
WRITE(50,1000) NPAGE,I10,I20,I30,I40,I50,ICYCL
1000 FORMAT(1H1/90X,5HPAGE=,I3,/5X,5A8,10X,6HCYCLE=,I3)
RETURN

```

END

- 10-

```

C      SUBROUTINE INDI(NEN,NW)
C      BAXIFEM11
C      AXISYMMETRIC ANALYSIS PROGRAM BY FINITE ELEMENT METHOD
COMMON /AA1/ X(300),Y(300),WK(6,6),
1      USM(60,60),FOCE(60),TSM(60,60),
2      WFOCE(60),ADEF(600),ISPU(300)
DO 10 I1=1,NEN
FOCE(I)=FOCE(I)+USM(I,I1)*WFOCE(I1)
10 CONTINUE
RETURN
END

```

- 10-

第3章 解 析 例

§3-1 線形計算

(1) 上記のプログラムのチェック計算として、円孔を持つ板の一軸状態での応力解析と半無限体地盤に帯状荷重が作用した場合の応力解析を行なった。

まず、円孔を持つ板の弾性理論解は、板が無限に大きくて板の周辺の影響が円孔まで及ばないと考えるとき、有限な大きさの有孔板を有限板内の小さな孔の問題として解かれ³⁾ ている。

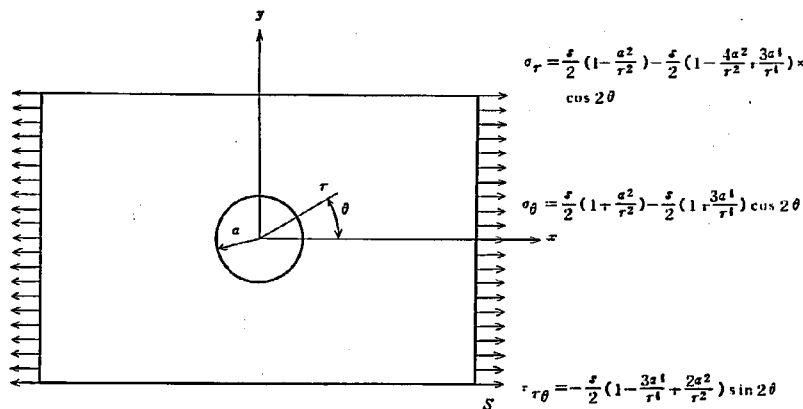


図 3-1

いま x 軸上の σ_y , および y 軸上の σ_x はそれぞれ $\theta = 0$, $\theta = \frac{\pi}{2}$ において

$$\sigma_y = (\sigma_\theta)_{\theta=0} = \frac{S}{2} \cdot \frac{a^2}{r^2} \left(1 - \frac{3a^2}{r^2}\right)$$

$$\sigma_x = (\sigma_\theta)_{\theta=\frac{\pi}{2}} = \frac{S}{2} \left(2 + \frac{a^2}{r^2} + \frac{3a^2}{r^4}\right)$$

有限要素法による解析断面は図3-2に示すもので、対称性を考えて、板の4分の1を要素分解している。その計算結果を図3-3に示す。

同図により理論解とよく一致していることがわかる。

次に半無限体地盤上に帯状荷重が作用した場合は平面ひずみ問題として解くことが出来、理論解⁴⁾ は鉛直方向の集中荷重に対する Boussinesq の解を荷重幅と無限長さにわたって積分することにより得られる(図3-4(a))。

これをプログラム化したものが図3-4(b)である。

この場合の解析断面は図3-5(a), (b)を用い、その各断面における垂直応力分布の結果を図3-6(a)に、最大主応力図を図3-6(b)に示す。

この図からわかるように理論解とよく一致している。なお6(a)図には、定方向要素断面による応力値を△印で示し、交番的断面による応力値を○印で示してある。この応力値は節点のまわりの要

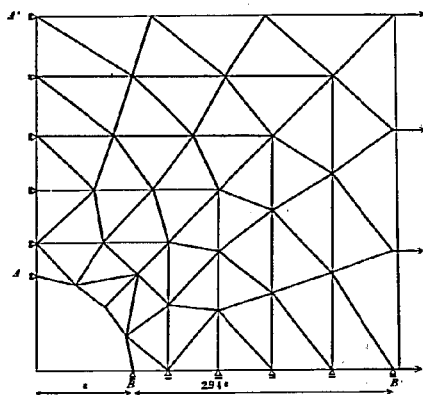


图 3-2 要素分割图

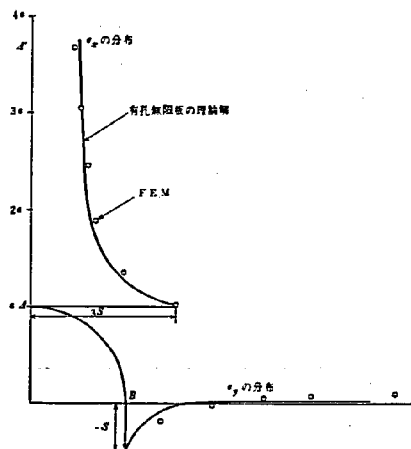


图 3-3 底力分布图

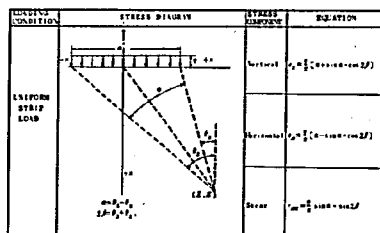


图 3-4(a)

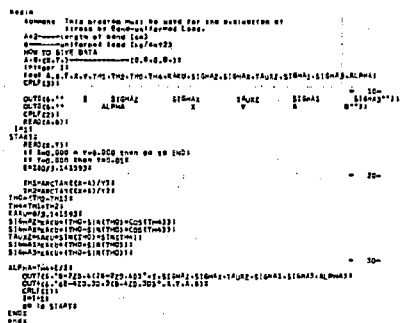


图 3-4(b)

素重心における応力を、平均したものであるため、要素形状による性質に影響されるものである。荷重直下では両者に差がみられるが、この差は荷重直下から離れるに従って、小さくなっている。

この計算は外力のみが作用した場合の結果である。なお物体力が作用する時も要素形状により値が異なり、方向性のないランダムな形状が望ましいといわれている。^{1), 5)}

(ロ) 軸対称問題の簡単な例題⁶⁾として無限量の厚肉円筒が内圧をうける場合の解析を行なった。

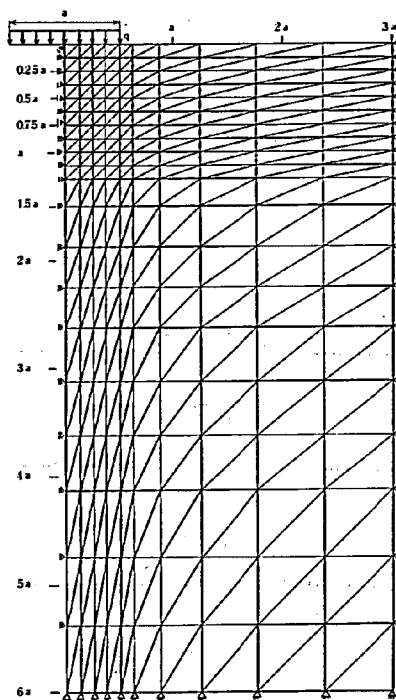


図 3-5(a) 定方向分割

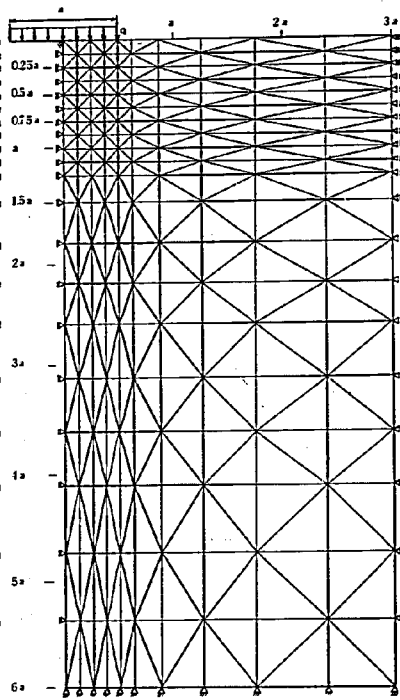


図 3-5(b) ランダム分割

この場合応力の一様性から長方形断面をとればよく、断面の上下端で外側に滑るローラーを仮定すればよい。

要素分割断面を図 3-7 に示し、結果を図 3-8、図 3-9 に示しておく。図から解るようによく一致した値が得られている。

なお、理論解の計算には図 3-10 のプログラムを用いている。

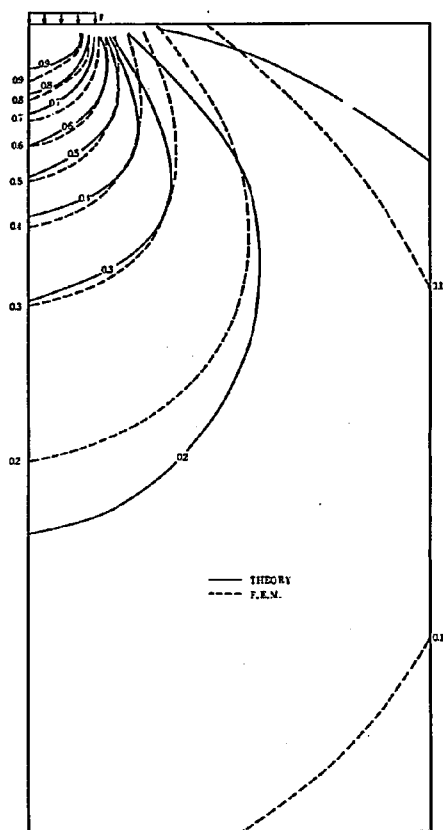


図 3-6(a) 最大主応力図 (単位 q)

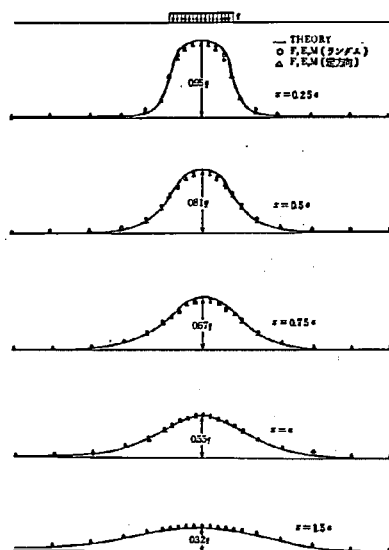


図 3-6(b) 各断面における垂直応力

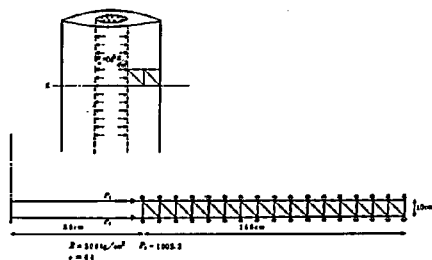


図 3-7 要素分解

(単位 10^{-3} cm)

mm	103	923	854	781	706	637	615	560	519	522	496	476	459	443	428	411	403
mm	103	923	854	782	704	633	613	570	541	521	497	476	458	442	427	411	402

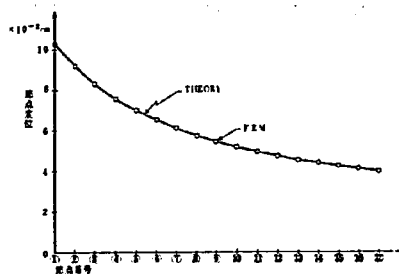


図 3-8 変位図

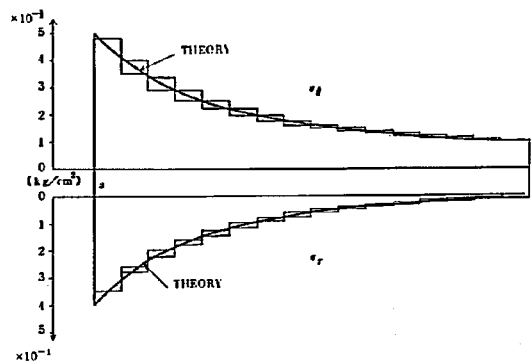


图 3-9 应力图

```

begin
  Comment This program must be used for evaluation
    of stress distribution symmetrical about an axis!
  Integer N; READ(N);
  begin
    Integer I;
    real A,AA,M,BB,PP,PQ,PP1,PI,RR;
    real MU,EE,M,EI;
    error R(I);E(I);N;E(I);N;U(I);N;SIG(I);N;S(I);N;
    for I=1 step 1 until N do
      R(I)=SIG(I);S(I)=0.0;
      CRLF(3);
      SPACE(10);
      OUT(6,"STRESS DISTRIBUTION SYMMETRICAL ABOUT AN AXIS");
      CRLF(2);
      READ(AA,BB,PP,PQ,PP1,MU,EE);
      A=BB/PQ;M=M*E*0.01;
      A=AA;
      B=BB;
      PP=PP1;
      PI=PP1; M=M*EI; EE=EE;
      IF PI=0 AND PQ=0 then go to ENU else go to S51;
      S51: IF PI=0 AND PQ=0 then go to S1 else go to S2;
      S1: SPACE(3); OUT(6,"STRESS DISTRIBUTION BY EXTERNAL AND INTERNAL PRESSURE");
      CRLF(2); go to JJ1;
      S2: IF PI=0 AND PQ=0 then go to S3 else go to S4;
      S3: SPACE(3); OUT(6,"STRESS DISTRIBUTION BY EXTERNAL PRESSURE ONLY");
      CRLF(2); go to JJ1;
      S4: SPACE(3); OUT(6,"STRESS DISTRIBUTION BY INTERNAL PRESSURE ONLY");
      CRLF(2); go to JJ1;
      JJ1: OUT(6,"YOUNG MODULUS"); OUT(6,"(3E-22D.6D)*E"); CRLF(1);
      OUT(6,"POISSON RATIO"); OUT(6,"(3E-22D.6D)*M"); CRLF(1);
      OUT(6,"I SIGMA=R SIGMA=THETA HENIU HIZUMI-THETA HIZUMI-RAD(U)");
      OUT(6,"A B PQ PI POINT"); CRLF(1);
      for I=1 step 1 until N do
        begin
          READ(RR);
          R(I)=RR;
          SIG(I)=A12*B12*(PQ-PI)/(B12-A12)*R(I);
          S(I)=A12*B12*(PQ-PI)/(B12-A12)*R(I);
          U(I)=1.0*(P1*A12-PQ*B12)/(C1-2*M)*R(I)/(R12-A12);
          E(I)=U(I)/R(I);
          ER(I)=(1+M)*((P1*A12-PQ*B12)/(C1-2*M)/(B12-A12)
            +A12*B12*(PQ-PI)/(B12-A12)*R(I)/E(I);
          OUT(6,"2D");
          OUT(6,"(3E-22D.6D)*SIG(I);U(I);E(I);ER(I);
          OUT(6,"(3E-22D.2D)*A,B,PQ,PI,R(I)"); CRLF(1);
        end; go to END2;
      END: OUT(6,"INPUT MISTAKE");
      END2:
    end;
  end;

```

图 3-10

§ 3-2 非線形計算

非線形解析例として盛土竣工後の圧縮沈下量の推定を試みる。

盛土の沈下量は施工中に生ずる沈下量と竣工後に生ずる沈下量とが考えられる。これらの沈下量は前者がその大部分をしめるものである。しかしながら盛土完成後に生ずる不等沈下など道路盛土の維持管理に問題となるのは後者の圧縮沈下量である。(図3-11参照)。盛土のこの沈下量は施工時の締固め機械重量と締固め回数に大きく支配されるものと考えられ、また数多くの粘性土(関東ローム)の室内圧縮実験ならびに実車実験よりこの沈下量を支配する、粘性土の圧縮特性として圧縮降伏応力値 P_0 なるものを得ている。^{7) 8)}

ところでB試験盛土と同じ試料による室内圧縮試験よりこの土に対する圧縮特性を得ているので、この特性値を用いてこの試験盛土における竣工後の圧縮沈下量を計算する。

いま図3-12に示した特性を用いることにする。沈下量を推定するにあたって盛土の施工時に転圧層に与えられた圧縮降伏応力値 P_0 を知らねばならない。試験盛土の施工時にはこの P_0 値は測定されていないが、締固め機械重量や転圧回数と P_0 値との関係が得られている(図3-13および図3-14)ので、これから P_0 値を推定した。ところで盛土は必ずしも均一に締固められていないので P_0 値も当然ばらついている。そこでこの P_0 値のばらつきの程度を決めるため、建設省の道路盛土工事現場における転圧実績より図3-15に示す盛土面に対する転圧回数の実態調査をもとにしてB盛土(転圧機械ブルドーザ10t級とタイヤローラー7t級、平均転圧回数4回)に対して P_0 値を表3-1のように割付けた。これをもとに図3-12の関係から観測期間の $t = 1$ 日と、 $t = 2$ 1ヶ月に対する $\sigma - \epsilon - t$ 曲線は図3-16のようになる。

これらの関係から上記の計算を行ない沈下板③~④間の沈下量として0.72 cmを得た。これは実測値の0.7 cmにきわめてよく一致している⁹⁾(図3-17、表3-2参照)

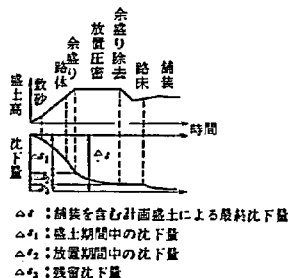


図3-11

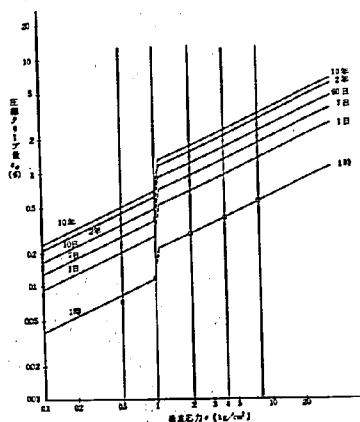


図3-12 $\sigma - \epsilon - t$ 曲線

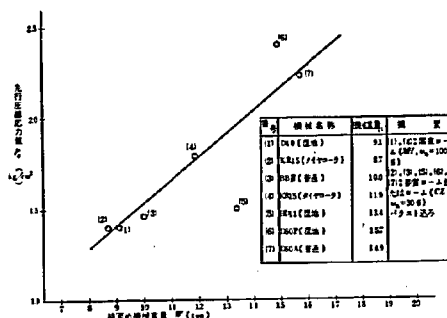


図 3-13 締固め機械重量と圧縮降伏応力値との関係

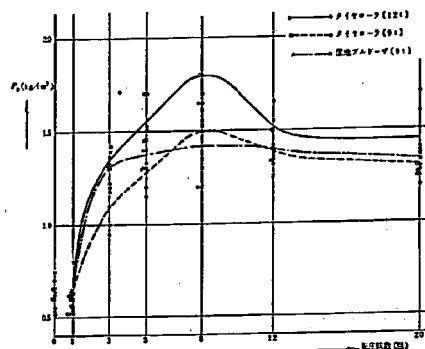


図 3-14 圧縮降伏応力値 (P_0) と転圧回数との関係

表 3-1 圧縮降伏応力 P_0 の割付け表

B 盛土	転 圧 回 数	2	3	4	5	6	7
	P_0	1.1	1.3	1.4	1.4	1.4	1.4
FEM要素 (300要素) への割付け数		16	104	35	43	17	85
FEM計算用パターン		①	②	③	④	⑤	⑥

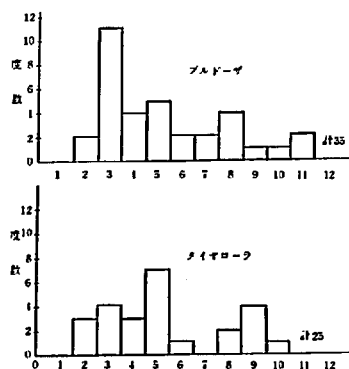


図 3-15 締固め回数頻度実績

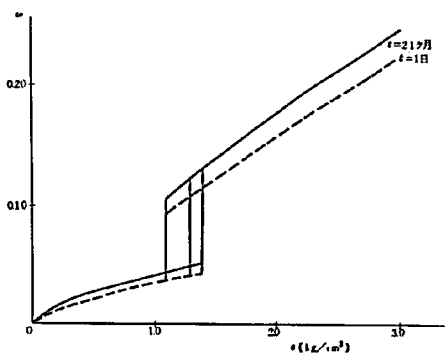
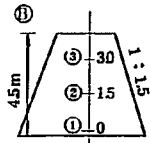


図 3-16 B 盛土の σ - ϵ - l 曲線

表3-2 B盛土の圧縮沈下量の要約

2ヶ月観測	層	(U) 施工中	(F) 竣工後	全沈 下量	F/U
	③ ①	1.3 (0.4)	0.7 (0.24)	2.0 (0.7)	0.6
	③ ②	0.7 (0.5)	0.15 (0.1)	0.9 (0.6)	0.2
	② ①	1.3 (0.7)	0.6 (0.4)	1.9 (1.3)	0.4

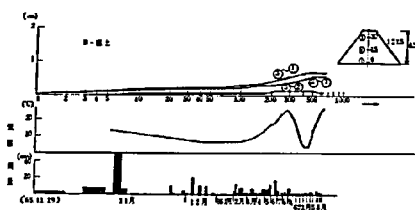


図3-17 盛土の竣工後の
圧縮沈下量

あ と が き

一般に固体力学における非線形問題には歪成分の有限性と応力-ひずみの非線形性によって次のような4つの種類の問題が考えられる。

(1) 微小変形問題

線形の応力ひずみ関係が成立し、ひずみが微小なる場合

(2) 材料非線形問題

応力ひずみ関係は非線形となるがひずみ成分は微小なる場合

(3) 有限変形問題

応力ひずみ関係は線形であるが、ひずみ成分が有限である場合

(4) 高次非線形問題

応力ひずみ関係も非線形であり、ひずみ成分も有限となる一般的非線形

これらの問題の解析方法として(1)反復法 (Iteration method), (2)荷重漸増法 (Load incremental method) の2種類考えられる。前者の方法は有限変形問題には適用しやすく、ある与えられた荷重に対する最終変形状態を求めることができる。が、しかし後者の方法はその荷重に対する初期から最終状態における各段階での変形状態を求めることができる。

この報文のプログラム(その1)は上記の(2)の分類に属するものといえよう。そしてプログラム(その2)は大変位と材料非線形性を同時に考慮することができ、上記の分類(4)に対する一つのアプローチといえよう。

参 考 文 献

- 1) 施工研究室「有限要素法による盛土等の内部応力変形解析」土木研究所資料/6 546
- 2) 宮本博「機械工学における有限要素法の応用」機械学会 1970 年 4 月講習会テキスト
- 3) TIMOSHENKO AND GOODIER 「THEORY OF ELASTICITY」
McGraw-Hill 1951
- 4) 土質工学ハンドブック 第 5 章
- 5) 川本眺万「有限要素法と基礎設計」
- 6) CLOUGH AND RASHID 「FINITE ELEMENT ANALYSIS OF
AXISYMMETRIC SOLID」 Proceedings OF ASCE 1965 EMI
- 7) 伊勢田哲也他 1 人「締固めた土の圧縮性状に関する研究」土木学会論文報告集/6 188
昭和 46 年 4 月
- 8) 施工研究室「土の締固めの一表示法に関する試験」土木研究所資料/6 83
- 9) 施工研究室「関東ロームを材料とする試験盛土」土木技術資料昭和 44 年 8 月
- 10) 川井忠彦「有限要素法研究の最近の動向」生産研究 Vol. 22, /6 1
- 11) Chiyayath V. Girijavallabhan and Lymon C. Reese
「FINITE ELEMENT METHOD FOR PROBLEMS IN SOIL
MECHANICS」 Proceedings OF ASCE 1968 SM2
- 12) 日本鋼構造協会「コンピューター使用によるマトリックス構造解析」
- 13) O. C. ZIENKIEWICZ 「THE FINITE ELEMENT METHOD IN
STRUCTURAL AND CONTINUUM MECHANICS」 McGraw-Hill
Publishing Company Limited 1967
- 14) O. C. ZIENKIEWICZ AND G. S. HOLISTER 「STRESS ANALYSIS」
John Wiley & Sons Ltd. 1964
- 15) SOKOLNIKOFF 「MATHEMATICAL THEORY OF ELASTICITY」
McGraw-Hill Book Company, Inc. 1956

付 録

有限要素法による軸対称問題解析用プログラム (ALGOL用) AXIFEM
境界条件としては強制変位と荷重条件が考慮されている。FACOM 270-30 (65kw, 16ビット) を用いると、とりうる節点数、要素数はそれぞれ42個, 60個程度である。

入力データ作成順序	施工研究室AXIFEM用
-----------	--------------

A 1 0, A 2 0, A 3 0, A 4 0, A 5 0, NP, NE, NEP, NFPXY, FPNXY[1.....NFPXY], (GAMMA[1.....NEP], EE[1.....NEP], V[1.....NEP]) (P, N), ... R[1.....NP], Z[1.....NP], (IE[1.....NE], JE[1.....NE], ME[1.....NE]) DDD, P, DISP[1.....NP2], LLL, { OW, (P, LOAD[1.....NP2]) OW, (P, LOAD[1.....NP2]) }	計算名 (5×8文字) 節点数, 要素数等 固定節点番号 弾性パターン情報 (ΣP=NE) R 座 標 Z 座 標 要素番号と節点番号の関係 強制変位 ΣP=NP2 荷重条件 LLL: くりかえし条件数 (2P=NP2)
--	--

プログラム

(AXIFEM)

```

begin
  comment AXIFEM TANAKA
    This program is the axisymmetric F.E.M.
    This is a program for stress and deformation analysis using Finite Element Method
    with vertical and horizontal roller ( triangular element )
  HOW TO GIVE DATA
  A10,A20,A30,A40,A50: Name of calculation (double real)
  NP: number of nodal points (integer)
  NE: number of elements (integer)
  NEP: number of elastic patterns (integer)

  NFPXY: number of fixed points considering R and Z direction (integer)
  FPNXY[1]: fixed points number considering R and Z direction (integer array)
  GAMMA[1]: bulk density of each element (g/cm3) (array)
  EE[1]: youngs modulus of each element (kg/cm2) (array)
  VEE[1]: Poissons ratio (array)
  P: number of seriesed same data (integer)
  N: elastic pattern number (integer)
  X[1]: r-coordinate of nodal points (cm) (array)
  Y[1]: z-coordinate of nodal points (cm) (array)
  IE[1]: nodal point number of I corner in each element (integer array)

  JEE[1]: nodal point number of J corner in each element (integer array)
  MEE[1]: nodal point number of M corner in each element (integer array)
  DDD: number of repeat for initial displacements data (integer)
  DISP[1]: initial displacement of each nodal point in R and Z direction (cm) (array)
  LLL: number of repeat for loads data (integer)
  OW: when you think of own weight, OW=1
      when you do not think of it, OW=0
  LOAD[1]: loads for each nodal points in R and Z direction (kg) (array)
  ORDER TO GIVE DATA
  A10, A20, A30, A40, A50, NP, NE, NEP,
  NFPXY, FPNXY[1...NFPXY],
  ( GAMMA[1...NEP], EE[1...NEP], VEE[1...NEP] ) ... ( P, N ) ... (P-NE),
  X[1...NP], Y[1...NP], ( IE[1...NE], JEE[1...NE], MEE[1...NE] ) ...
  DDD, ( P-DISP[1...NP2] ) ... (P-NP2),
  LLL, OW, C F, LOAD[1...NP2] ) ... (P-NP2),
  $
  double real A10,A20,A30,A40,A50:
  real
    X1,X2,Y1,Y2,YM,X1J,X2J,X1M,X2M,C1,C2,F1,F2,
    A1,B1,A1M,B1M,A1J,B1J,I1,I2,I3,I4,I5,I6,I7,I8,
    THET,SS,XP,XB,YJ1,XJ1,C33,CX,C22,C12,C11,C
    $
  integer
    SPB,S1,S2,SIG1,SIG2,THETA,T,GK1,GK2,GR3,GK4,C0:
    NP,NE,NMP,PS,OW,NP2,NP4,I,J,L,N,MS,I1,JJ,MM,KK,A1,A2,B1,B2,
    NMP2,P1,NK,P,0,1,I2,AA1,AA2,BB1,BB2,NMPXY,NMPXY2,NFPXY,NEP,
    DP,DDD,LLL,LLL,PPP:
  $
  START:
  READA(A10,A20,A30,A40,A50):
  READ(NP,NE,NEP,NFPXY):
  NP2=2*NP: NMPXY=NP2-NFPXY: NMPXY2=2*NMPXY:
  begin
  array
    EF,GAMMA,V,VEE[1...NE],X,Y[1...NP],LOAD,DISP,DISPA,RF[1...NP2],

```

```

CC, A, K11:ME, 1:6, 1:6J, DE1:4, 1:4J, H12UM11:4, 1:6J, GG31:NEJ,
K11:NE, 1:6, 1:6J, MAT11:NP2, 1:NP2J, MATME11:NMPXY, 1:NMPXY2J, MAC11:4, 1:6J,
EP1:EP2, EP31:NEP, UV11:6J, SIGMA11:4J,
G1, G2, G3, G4, G5, G6, G7, G81:NEJ,
Integer array JE, JF, ME11:NEJ, ENP, CNP11:NP, 1:8J, MPNXY11:NMPXYJ, FPNXY11:NFPXYJ,
HEADARPA1(FPNXY, 1, NFPXYJ),
if NFPXY=NP2 v NMPXY=NP2 then go to L500;
for I=1 step 1 until NMPXY do
begin
    if FPNXY[I]≠1 then MPNXY[I]=1 else go to L100;
end;
L100:
N=J=1;
for I=2 step 1 until NFPXY do
begin
    I1=FPNXY[I]-FPNXY[I-1];
    if I1≠1 then go to L200 else go to L300;
L200: MM=0;
    for JJ=N step 1 until (N+I-2) do
    begin
        MM=MM+1;
        MPNXY[J+JJ]=FPNXY[I-1]+MM;
    end;
    N=JJ;
L300:
end;
L400:
MM=0;
for JJ=N step 1 until NMPXY do
begin
    MM=MM+1;
    MPNXY[J+JJ]=FPNXY[NFPXY]+MM;
end;
L500:
for I=1 step 1 until NEP do
begin
    PEAD(GK1, GK2, GK3);
    EP1[I]=GK1, EP2[I]=GK2, EP3[I]=GK3;
end;
P=0;
for I=1 step 1 until NE do
begin
    READ(P, N);
    for J=(1+N) step 1 until (P+Q) do
    begin
        GAMMA[J]=EP1[N];
        EF[J]=EP2[N];
        VF[J]=EP3[N];
    end;
    if (P+Q)=NE then go to L600;
    Q=P+Q;
end;

```

- 60-

- 70-

- 80-

- 90-

- 100-

```

L600:
HEADARRAY(V+1,NP);
HEADARRAY(V+1,NP);
for I=1 step 1 until NE do
begin READ(A1,A2,B1);
end;
TEC[I]=A1; JET[I]=A2; MEC[I]=B1;
end;
for I=1 step 1 until NP do
for J=1 step 1 until B do
begin ELPF[I,J]=CNP[I,J]; end;

for I=1 step 1 until NP do
begin A1=A2=B1=0;
for J=1 step 1 until NE do
begin
if JET[J]=I then begin A1=A1+1; ENP[I,A1]=J; CNP[I,A1]=1; end;
if JET[J]=I then begin A1=A1+1; ENP[I,A1]=J; CNP[I,A1]=2; end;
if MEC[J]=I then begin A1=A1+1; ENP[I,A1]=J; CNP[I,A1]=3; end;
end;
end;
HEAD(DDN); PPP=0;

for DD=1 step 1 until DDD do
begin
PPP=0;
for I=1 step 1 until NP2 do
begin
HEAD(P,GK1);
for J=1 to step 1 until (P+q) do
DISPALJ=MGK1;
if (P+q)=NP2 then go to L700;
GPP=P+q;
end;
L700:
READ(LLL);
for LL=1 step 1 until LLL do
begin
for I=1 step 1 until NP2 do
DISP[I]=0;SPAL[I];
P=0;
READ(OW);
for I=1 step 1 until NP2 do
begin
READ(P,GR2);
for J=1 to step 1 until (P+q) do
LOADLJ=GR2;
if (P+q)=NP2 then go to L800;
GPP=P+q;
end;
L800:
PPP=PPP+1; if PPP=1 then go to L900;
for I=1 step 1 until NE do

```

- 110-

- 120-

- 130-

- 140-

- 150-


```

+AMT2*B1M*(XM-XI)+AMJT2*BMJ*(XJ-XM)+AJI+2*BJI*(XI-XJ)
+ (0.5)*(AI+BMIT2*(XMT2-XIT2)+AMJ+BMJT2*(XJT2-XMT2)+AJI+BJIT2*(XIT2-XJT2))
+ (1/9)*(BIMT3*(XMT3-XIT3)+BMJT3*(XJT3-XMT3)+BJIT3*(XIT3-XJT3))
17=(XI+T2-XJT2-XMT2*(XI+XM+XJ)T2)*111/121
18=(XI+YI+XJ+YJ+XM+YM*(XI+XM+XJ)*CI+YJ+YH)*111/121
K1I[.1.17]=C11+141
K1I[.2.17]=(CX+C11)*111
K1I[.3.17]=C11+151
K1I[.4.17]*K1I[.5.17]=0.01
K1I[.4.17]*C12=111
- 210-
K1I[.2.27]=2*(C11+CX)*1121
K1I[.3.27]=(C11+CX)*131
K1I[.4.27]=0.01
K1I[.4.27]=0.01
K1I[.4.27]=2+C12=1121
K1I[.5.37]=C11+16+C33=1121
K1I[.6.37]=0.01
K1I[.5.37]*C33=1121
K1I[.4.37]*C12=131
K1I[.5.37]*C33=1121
- 220-
K1I[.4.67]*C22=1121
K1I[.1.27]*K1I[.2.17]
K1I[.1.37]*K1I[.3.17]
K1I[.1.47]*K1I[.4.17]
K1I[.1.67]*K1I[.6.17]
K1I[.1.57]*K1I[.5.17]
K1I[.2.37]*K1I[.3.27]
K1I[.2.47]*K1I[.4.27]
K1I[.2.57]*K1I[.5.27]
K1I[.2.67]*K1I[.6.27]
- 230-
K1I[.4.47]*K1I[.4.37]
K1I[.5.57]*K1I[.5.37]
K1I[.4.67]*K1I[.6.37]
K1I[.4.47]*K1I[.4.57]*K1I[.4.67]*K1I[.5.47]*K1I[.5.67]*K1I[.6.57]=0.01
ACI[.1.17]=(XJ+YM-XM+YJ)/(2.0*111)
ACI[.2.17]=(YJ-YM)/(2.0*111)
ACI[.3.17]=(XM-XJ)/(2.0*111)
ACI[.4.17]=AI+.5.17=AI[.6.17]*ACI[.1.27]=AI[.2.27]=AI[.3.27]=0.01
ACI[.4.27]=ACI[.1.17]
ACI[.5.27]=ACI[.2.17]
- 240-
ACI[.6.27]=AI[.3.17]
ACI[.1.37]=(XM+YI-XI+YH)/(2.0*111)
ACI[.2.37]=(YM-YI)/(2.0*111)
ACI[.3.37]=(XI-YM)/(2.0*111)
ACI[.4.37]=ACI[.5.37]*ACI[.6.37]*ACI[.1.47]=ACI[.2.47]=ACI[.3.47]=0.01
ACI[.4.47]=ACI[.1.37]
ACI[.5.47]=ACI[.2.37]
ACI[.6.47]=ACI[.3.37]
ACI[.1.57]=(XI+YJ-XJ+YI)/(2.0*111)
ACI[.2.57]=(YI-YJ)/(2.0*111)
- 250-

```



```

for I=1 step 1 until NP do
begin
  I1=2-I-1; I2=2+I-1
  OUT1(6, '2D, 3R2D, 82D, 2B-62D, 2D, -62D, 2D', I, I1, I2, X(I), Y(I));
  OUT4(6, 'R, 2B-62D, 2D, 3B, 2(R-32D, 3D)', LOAD(I), LOAD(I2), DISPE(I), DISPE(I2));
  CRLF(1);
end;
CRLF(3);
OUT1(6, 'NFPXY=' '2D', NFPXY); CRLF(1);
GK1=GK2=0.000000000;

for I=1 step 1 until NFPXY do
begin
  I(GK1)=0.0100000000*GK1(0.0100000000) then SPACE(2);
  OUT1(6, '3D', FPNXY(I));
  I(JENTIER(I/19+0.0100000000)); JJ=JENTIER(I/20+0.0100000000);
  GK1(I/10-FLOAT(I)); GK2=I/20-FLOAT(JJ);
  I(GK2)=0.0100000000*GK2(0.0100000000) then CRLF(1);
end;
CRLF(1);
if NFPXY=NP2 v NMPXY=NP2 then
begin CRLF(3);
  OUT1(6, '*****Ifwounder#Ifyou#are#ammonof#sense,***');
  CRLF(1); go to L39;
end;
if PPP=1 then go to L910;
L2: for I=1 step 1 until NP2 do
begin
  for J=1 step 1 until NP2 do
  begin MAT(I, J)=0.000000000; end;
end;

for I=1 step 1 until NP do
begin
  A1=2-I-1;
  A2=2+I-1;
  for J=1 step 1 until A do
  begin
    if ENP(I, J)=0 then go to L3 else go to L4;
    KK=ENP(I, J);
    AA1=2-CNPE(I, J)-1;
    AA2=2-CNPE(I, J);

    for I1=1 step 1 until NP do
    begin
      A1=2-I1-1;
      A2=2+I1-1;
      for J1=1 step 1 until A do
      begin
        if ENP(I1, J1)=0 then go to L5 else go to L6;
        if ENP(I1, J1)=KK then go to L7 else go to L8;
        CB1=2-CNPE(I1, J1)-1;

```

- 310-

- 320-

- 330-

- 340-

- 350-

```

      BB2:=2*CNPF[I,JJ];
      MAT[A1,B1]:=MAT[A1,B1]+K[KK,AA1,BB1];
      MAT[A1,B2]:=MAT[A1,B2]+K[KK,AA1,BB2];
      MAT[A2,B1]:=MAT[A2,B1]+K[KK,AA2,BB1];
      MAT[A2,B2]:=MAT[A2,B2]+K[KK,AA2,BB2];
L8:      end;
L6:      end;
L4:      end;
      end;
      end;
L910:
if QW=0 then go to L99;
for I=1 step 1 until NP do
begin
  A2:=2*I;
  for J=1 step 1 until 3 do
  begin
    if CNPF[I,J]=0 then go to L9 else go to L10;
    L9: KK=CNPF[I,J]; LOAD[A2]=LOAD[A2]+WE[KK];
    end;
    L10:
    end;
L99:
for I=1 step 1 until NP2 do
begin
  for J=1 step 1 until NP2 do
  begin
    LOAD[I]=LOAD[I]-MAT[I,J]*DISP[J]; end;
  end;
end;
if PPP=1 then go to L920;
for I=1 step 1 until NMPXY do
begin
  for J=1 step 1 until NMPXY do
  begin
    I:=PNVY[I];
    J:=PNXY[J];
    MAT[I,J]:=MAT[I,J];
  end;
end;
for I=1 step 1 until NMPXY do
begin
  for J=1 step 1 until NMPXY do
  begin
    if I=J then MAT[I,NMPXY+J]=1.0000000000 else MAT[I,NMPXY+J]=0.0000000000;
  end;
end;
for I=1 step 1 until NMPXY do
begin
  O=2+NMPXY;

```

- 360-

- 370-

- 380-

- 390-

- 400-

```

    if MATHEP.NK=0.00000000 then go to L11 else P=NK+1;
    if P=NMPXY then go to L12;
L13: if MATHEP.NK=0.00000000 then go to L14 else P=P+1;
    if P=NMPXY then go to L12 else go to L13;
L14: for J=1 step 1 until 9 do
    begin MATHEP.JJ=MATHEP.JJ+MATHEP.JJ; end;
L11: RRMATHEP.NK=1;
    for J=1 step 1 until 9 do
    begin MATHEP.JJ=MATHEP.JJ/RB; end;
    for I=1 step 1 until NMPXY do
        begin
            RRMATHEP.I.NK=1;
            for J=1 step 1 until 9 do
            begin if I=NK then MATHEP.I.JJ=CHATEP.I.JJ-RB*MATHEP.I.JJ else MATHEP.I.JJ=MATHEP.I.JJ; end;
            end;
            if NK=NMPXY then go to L40;
            for I=1 step 1 until NMPXY do
            begin
                for J=1 step 1 until NMPXY do
                begin MATHEP.I.JJ=MATHEP.I.NMPXY+JJ; end;
            end;
        end;
L40:
    end;
L920:
    for I=1 step 1 until NMPXY do
    begin
        I=NMPXY+1; S=DISP(I);
        for J=1 step 1 until NMPXY do
        begin JJ=NMPXY+JJ; S=S+MATHEP.I.JJ+LOAD(JJ);
        end;
        DISP(I)=S;
    end;
    for I=1 step 1 until NMPXY do
    begin I=NMPXY+1; RFI=0.000000000; end;
    for I=1 step 1 until NMPXY do
    begin
        I=NMPXY+1;
        RFI=LOAD(I);
        for J=1 step 1 until NMPXY do
        begin JJ=NMPXY+JJ; RFI=RFI+MATHEP.I.JJ+DISP(JJ); end;
    end;
    EJECT; CRLF(1);
    OUTOC(,"113","PAGE=2"); CRLF(1);
    OUTOC(,"**RESULT=1*MODAL*POINT*INFORMATION**");
    CRLF(3);
    OUTOC(,"NP",RB,"LOAD(R)",RB,"LOAD(Z)",RB,"REAC(R)",RB,"REAC(Z)",
    RB,"DISP(R)",RB,"DISP(Z)",RB,"R",128,"Z");
    CRLF(2);
    GK1=GK2+GK3=GK+0.000000000;
    for I=1 step 1 until NP do

```

- 410-

- 420-

- 430-

- 440-

- 450-

```

begin
  I1=2*I-1; I2=2*I;
  GK1=GR1+1,OADF(I1); GK2=GK2+LOAD(I2);
  GK3=GK3+HF(I1); GK4=GK4+HF(I2);
  OUT1(A,'2ZD6',I1);
  OUT8(A,'2(-6ZD,6D),2(-7ZD,6D),2B,4(-6ZD,4D)',LOAD(I1),LOAD(I2),HF(I1),HF(I2),
    DISP(I1),DISP(I2),X(I1)+DISP(I1),Y(I1)+DISP(I2));
  CRLF(I);
end

```

```

CHLF(1);
- 460-

```

```

OUTO(6,'*GK',5);
OUTA(6,'2(-6ZD,5D),2(-7ZD,4D)',GK1,GK2,GK3,GK4); CRLF(1);
EJECT; CHLF(1);
OUTO(6,'117B,*PAGE#3'); CRLF(1);
OUTO(6,'*RESULT=2#(ELEMENT#INFORMATION#)#');
CHLF(3);
OUTO(4,'*HE',9B,'SIGMA=Z',7B,'SIGMA=R',7B,'SIGMA=TH',7B,'TAU=RZ',8B,'TAU=MAX',
  7B,'SIGMA=I',7B,'SIGMA=2',9B,'THETA',7B,'K#(X/Y)');
CHLF(2);

```

```

  for I=1 step 1 until 4 do
    for KK=1 step 1 until 4 do
      begin P(I,KK)=0.0; end;
- 470-

```

```

    for I=1 step 1 until 4 do
      for KK=1 step 1 until 4 do
        begin PIZUM(I,KK)=0.0; end;
      for I=1 step 1 until NE do
        begin GK3(I)=0.0; end;
      for I=1 step 1 until NE do
        begin

```

```

        for I=1 step 1 until NE do
          begin
- 480-

```

```

            I1=IF(I);
            JJ=JF(I);
            MM=MF(I);
            XI=XF(I);
            XJ=XF(JJ);
            XM=XF(MM);
            YI=YF(I);
            YJ=YF(JJ);
            YM=YF(MM);
            UV(I)=DISP(2*I-1);
- 490-

```

```

            UV(2)=DISP(2*I);
            UV(3)=DISP(2*JJ-1);
            UV(4)=DISP(2*JJ);
            UV(5)=DISP(2*MM-1);
            UV(6)=DISP(2*MM);
            C=EF(I)=(1-V(I))/(1+V(I))/(1-2*V(I));
            C11=C2=C;
            C12=C*V(I)/(1-V(I));
            C33=C*(1-2*V(I))/(2*(1-V(I)));
            D(1,1)=C2;
- 500-

```



```

1
end;
S1:=(SIGMA[1]+SIGMA[2])/2.000000000;
S2:=SQRT(0.25*(SIGMA[2]-SIGMA[1])^2+SIGMA[4]^2);
SIG1:=S1+S2;
SIG2:=S1-S2;
IF SIGMA[2]=SIGMA[1] then go to HHH1;
THETA:=28.6478972654*(ARCTAN(2.0*(SIGMA[4]/(SIGMA[2]-SIGMA[1]))));
IF SIGMA[2]<SIGMA[1] then THETA=THETA+90.0; go to HHH1;
IF SIGMA[2]>SIGMA[1] A SIGMA[4]>0 then THETA=THETA+0.0; go to HHH1;
IF SIGMA[2]>SIGMA[1] A SIGMA[4]<0 then THETA=THETA+180.00; go to HHH1;
HHH1:
IF SIGMA[2]=SIGMA[1] A SIGMA[4]>0 then THETA=45.0; go to HHH1;
IF SIGMA[2]=SIGMA[1] A SIGMA[4]<0 then THETA=135.0; go to HHH1;
IF SIGMA[2]=SIGMA[1] A SIGMA[4]=0 then THETA=360.0; go to HHH1;
HH1: THETA=THETA;
- 560-

L20:
IF SIGMA[1]=-0.0000001 A SIGMA[1]=0.0000001 then
GK1:=0.99.9999 with GK1:=SIGMA[2]/SIGMA[1];
OUT1(A:='2703H',I);
OUT1(A:='7(2R-32D.6D)*5R-22D.3D+4R-32D.4D'*SIGMA[1]*SIGMA[2]*SIGMA[3]*SIGMA[4]*S2*SIG1*SIG2*THETA*GK1);
- 570-
CRLF(1);
G1[1]:=SIGMA[2]; G2[1]:=SIGMA[1];
G3[1]:=SIGMA[4]; G4[1]:=G2; G5[1]:=SIGMA[3];
G6[1]:=G1; G6[1]:=G2;
G7[1]:=THETA; G8[1]:=GK1;
end;
EJECT; CRLF(1);
OUTOC6, '119R, PAGE#4'; CRLF(1);
OUTOC6, '***RESULT-3#C*MEAN*VALUE#AROUND#NODAL#POINT#***';
CRLF(3);
- 580-
OUTOC6, 'HP', 'PB', 'SIGMA-2', '7B', 'SIGMA-R', '7B', 'SIGM-TH', '7B', 'TAU-RZ', '8B', 'TAU-MAX', '7B', 'SIGM-1',
7B, 'SIGMA-2', '9B', 'THETA', '7B', 'K (R/Z)';
CRLF(2);
for I:=1 step 1 until HP do
begin
SIGMA[1]:=SIGMA[2]=SIGMA[3]=S2=SIG1=SIG2=THETA=GK1=0.000000000;
SIGMA[4]=0.0000000;
J:=0;
for J:=1 step 1 until 8 do
begin
KK:=FNPI+J;
if KK=0 then go to L50;
J:=J+1;
SIGMA[1]:=SIGMA[1]+G2[KK];
SIGMA[2]:=SIGMA[2]+G1[KK];
SIGMA[3]:=SIGMA[3]+G5[KK];
SIGMA[4]:=SIGMA[4]+G3[KK];
S2:=S2+G4[KK];
SIG1:=SIG1+G6[KK];
SIG2:=SIG2+G6[KK];
- 590-
- 600-

```

```

      THETA=THLTA+G7/TK1:
      GK1=GK1+G8/TK1:
    end:
  L30:
      GK2=FI OAT (JJ):
      SIGMA11=SIGMA11/GK2:
      SIGMA12=SIGMA12/GK2:
      SIGMA13=SIGMA13/GK2:
      SIGMA14=SIGMA14/GK2:
      SIG1=SIG1/GK2:
      SIG2=SIG2/GK2:
      THETA=THETA/GK2:
      GK1=GV1/GK2:
      OUT1(6,'72039',1):
      OUT0(6,'7(29-3ZD.6U).58-2ZD.3U.48-3ZD.40':
      SIGMA11=SIGMA11+SIGMA12+SIGMA13+SIGMA14+S2+SIG1+SIG2+THETA+GK1:
      CRLF(1):
    end:
  go to L34:

  L12: CRLF(5): OUT0(6,'CAN#NOT#CAPRY#ON'): CRLF(1):
  L34: FJFCT:
    end:
  end: if KEY(8) then go to START:
end:

```

- 610-

- 620-

有限要素法による軸対称問題解析用プログラム (ALGOL用) NAXIFEM

境界条件としては強制変位と荷重条件が考慮されている。前のプログラムAXIFEMにさらに任意方向のローラ条件が考慮されている。

入力データ作成順序

施工研究室NAXIFEM用

A10, A20, A30, A40, A50,
NP, NE, NEP, NFPXY,
FNPXY[1.....NFPXY],
(GAMMA[1.....NEP], EE[1.....NEP],
V[1.....NEP]).....(P, N), ...
P, GK11[1.....NP],
R[1.....NP],
Z[1.....NP],
(IE[1.....NE], JE[1.....NE], ME[1.....NE])
DDD, P, DISP[1.....NP2],
LLL, { OW(P, LOAD[1.....NP2]).....
 OW, (P, LOAD[1.....NP2]).....

計算名 (5×8文字)

節点数, 要素数等

固定節点番号

} 弾性パターン情報

($\Sigma P = NE$)

傾斜角度 ($\Sigma P = NP$)

R 座 標

Z 座 標

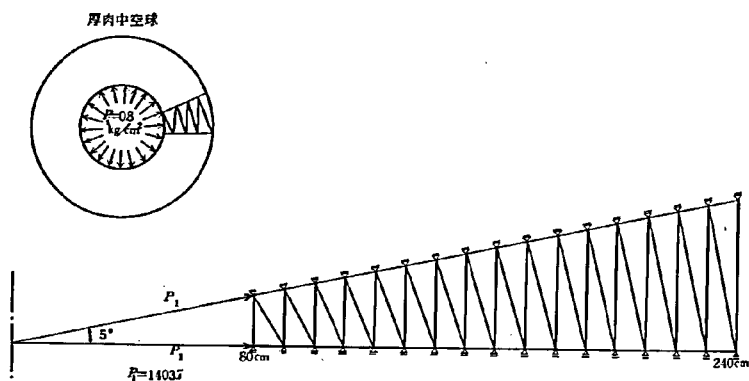
要素番号と節点番号の関係

強制変位 $\Sigma P = NP2$

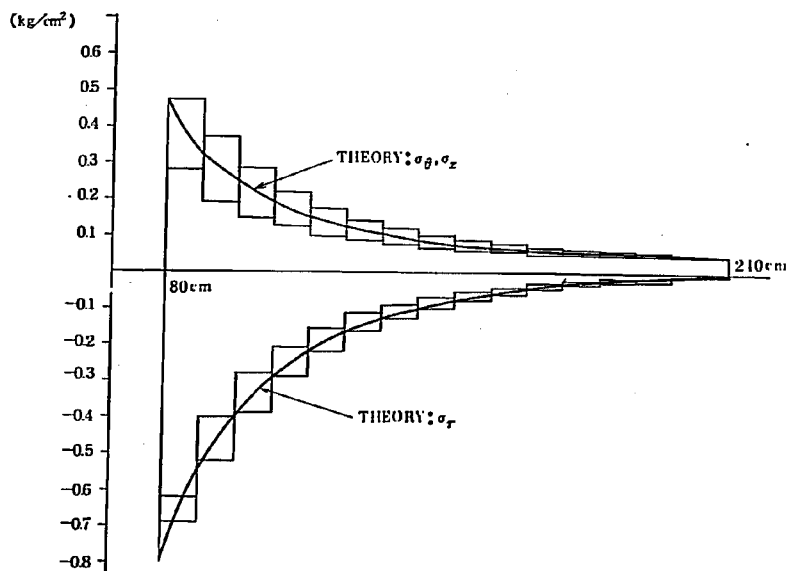
荷重条件LLL: くりかえし条件数

($\Sigma P = NP2$)

このプログラムを用いた例題として厚肉中空球が内圧 $P_i = 0.8 \text{ kg/cm}^2$ をうけた場合の応力解析を行なった。以下に示す。



付 図 1



付 図 2

プログラム

(NAXIFEM)

```

begin
  comment NAXIFEW TANAKA
    This program is the axisymmetric F.E.M.
    This is a program for stress and deformation analysis using Finite Element Method
    with vertical and horizontal roller in any direction (triangular element)
  HOW TO GIVE DATA
    A10,A20,A30,A40,A50: Name of calculation (double real)
    NP: number of nodal points (integer)
    NE: number of elements (integer)
    NEP: number of elastic patterns (integer)

    NFPXY: number of fixed points considering R and Z direction (integer) - 10-
    FPNXY[1]: fixed points number considering R and Z direction (integer array)
    GAMMA[1]: bulk density of each element (g/cm3) (array)
    EE[1]: Young's modulus of each element (kg/cm2) (array)
    V[1]: Poisson's ratio (array)
    P: number of seriesed same data (integer)
    N: elastic pattern number (integer)
    X[1]: R-coordinate of nodal points (cm) (array)
    Y[1]: Z-coordinate of nodal points (cm) (array)
    IE[1]: nodal point number of I corner in each element (integer array)

    JEC[1]: nodal point number of J corner in each element (integer array) - 20-
    MEC[1]: nodal point number of M corner in each element (integer array)
    DDU: number of repeat for initial displacements data (integer)
    DISP[1]: initial displacement of each nodal point in R and Z direction (cm) (array)
    GK11: angle of roller (DO) (array)
    LLL: number of repeat for loads data (integer)
    OW: when you think of own weight, OW=1
    when you do not think of it, OW=0
    LOAD[1]: loads for each nodal point in R and Z direction (kg) (array)
  UNDEN TO GIVE DATA
    A10, A20, A30, A40, A50, NP, NE, NEP, - 30-
    NFPXY, FPNXY[1...NFPXY],
    ( GAMMA[1...NEP], EE[1...NEP], V[1...NEP] ) ... ( P, N ) ... (NP=NE),
    GK11[1...NP], X[1...NP], Y[1...NP], ( IE[1...NE], JEC[1...NE], MEC[1...NE] )....
    DDU, ( P, DISP[1...NP2] ) ... (NP=NP2),
    LLL, OW, ( P, LOAD[1...NP2] ) ... (NP=NP2),
    1
  double real A10,A20,A30,A40,A50:
  real X[1],X[2],X[3],Y[1],Y[2],Y[3],X[4],X[5],X[6],Y[4],Y[5],Y[6],C1,C2,F1,F2,
    A[1],B[1],A[2],B[2],A[3],B[3],I1,I2,I3,I4,I5,I6,I7,I8,
    GK11,THET,SS,YB,XB,YJ,XJ,C33,CX,C22,C12,C11,C,
    S,RD,S1,S2,SIG1,SIG2,THETA,T,GK1,GK2,GK3,GK4,CDI
  integer NP,NE,NMP,PS,QW,NP2,NMP1,I,J,L,N,MS,I1,JJ,MM,KK,A1,A2,B1,B2,
    NMP2,P1,NF,P,Q,I1,I2,AA1,AA2,BB1,BB2,NMPXY,NMPXY2,NFPXY,NEP,
    DD,DDD,LL,LLL,PPP:
  START:
  READA(A10,A20,A30,A40,A50):
  READ(NP,NE,NEP,NFPXY):
  NP2=2*NP: NMPXY=NP2-NFPXY: NMPXY2=2*NMPXY:
  begin

```

```

array  EF,GAHMA,V,W,E[1:NE],Y,Y[1:NP],LOAD,DISP,DISPA,RF[1:NP2],
        CC,AT,A,K[1:NE,1:6,1:6],D[1:4,1:4],HIZUMI[1:4,1:6],GG3[1:NE],
        K[1:NE,1:6,1:6],MAT[1:NP2,1:NP2],MATM[1:NMPXY,1:NMPXY2],MA[1:4,1:6],
        EP1,EP2,EP3[1:NE],UV[1:6],SIGMA[1:4],
        FA[1:NP],LF[1:NP,1:2,1:2],G1,G2,G3,G4,G5,G6,G7,G8[1:NE],
integer array IE,JE,ME[1:NE],ENP,CHP[1:NP,1:8],MPNXY[1:NMPXY],FPNXY[1:NFPXY],
READHPXY(FPNXY,1,NFPXY),
if NFPXY=NP2 v NMPXY=NP2 then go to L500:
for [n] step 1 until NMPXY do
begin
    if FPNXY[1]=1 then MPNXY[1]=1 else go to L100:
end:
L100:
N=Jn[1]
for [n2] step 1 until NFPXY do
begin
    if FPNXY[1]=1 then MPNXY[1]=1 else go to L200:
    if [1]=1 then go to L200 else go to L300:
    L200: MM=0:
    for [JJ] step 1 until (N+[1]=2) do
begin
    MM=MM+1:
    FPNXY[JJ]=FPNXY[1]-1+MM:
end:
N=JJ:
L300:
end:
L400:
MM=0:
for [JJ] step 1 until NMPXY do
begin
    MM=MM+1:
    FPNXY[JJ]=FPNXY[NFPXY]+MM:
end:
L500:
for [n] step 1 until NE do
begin
    READ(GK1,GK2,GK3),
    EP1[1]=GK1: EP2[1]=GK2: EP3[1]=GK3:
end:
P=0:
for [n] step 1 until NE do
begin
    READ(P,N):
    for [n2] step 1 until (P=0) do
begin
        GAKMA[1]=EP1[N]:
        FF[1]=EP2[N]:
        VF[1]=EP3[N]:
end:
    if (P=0)=NE then go to L600:
    Q=P+N:

```

```

ends;
L600:
  for I=1 step 1 until NP do
    begin FA[I]=0.0; end;
    P=0; GK1=0.0;
    for J=1 step 1 until NP do
      begin
        READ(P,GK1);
        for J=(1+0) step 1 until (P+0) do
          begin FA[IJ]=GK1+0.0174533; end;

        if (P+0)=NP then go to L77;
        P=P+0;
      end;
    L77:
    READARRAY(X,1,NP);
    READARRAY(Y,1,NP);
    for I=1 step 1 until NE do
      begin READ(A1,A2,B1);
             IE[I]=A1; JE[I]=A2; ME[I]=B1;
      end;

      for I=1 step 1 until NP do
        for J=1 step 1 until 8 do
          begin ENP[I,J]=CNP[I,J]=0; end;
        for I=1 step 1 until NP do
          begin A1=2=B1=0;
            for J=1 step 1 until NE do
              begin
                if IE[J]=I then begin A1=A1+1; ENP[I,A1]=J; CNP[I,A1]=1; end;
                if JE[J]=I then begin A1=A1+1; ENP[I,A1]=J; CNP[I,A1]=2; end;
                if ME[J]=I then begin A1=A1+1; ENP[I,A1]=J; CNP[I,A1]=3; end;
              end;
            end;
          end;
        READ(DD0); PPP=0;
        for DD=1 step 1 until DD0 do
          begin
            P=0;
            for J=1 step 1 until NP2 do
              begin
                READ(P,GK1);
                for J=(1+0) step 1 until (P+0) do
                  DISPA[IJ]=GK1;
                if (P+0)=NP2 then go to L700;
                P=P+0;
              end;
            L700:
            READ(LL0);
            for LL=1 step 1 until LLL do
              begin
                for I=1 step 1 until NP2 do
                  DISP[I]=DISPA[I];

```

- 110-

- 120-

- 130-

- 140-

- 150-

```

P=0;
HEAD(QW);
for i=1 step 1 until NP2 do
begin
  READ(P,G*2);
  for j=(1-Q) step 1 until (P+Q) do
    LQAD(J)=G*2;
  if (P+Q)=NP2 then go to L800;
  Q=P+Q;
end;

L800:
PPP=PPP+1; if PPP=1 then go to L900;
for i=1 step 1 until NE do
for j=1 step 1 until 6 do
for k=1 step 1 until 6 do
begin
  AT(I,I,KK)=A(I,I,KK)=K(I,I,KK)=CCE(I,I,KK)=0.0;
end;
for i=1 step 1 until NE do
begin
  IJ=JF(I);
  JJ=JF(I);
  MM=MF(I);
  XI=X(I);
  XJ=X(JJ);
  YI=Y(I);
  YJ=Y(JJ);
  YM=YM(I);
  XI=XI-XJ;

  XJ=MY-J-X;
  XM=X-XI;
  YI=YI-YJ;
  YJ=MY-Y;
  YM=Y-YI;
  CE=EI=(1-V(I))/(1+V(I))/(1-2*V(I));
  C1=C2=C;
  C12=CX=C*(V(I)/(1-V(I)));
  C3=C*(1-2*V(I))/(2*(1-V(I)));
  XJ=X-IJ;

  YJ=Y-VJ;
  if XI=XM then go to SX1 else go to SX2;
  SX1: AM=BI=0.0; go to JJ1;
  SX2: AM=(XM*YI-YI*YM)/(XM-XI);
  JJ1: if XJ=XM then go to S3 else go to S4;
  S3: AMJ=(XJ-YM-XM*YJ)/(XJ-XM);
  S4: AMJ=(YJ-YM)/(XJ-XM);
  JJ2: if XI=XJ then go to S5 else go to S6;

```

- 160-

- 170-

- 180-

- 190-

- 200-

S5: AJI=PIJIN0.01: 00 10 JJ31
 S6: AJI=(X1+YJ-XJ*YI)/(X1-XJ)
 PJI=(YI-YJ)/(X1-XJ)
 JJ3: I11=(XJ1+YI1-XM1+YJ1)/21
 I12=(X1+XJ+XM)*I11/31
 I13=(Y1+YJ+YI)*I11/31
 I14=3*I11/(X1+XM+XJ)
 I15=0.5*((CAJIT2-AMJT2)*LN(XI)+
 (AMJT2-AJIT2)*LN(XJ)+
 (AMJT2-AMJT2)*LN(XM))
 +AIM*BI1*(XM-X1)+AMJ*BMJ*(XJ-XM)+AJI*BJI*(X1-XJ)
 + (1/4)*(CBIT2*(XMT2-XIT2)+BMJT2*(XJT2-XMT2)+BJIT2*(XIT2-XJT2))
 I16=(1/3)*((CAJIT3-AMJT3)*LN(X1)+(AMJT3-AJIT3)*LN(XJ)+(AMJT3-AMJT3)*LN(XM))
 +AIMT2*BI1*(XM-X1)+AMJT2*BMJ*(XJ-XM)+AJIT2*BJI*(X1-XJ)
 +C0.9)*(AIM*BI1*(XMT2-XIT2)+AMJ*BMJT2*(XJT2-XMT2)+AJI*BJIT2*(XIT2-XJT2))
 + (1/9)*(CBIT3*(XMT3-XIT3)+BMJT3*(XJT3-XMT3)+BJIT3*(XIT3-XJT3))
 I17=CAJIT2-XJT2+XMT2+(X1+XM+XJ)*I15
 I18=CAJIT2-XJT2+XMT2+(X1+XM+XJ)*I15
 K1I1.1.13=C11*I14
 K1I1.2.13=(CX+C11)*I11
 K1I1.3.13=C11*I15
 K1I1.4.13=K1I1.5.13=0.0
 K1I1.6.13=C12*I11
 K1I1.7.21=2*(C11+CX)*I12
 K1I1.7.21=(C11+CX)*I13
 K1I1.4.21=0.0
 K1I1.5.21=0.0
 K1I1.4.21=2*C12*I12
 K1I1.7.31=C11*I6+C33*I12
 K1I1.4.31=0.0
 K1I1.5.31=C33*I12
 K1I1.6.31=C12*I13
 K1I1.7.31=C33*I12
 K1I1.4.61=C22*I12
 K1I1.1.21=K1I1.2.11
 K1I1.1.31=K1I1.3.11
 K1I1.1.41=K1I1.4.11
 K1I1.1.61=K1I1.6.11
 K1I1.1.51=K1I1.5.11
 K1I1.2.31=K1I1.3.21
 K1I1.2.41=K1I1.4.21
 K1I1.2.51=K1I1.5.21
 K1I1.2.61=K1I1.6.21
 K1I1.3.41=K1I1.4.31
 K1I1.3.51=K1I1.5.31
 K1I1.3.61=K1I1.6.31
 K1I1.4.41=K1I1.4.51=K1I1.4.61=K1I1.5.41=K1I1.5.61=K1I1.6.41=K1I1.6.51=0.0
 A1I.1.11=(XJ+YI-XM+YJ)/(2.0*I11)
 A1I.2.11=(YJ-YI)/(2.0*I11)
 A1I.3.11=(XM-XJ)/(2.0*I11)

- 210-

- 220-

- 230-

- 240-

- 250-

```

AC[1,4,1]=AC[1,5,1]*AC[1,6,1]*AC[1,1,2]*AC[1,2,2]*AC[1,3,2]*0.0;
AC[1,4,2]=AC[1,1,1];
AC[1,5,2]=AC[1,2,1];
AC[1,6,2]=AC[1,3,1];
AC[1,1,3]=(X*Y-Y*X)/(2.0*111);
AC[1,2,3]=(Y*X-X*Y)/(2.0*111);
AC[1,3,3]=(X*Y-Y*X)/(2.0*111);
AC[1,4,3]=AC[1,5,3]*AC[1,6,3]*AC[1,1,4]*AC[1,2,4]*AC[1,3,4]*0.0;
AC[1,4,4]=AC[1,1,1];
AC[1,5,4]=AC[1,2,1];
AC[1,6,4]=AC[1,3,1];
- 260-
AC[1,4,5]=AC[1,1,1];
AC[1,5,5]=X*Y-Y*X/(2.0*111);
AC[1,6,5]=Y*X-X*Y/(2.0*111);
AC[1,1,6]=X*Y-Y*X/(2.0*111);
AC[1,4,6]=AC[1,5,6]*AC[1,6,6]*AC[1,1,6]*AC[1,2,6]*AC[1,3,6]*0.0;
AC[1,4,7]=AC[1,1,1];
AC[1,5,7]=AC[1,2,1];
AC[1,6,7]=AC[1,3,1];
for [i]=1 step 1 until 6 do
- 270-
for [j]=1 step 1 until 6 do
begin
AT[1,1,12]=AT[1,12,11]; end;
WE[1,1]=0.0; WE[2,1]=2.0*3.14159265*EAMMA[1]*(X+XJ+XM)*11*(1/9.0);
for [i]=1 step 1 until 6 do
begin
for [j]=1 step 1 until 6 do
begin
SS[0,0];
for [j]=1 step 1 until 6 do
SS[5]=AC[1,1,11]*X[1,1,11]*X[1,1,11];
- 280-
CC[1,1,11]=SS;
end;
end;
for [i]=1 step 1 until 6 do
for [j]=1 step 1 until 6 do
begin
SS[0,0];
for [j]=1 step 1 until 6 do
SS[5]=CC[1,1,11]*AC[1,1,11]*X[1,1,11];
- 290-
V[1,1,11]=2.0*3.14159265*SS;
end;
end;
L903:
CRLF(1); OUT(6,1034,'MAXIFEM PAGE1'); CRLF(1);
OUT(6,104,'*****STRESS*AND*DEFORMATION*ANALYSIS*BY*FINITE*ELEMENT*METHOD***',5B,'(K*CM)');
CRLF(1);
OUT(6,105,'*****',8A,8A,8A,8A,8A,A10,A20,A30,A40,A50);
CRLF(6);
OUT(6,106,'INPUT*DATA*');
- 300-

```

```

CRLF(3); PSM:=T360.0;
IF PS=1 then OUTO(6,'5R','AXISYMMETRICANALYSIS') else
  OUTO(6,'5B','AXISYMMETRICANALYSIS');
OUT1(6,'10A','TWICKNESS','3D.ZD'T);
IF Own0 then OUTO(6,'#####ON#WEIGHT');
CRLF(3);
OUTO(6,'**MF'.9R,'GAMMA'.5B,'YOUNG-M'.4R,'POISSONS-R'.8B,'|E'.5B,'JE'.5B,'ME'');
CRLF(2);
for I=1 step 1 until NE do
begin
  OUT7(6,'.2ZD.7B.2D.2R7ZD.0.7BD.3D.9B7ZD.4B2ZD.4B2ZD',
    | GAMMA[I].EE[I].VE[I].|E[I].JEE[I].ME[I]);
  CRLF(1);
end;
CRLF(4);
OUTO(6,'.NF'.5B,'(ORDER)'.9R,'R0'.9B,'Z0'.8B,'LOAD(R)'.5B,'LOAD(Z)'.
  6B,'DISP(X)'.3B,'DISP(Z)');
CRLF(2);
for I=1 step 1 until NP do
begin
  I1=2+I-1; I2=I+1;
  OUT1(6,'.2ZD.3B2ZD.8ZD.2B-6ZD.2D-.6ZD.2D'.|I1.I2.X[I].Y[I]);
  OUT4(6,'.R.2(5-6ZD.2D).3R.2(R-3ZD.3D)'.LOAD[I1].LOAD[I2].DISP[E11].DISP[E2]);
  CRLF(1);
end;
CRLF(3);
OUT1(6,'.NFPXY'.2ZD'.NFPXY); CRLF(1);
GK1=GK2=0.000000000;
for I=1 step 1 until NFPXY do
begin
  IF (GK1>0.0100000000&GK1<0.0100000000) then SPACE(2);
  OUT1(6,'.3ZD'.FNPXY[I]);
  I1=ENTIER(I/10+0.0100000000); J=J+ENTIER(I/20+0.0100000000);
  GK1/20=FLUAT(I1); GK2/20=FLUAT(JJ);
  IF (GK2>0.0100000000&GK2<0.0100000000) then CRLF(1);
end;
CRLF(1);
IF NFPXY=NP2 v NMPXY=NP2 then
begin
  CRLF(3);
  OUTO(6,'####founder*!#you#are#ammonof#sense.###');
  CRLF(1); go to L39;
end;
IF PPP=1 then do to L910;
L2: for I=1 step 1 until NP2 do
begin
  for J=1 step 1 until NP2 do
  begin
    MAT[I,J]=0.000000000; end;
  end;
  for I=1 step 1 until NP do
  begin
    AJ=2+I-1;

```

```

A2:=2+1;
for J:=1 step 1 until 8 do
begin
  if ENP[1,J]=0 then go to L3 else go to L4;
  KK:=ENP[1,J];
  AA1:=2+ENP[1,J]-1;
  AA2:=2+ENP[1,J];
  for I:=1 step 1 until NP do
  begin
    R1:=2+I-1;
    A2:=1+I;
    for JJ:=1 step 1 until 8 do
    begin
      if ENP[1,JJ]=0 then go to L5 else go to L6;
      if ENP[1,JJ]=KK then go to L7 else go to L8;
      BB1:=2+ENP[1,JJ]-1;
      BB2:=2+ENP[1,JJ];
      MAT[A1,B1]:=MATEA1,B1+K*KK,AA1,BB1;
      MAT[A1,B2]:=MATEA1,B2+K*KK,AA1,BB2;
      MAT[A2,B1]:=MAT[A2,B1]+K*KK,AA2,BB1;
      MAT[A2,B2]:=MAT[A2,B2]+K*KK,AA2,BB2;
    end;
  end;
end;
for I:=1 step 1 until NP do
begin
  for J:=1 step 1 until 2 do
  for KK:=1 step 1 until 2 do
  begin
    LFC[1,J,KK]:=0.0; end;
    for I:=1 step 1 until NP do
    begin
      LFC[1,1,1]:=LFC[1,2,2]*COS(FA[1,1]);
      LFC[1,1,2]:=SIN(FA[1,1]); LFC[1,2,1]:=LFC[1,1,2];
      for J:=1 step 1 until NP do
      begin
        MAT[2*J-1,2*J-1]:=LFC[1,1,1]*MAT[2*J-1,2*J-1]+LFC[1,2,1]*MAT[2*J-1,2*J];
        MAT[2*J-1,2*J]:=LFC[1,1,1]*MAT[2*J-1,2*J]+LFC[1,2,1]*MAT[2*J-1,2*J+1];
        MAT[2*J,2*J-1]:=LFC[1,1,2]*MAT[2*J,2*J-1]+LFC[1,2,2]*MAT[2*J,2*J];
        MAT[2*J,2*J]:=LFC[1,1,2]*MAT[2*J,2*J]+LFC[1,2,2]*MAT[2*J,2*J+1];
      end;
    end;
  end;
  for I:=1 step 1 until NP do
  for J:=1 step 1 until NP do
  begin
    MAT[2*J-1,2*J-1]:=LFC[1,1,1]*MAT[2*J-1,2*J-1]+LFC[1,2,1]*MAT[2*J-1,2*J];
    MAT[2*J-1,2*J]:=LFC[1,1,2]*MAT[2*J-1,2*J]+LFC[1,2,2]*MAT[2*J-1,2*J+1];
  end;
end;

```

- 360-

- 370-

- 380-

- 390-

- 400-

```

MAT[2,J,2+1-1]=LFC[1,1,1]*MAT[2,J,2+1-1]+LFC[1,2,1]*MAT[2,J,2+1];
MAT[2,J,2+1]=LFC[1,1,2]*MAT[2,J,2+1-1]+LFC[1,2,2]*MAT[2,J,2+1];
end;
L910:
if OW=0 then go to L99;
for I=1 step 1 until NP do
begin
  A2=2+1;
  for J=1 step 1 until 8 do
  begin
    if FNP[I,J]=0 then go to L9 else go to L10;
    L9: KK=FNP[I,J]; LOAD[A2]=LOAD[A2]+WEKK;
  end;
  L10:
end;
end;
L99:
for I=1 step 1 until NP2 do
begin
  for J=1 step 1 until NP2 do
  begin LOAD[I]=LOAD[I]-MAT[I,J]*DISP[J]; end;
end;
if PPP=1 then go to L920;
for I=1 step 1 until NMPXY do
begin
  for J=1 step 1 until NMPXY do
  begin
    IIM=NMPXY[I];
    JJ=NMPXY[J];
    MAT[I,J]=MAT[I,J];
  end;
end;
for I=1 step 1 until NMPXY do
begin
  for J=1 step 1 until NMPXY do
  begin
    if I=J then MAT[I,NMPXY+J]=1.000000000 else MAT[I,NMPXY+J]=0.000000000;
  end;
end;
for NK=1 step 1 until NMPXY do
begin
  Q=2+NMPXY;
  if MAT[NK,NK]=0.000000000 then go to L11 else P=NK+1;
  if P>NMPXY then go to L12;
L13: if MAT[P,NK]=0.000000000 then go to L14 else P=P+1;
  if P>NMPXY then go to L12 else go to L13;
L14: for J=1 step 1 until Q do
  begin MAT[NK,J]=MAT[NK,J]+MAT[P,J]; end;
L11: RB=MAT[NK,NK];
  for J=1 step 1 until Q do
  begin MAT[NK,J]=MAT[NK,J]/RB; end;

```

- 410-

- 420-

- 430-

- 440-

- 450-

```

for [n] step 1 until NMPXY do
begin
  DUMMATH[NK];
  for [m] step 1 until Q do
  begin if [NK] then MATH[I,J]=MATH[I,J]-RB*MATH[NK,J] else MATH[I,J]=MATH[NK,J] end;
end;
if NK<NMPXY then go to L40;
for [m] step 1 until NMPXY do
begin
  for [j] step 1 until NMPXY do
  begin MATH[I,J]=MATH[I,NMPXY+J] end;
end;
L40:
end;
L420:
for [m] step 1 until NMPXY do
begin
  [MPPHXYF]; S=DISP[1];
  for [j] step 1 until NMPXY do
  begin JJ=MPPHYF[J]; S=S+MATH[I,J]*LOAD[J];
  end;
  DISP[1]=S;
end;
for [m] step 1 until NP do
begin
  [SP2]=1-[LFC[1,1]*DISP[2]-1]+LFC[1,2]*DISP[2];
  [SP2]=1-[LFC[2,1]*DISP[2]-1]+LFC[2,2]*DISP[2];
end;
for [m] step 1 until NMPXY do
begin [MPPHXYF]; RFI[1]=0.000000000; end;
for [m] step 1 until NMPXY do
begin
  [MPPHXYF];
  RFI[1]=LOAD[1];
  for [j] step 1 until NMPXY do
  begin JJ=MPPHYF[J]; RFI[1]=RFI[1]+MATH[I,J]*DISP[J]; end;
end;
EJECT; CRLF(1);
OUTO(6,'11*'*PAGE*2'); CRLF(1);
OUTO(6,'**RESULT=1*(NODAL*POINT#)INFORMATION#*');
CRLF(3);
OUTO(6,' NP*RB*LOAD(R)*BH*LOAD(Z)*.9B*REAC(R)*.9B*REAC(Z)*.
  9B*DISP(Z)*.7B*R*.12B*Z');
CRLF(2);
GK1=GK2=GK3=GK4=0.000000000;
for [m] step 1 until NP do
begin
  [M2]=1-[LFC[1,1]*GK1+GK2*LOAD[1];
  GK1=GK2*LOAD[1]; GK2=GK2*LOAD[2];
  GK3=GK3+RFI[1]; GK4=GK4+RFI[2];
  OUTL(6,'220B*');

```

- 460-

- 470-

- 480-

- 490-

- 500-

```

OUTB(4,'2(-6ZD.6D)+2(-7ZD.6D)+2B,4(-6ZD.4D)',LOAD(11),LOAD(12),RFE(11),RFE(12),
DISP(11),DISP(12),X(1)+DISP(11),Y(1)+DISP(12));
CRLF(1);
end;
CHLF(1);
OUTO(6,'*GK',9);
OUTA(6,'2(-6ZD.6D)+2(-7ZD.6D)',GK1,GK2,GK3,GK4); CRLF(1);
EJECT; CRLF(1);
OUTO(6,'1174,'PAGE#3'); CRLF(1);
OUTO(6,'**RESULT=2*(#LEMENT#INFORMAT(ON#)*)');

CHLF(3);
OUTO(4,'*ME',9B,'SIGMA=2',7B,'SIGMA=R',7B,'SIGMA=TH',7B,'TAU=RZ',8B,'TAU=MAX',
7B,'SIGMA=1',7B,'SIGMA=2',9B,'THETA',7B,'K*(X/Y)');
CRLF(2);
for [m] step 1 until 4 do
for [n] step 1 until 4 do
begin D(11,KK)=0.0; end;

for [m] step 1 until 4 do
for [n] step 1 until 6 do

begin HIZUM(11,KK)=0.0; end;
for [m] step 1 until NE do
begin G63(11)=0.0; end;
for [n] step 1 until NE do
begin
I(1)=F(1);
J(1)=J(1);
MM(1)=F(1);
X(1)=X(1);
XJ(1)=J(1);

XM=X(1);
Y=Y(1);
YJ=Y(1);
YM=Y(1);
UV(1)=DISP(2+1-1);
UV(2)=DISP(2+1);
UV(3)=DISP(2+JJ-1);
UV(4)=DISP(2+JJ);
UV(5)=DISP(2+MM-1);
UV(6)=DISP(2+MM);

CHEE(1)=C(1-V(1))/(1+V(1))/(1-2*V(1));
C11=C22=C;
C12=CX=C*(V(1)/(1-V(1)));
C33=C*(1-2*V(1))/(2*(1-V(1)));
D(1,1)=C22;
D(1,2)=C12;
D(1,3)=C12;
D(1,4)=D(2,4)=D(3,4)=D(4,1)=D(4,2)=D(4,3)=0.0;
D(2,1)=D(1,2);
D(2,2)=C11;

```

- 510-

- 520-

- 530-

- 540-

- 550-

```

DE2.5)M(CX:
DE3.1)M(F1.3):
DE3.2)M(F2.3):
DE3.3)M(C11:
DE4.4)M(C31:
XB=(X1-XJ+XM)/3:
YB=(Y1-YJ+YM)/3:
XJ1=XJ-X1:
YJ1=YJ-Y1:
YM1=YM-Y1:

- 560-
XM1=X1-X1:
I1=(CXJ+YM1-YM1+YJ1)/2:
HI(ZUMI[1.1])=HI(ZUMI[1.3])=HI(ZUMI[1.5])=HI(ZUMI[2.2])=HI(ZUMI[2.4])=
HI(ZUMI[2.6])=HI(ZUMI[3.2])=HI(ZUMI[3.4])=HI(ZUMI[3.6])=0.01
HI(ZUMI[1.2])=(XM-XJ)/(2*I1):
HI(ZUMI[1.4])=(X1-XM)/(2*I1):
HI(ZUMI[1.6])=(YJ-X1)/(2*I1):
HI(ZUMI[2.1])=(YJ-YM)/(2*I1):
HI(ZUMI[2.3])=(YM-Y1)/(2*I1):
HI(ZUMI[2.5])=(Y1-YJ)/(2*I1):

- 570-
HI(ZUMI[3.1])=(CXJ+YM-XM+YJ)/XB+(CYJ+YM)+(XM-XJ)+(YB/XB))/(2*I1):
HI(ZUMI[3.3])=(XM+Y1-X1+YM)/XB+(YM-Y1)+(X1-XM)+(YB/XB))/(2*I1):
HI(ZUMI[3.5])=(X1+YJ-XJ+Y1)/XB+(Y1-YJ)+(XJ-X1)+(YB/XB))/(2*I1):
HI(ZUMI[4.1])=(XM-YJ)/(2*I1):
HI(ZUMI[4.3])=(YJ-YM)/(2*I1):
HI(ZUMI[4.5])=(X1-YM)/(2*I1):
HI(ZUMI[4.7])=(YM-Y1)/(2*I1):
HI(ZUMI[4.9])=(XJ-X1)/(2*I1):
HI(ZUMI[4.6])=(Y1-YJ)/(2.0*I1):
for I1=1 step 1 until 4 do

- 580-
for K1=1 step 1 until 6 do
begin
S0.0:
for JJ=1 step 1 until 4 do
S=S+D[I1,JJ]*HI(ZUMI[J,J,KK]):
MA[I1,KK]=S:
end:
for J1=1 step 1 until 4 do
begin
S0.00000000:
for L1=1 step 1 until 6 do
begin S=S+MA[J1,L1]*HVL1: end:
SIGMA[L1]=S:
end:
S1=(SIGMA[1]+SIGMA[2])/2.00000000:
S2=50*(S1-S2)+(SIGMA[2]-SIGMA[1])/2+S1:
S1=S2:
S2=S1:
S1=S2:
S2=S1:
if SIGMA[2]=SIGMA[1] then go to HHH1:
THETA=28.64788775654*ARCTAN(2.0*SIGMA[4]/(SIGMA[2]-SIGMA[1])):

- 600-

```

```

IF SIGMA[2]<SIGMA[1] THEN THETA=THETA+90.0; GO TO HH1;
IF SIGMA[2]>SIGMA[1] A SIGMA[4]=0 THEN THETA=THETA+0.0; GO TO HH1;
IF SIGMA[2]>SIGMA[1] A SIGMA[4]<0 THEN THETA=THETA+180.00; GO TO HH1;
HH1:
IF SIGMA[2]=SIGMA[1] A SIGMA[4]>0 THEN THETA=45.0; GO TO HH1;
IF SIGMA[2]=SIGMA[1] A SIGMA[4]<0 THEN THETA=135.0; GO TO HH1;
IF SIGMA[2]=SIGMA[1] A SIGMA[4]=0 THEN THETA=360.0; GO TO HH1;
HH1: THETA=THETA;

```

L20:

```

IF SIGMA[1]=0.0000001 A SIGMA[1]=0.0000001 THEN

```

- 610-

```

GK1=999.9999 ELSE GK1=SIGMA[2]/SIGMA[1];
OUT1(6,'27038',1);
OUT9(6,'7(28=32D.6D),5B=22D.3D,4R=32D.4D',SIGMA[1],SIGMA[2],SIGMA[3],SIGMA[4],S2,SIG1,SIG2,THETA,GK1);
CPLF(1);
G1[1]=SIGMA[2]; G2[1]=SIGMA[1];
G3[1]=SIGMA[4]; G4[1]=S2; GG3[1]=SIGMA[3];
G5[1]=SIG1; G6[1]=SIG2;
G7[1]=THETA; G8[1]=GK1;
END;
EJECT; CRLF(1);

```

- 620-

```

OUTO(6,'113R','PAGE#4'); CRLF(1);
OUTO(6,'*RESULT=3#(MEAN#VALUE#AROUND#NODAL#POINT#)*');
CRLF(3);
OUTO(6,'NP',9H,'SIGMA=Z',7R,'SIGMA=R',7B,'SIGM=TH',7B,'TAU=RZ',8B,'TAU=MAX',7B,'SIGM=1',
7R,'SIGMA=2',9R,'THETA',7R,'K#(X/Y)');
CRLF(2);
FOR I=1 STEP 1 UNTIL NP DO
BEGIN

```

```

SIGMA[1]=SIGMA[2]=SIGMA[3]=S2=SIG1=SIG2=THETA=GK1=0.000000000;
SIGMA[4]=0.0000000;

```

- 630-

```

JJ=0;
FOR J=1 STEP 1 UNTIL A DO
BEGIN

```

```

KK=ENP[I,J];
IF KK=0 THEN GO TO L50;
JJ=JJ+1;
SIGMA[1]=SIGMA[1]+G2[KK];
SIGMA[2]=SIGMA[2]+G1[KK];
SIGMA[3]=SIGMA[3]+GG3[KK];
SIGMA[4]=SIGMA[4]+G3[KK];

```

- 640-

```

S2=S2+G4[KK];
SIG1=SIG1+G5[KK];
SIG2=SIG2+G6[KK];
THETA=THETA+G7[KK];
GK1=GK1+G8[KK];
END;

```

```

L50:
GK2=FLOAT(JJ);
SIGMA[1]=SIGMA[1]/GK2;
SIGMA[2]=SIGMA[2]/GK2;

```

- 650-

```

SIGMA[3]*SIGMA[3]/GK2;
S2=S2/GK2;
SIGMA[4]*SIGMA[4]/GK2;
S101=S101/GK2;
S102=S102/GK2;
THETA=THETA/GK2;
GCL=CCL/GK2;
OUT1(A,'27038',1);
OUT08,'7(24-32D.6D),5B-22D.3D,4B-32D.4D';
SIGMA[1]*SIGMA[2]*SIGMA[3]*SIGMA[4]*S2*S101*S102*THETA*GCL;

```

- 660-

```

CRLF(1);
end;
go to 139;
L12: CRLF(1); OUT0(6,'CAN#NOT#CARRY#ON'); CRLF(1);
L39: EJECT;
end;
end;
end; If KEY(2) then go to START;
end;

```

有限要素法による平面問題解析プログラムその3 (ALGOL用) B02991

非線形解析(反復法)で新しい弾性係数を計算する場合応力制御方式から歪制御方式に切り換える事が可能である。

入力データ作成順序

施工研究室2991用

A10, A20, A30, A40, A50,
NP, NE, PS, T, NEP, NFPXY,
FPNXY[1.....NFPXY],
(GAMMA[1.....NEP], EE[1.....NEP],
V[1.....NEP]).....(P, N), ...
X[1.....NP],
Y[1.....NP],
(IE[1.....NE], JE[1.....NE], ME[1.....NE])
DDD, P, DISP[1.....NP2],
LLL, {OW, (P, LOAD[1.....NP2]).....
OW, (P, LOAD[1.....NP2]).....
XYZ,

計算名(5×8文字)

節点数, 要素数等

固定節点番号

弾性パターン情報

(ΣP=NE)

X 座 標

Y 座 標

要素番号と節点番号の関係

強制変位 ΣP=NP2

荷重条件

LLL: くりかえし条件数 ΣP=NP2

切り換え数

プログラム

(BO2991)

begin

comment B02991 NON LINEAR ANALYSIS

This is a program for stress and deformation analysis using Finite Element Method
with vertical and horizontal roller (triangular element)

HOW TO GIVE DATA

A10,A20,A30,A40,A50: Name of calculation (double real)

NP: number of nodal points (integer)

NE: number of elements (integer)

PS: for plane strain, PS=0

for plane stress, PS=1

T: thickness of each element (cm) (real)

NEP: number of elastic constants (integer)

NFPXY: number of fixed points considering x and y direction (integer)

FNPXY[1]: fixed point number considering x and y direction (integer array)

GAMMA[1]: bulk density of each element (g/cm3) (array)

EE[1]: Young's modulus of each element (kg/cm2) (array)

V[1]: Poisson's ratio (array)

P: number of seriesed same data (integer)

N: elastic pattern number (integer)

X[1]: x-coordinate of nodal points (cm) (array)

Y[1]: y-coordinate of nodal points (cm) (array)

IE[1]: nodal point number of I corner in each element (integer array)

JE[1]: nodal point number of J corner in each element (integer array)

ME[1]: nodal point number of M corner in each element (integer array)

DDD: number of repeat for initial displacements data (integer)

DISP[1]: initial displacement of each nodal point in x and y direction (cm) (array)

LLL: number of repeat for loads data (integer)

OW: when you think of own weight, OW=1

when you do not think of it, OW=0

LOAD[1]: loads for each nodal point in x and y direction (Kg) (array)

R1: error limit (real)

AS2: non linear analysis points number (integer)

ORDER TO GIVE DATA

A10, A20, A30, A40, A50, NP, NE, PS, T, NEP,

NFPXY, FNPXY[1...NFPXY],

(GAMMA[1...NEP], EE[1...NEP], V[1...NEP]) ... (P : N) ... (@P=NE),

X[1...NP], Y[1...NP], (IE[1...NEP], JE[1...NEP], ME[1...NEP]) ... ,

DDD, (P : DISP[1...NP2]) ... (@P=NP2),

LLL, OW, (P : LOAD[1...NP2]) ... (@P=NP2),

R1, AS2, (GK1, GK2) ... ,

!

double real A10,A20,A30,A40,A50:

real X1,XJ,XM,Y1,YJ,YM,XIJ,XJM,XMI,YIJ,YJM,YMI,C1,C2,F1,F2,

S,PB,S1,S2,SIG1,SIG2,THETA,T,GK1,GK2,GK3,GK4,C0,

R1,R2,P3,RA,R5,R6,ES1,ES2,EPSIR1,EPSIR2,ETHETA,

NP,NE,NMP,PS,OW,NP2,NP4,I,J,L,N,M,S,I,J,J,M,M,K,A1,A2,B1,B2,

NP2,P1,NK,P,0,1,12,AA1,AA2,BB1,BB2,NMPXY,NMPXY2,NFPXY,NEP,

DD,DDD,LLL,PPP,XYZ,

AS1,AS2,AS3,AS4T

START:

```

HEAD(A10,A20,A30,A40,A50);
READ(NP,NE,PS,T,NEP,NFPXY);
NP2=2*NP; NMPXY=NFPXY; NMPXY2=2*NMPXY;
begin
array GAMMA(EF,V,KE[1:NE],X,Y[1:NP],LOAD,LOADA,DISP,DISPA,RFE[1:NP2],
X[1:NE,1:6],1:6],MATE[1:NP2,1:NP2],MATHC[1:NMPXY,1:NMPXY2],MAC[1:3,1:6],
EP,EP2,EP3[1:NEP],UV[1:6],SIGMA[1:3],
G1,G2,G3,G4,G5,G6,G7,G8,EG1,EG2,EG3,EG4,EG5,EG6,EG7[1:NE],
EPSIN[1:3],Z1,Z2,Z3,Z4[1:20]);
integer array IE,JE,MFE[1:NE],ENP,CNP[1:NP,1:8],MPNXY[1:NMPXY],FNPXY[1:NFPXY];
- 60-
READARRAY(FNPXY,1,NFPXY);
if NFPXY=NP2 v NMPXY=NP2 then go to L500;
for j=1 step 1 until NMPXY do
begin
if FNPXY[j]=1 then MPNXY[j]=1 else go to L100;
end;
L100:
N=J;
for j=2 step 1 until NFPXY do
begin
if FNPXY[j]-FNPXY[j-1];
- 70-
if j=1 then go to L200 else go to L300;
L200: MM=0;
for j=1 step 1 until (N+1)-2 do
begin
if MM=1;
FNPXY[j,j]=FNPXY[j-1]+MM;
end;
N=j;
L300:
- 80-
end;
L400:
MM=0;
for j=N step 1 until NMPXY do
begin
if MM=1;
MPNXY[j,j]=FNPXY[j]+MM;
end;
L500:
for j=1 step 1 until NEP do
- 90-
begin READ(GK1,GK2,GK3);
EP1[j]=GK1; EP2[j]=GK2; EP3[j]=GK3;
end;
P=0;
for j=1 step 1 until NE do
begin READ(P,N);
for j=(1+N) step 1 until (P+8) do
begin GAMMA[j]=EP1[N];
CF[j]=EP2[N];
- 100-

```

- 110-

- 120 -

- 130-

- 140-

- 150-

```

L900:
ppppppp+1: if ppp+1 then go to L900;
AS1=0;
L810:
for i= 1 step 1 until NP2 do
begin
  NISPI:=DISPA[1];
  LOANT:=LOADA[1];
end;
ppp+1: AS1=AS1+1;

```

- 160-

```

for i=1 step 1 until NE do
begin
  I:=I+1;
  J:=J+1;
  M:=M+1;
  X:=X+1;
  XJ:=XJ+1;
  XM:=XM+1;
  Y:=Y+1;
  YJ:=YJ+1;

```

- 170-

```

  YM:=Y+M;
  *E[1]=(-0.001000000+GAMMA[1])*T*((XJ-XI)*(YM-YI)-(XM-XI)*(YJ-YI))/6.000000000;
L1: if p=1 then begin C1:=E[1]/(1.000000000-V[1])*V[1]; C2:=V[1]; end
  else begin C1:=E[1]/(1.000000000-V[1])/((1.000000000-V[1])*(1.000000000-2.000000000+V[1]));
    C2:=V[1]/(1.000000000-V[1]); end;

```

- 180-

```

  F1:=(1.000000000-C2)/2.000000000;
  F2:=(1.000000000-C2)/2.000000000;
  C0:=C1/(C2.000000000*(XJ-XI)*(YM+(XI-XM)*YJ+(XM-XJ)*YI));
  XJ:=XJ-XI;
  XM:=XM-XI;
  XM:=XM-XI;
  YI:=YI-YI;
  YJ:=YJ-YI;
  YM:=YM-YI;
  C[1,1,1]:=(F1*(XJ+YJ+YJ+YJ)+C0;
  C[1,1,2]:=(F1*(XJ+YJ)+C0;
  C[1,1,3]:=(F1*(XJ+YJ)+YJ+YJ)+C0;
  C[1,1,4]:=(F1*(XJ+YI)+C2*(XM+YJ)+C0;
  C[1,1,5]:=(F1*(YI+XJ+YJ)+YJ)+C0;
  C[1,1,6]:=(F1*(XJ+YI)+C2*(XJ+YJ)+C0);

```

- 190-

```

  C[1,2,1]:=(YI+YI);
  C[1,2,2]:=(XJ+XJ+YI+YJ+YJ)+C0;
  C[1,2,3]:=(F1*(XM+YI+YJ+YJ)+C0);
  C[1,2,4]:=(XJ+XJ+YI+YJ+YJ)+C0;
  C[1,2,5]:=(F1*(XJ+YI)+C2*(XJ+YJ)+C0;
  C[1,2,6]:=(XJ+XJ+YI+YJ+YJ)+C0;
  C[1,3,1]:=(YI+YI);
  C[1,3,2]:=(YI+YI);
  C[1,3,3]:=(F1*(XM+YI+YJ+YJ)+C0;
  C[1,3,4]:=(F1*(XJ+YI)+C2*(XJ+YJ)+C0;

```

- 200-

```

K(1,3,5) = (F1 * X1 * X1 + V1 * Y1 * Y1) * C0;
K(1,3,6) = -(F1 * X1 * Y1 + C2 * X1 * Y1) * C0;
K(1,4,1) = L(1,1,4);
K(1,4,2) = L(1,2,4);
K(1,4,3) = L(1,3,4);
K(1,4,4) = (Y1 * X1 + 1 * Y1 * Y1) * C0;
K(1,4,5) = -(F1 * X1 * Y1 + C2 * X1 * Y1) * C0;
K(1,4,6) = (A1 * X1 * F1 + Y1 * Y1) * C0;
K(1,5,1) = L(1,1,5);
K(1,5,2) = L(1,2,5);

- 210-

K(1,5,3) = L(1,3,5);
K(1,5,4) = L(1,4,5);
K(1,5,5) = (F1 * X1 * X1 + V1 * Y1 * Y1) * C0;
K(1,5,6) = -(F2 * X1 * Y1) * C0;
K(1,6,1) = L(1,1,6);
K(1,6,2) = L(1,2,6);
K(1,6,3) = L(1,3,6);
K(1,6,4) = L(1,4,6);
K(1,6,5) = L(1,5,6);
K(1,6,6) = (X1 * X1 + F1 * Y1 * Y1) * C0;

- 220-

end;
LY00;
CHLF(1): CH10(4, '103B, '802991###PAGE#1'); CRLF(1);
OUT0(6, '10P, '###STRESS#AND#DEFORMATION#ANALYSIS#BY#FINITE#ELEMENT#METHOD###, 5B, '({K0, C0#})');
CHLF(2);
OUT0(6, '14P, '##NON#LINEAR#ANALYSIS#');
if (AS1-1)=0 then go to NON;
if (AS1-1)*XY2 then OUT0(6, '47B, 'METHOD#OF#ORYOKU-SEIGYOU');
else OUT0(6, '47B, 'METHOD#OF#HIZUMI-SEIGYOU');
OUT1(6, '2B, 'CYCLE='3ZD', AS1-1);

- 230-

CHLF(1);
NON;
CRLF(3);
OUT0(6, '9B, '###, 8A, 8A, 8A, 8A, A10, A20, A30, A40, A50);
CHLF(6);
OUT0(6, '##INPUT#DATA##');
CHLF(3);
if PS=1 then OUT0(6, '5B, 'PLANE#STRESS') else OUT0(6, '5B, 'PLANE#STRAIN');
OUT1(6, '10H, 'THICKNESS='3ZD, 2D, T);
if Qw=0 then OUT0(6, '#####OWN#WEIGHT');

- 240-

CRLF(3);
OUT0(6, '##NE, 8B, 'GAMMA, 5B, 'YOUNG-M, 4B, 'POISSONS-R, 8B, 'IE, 5B, 'JE, 5B, 'ME');
CRLF(2);
for I=1 step 1 until NE do
begin OUT7(6, '2ZD, 7B, 2D, 3D, 2B7ZD, D, 7BD, 3D, 9B2ZD, 4B2ZD, 4B2ZD,
1, GAMMA[I], EE[I], V[I], IE[I], JE[I], ME[I]);
CRLF(1);
end;
CHLF(4);
OUT0(6, '##NP, 5B, 'ORDER, 9B, 'X0, 9B, 'Y0, 8B, 'LOAD(X), 5B, 'LOAD(Y);

- 250-

```

```

4d.'DISP(X)'.3B.'DISP(Y)')
CHLF(2);
for i=1 step 1 until NP do
begin
  I1#2=I-1 I2#2=I;
  OUT5(4.'2ZD.3#2ZD.B2ZD.2#-6ZD.2D.-6ZD.2D'.1.11.12.X(1).Y(1));
  OUT6(4.'A.2(A-6ZD.2D).3B.2(A-3ZD.3D)'.LOAD(11).LOAD(12).DISP(11).DISP(12));
  CRLF(1);
end;
CHLF(3);

OUT1(6.'NFPXY'.2ZD.'NFPXY'); CRLF(1);
- 260-
GK1#GK2#0.090000000;
for i=1 step 1 until NFPXY do
begin
  I(GK1)=0.01000000000AGK1<0.0100000000 then SPACE(2);
  OUT1(4.'1ZD'.FNPXY(1));
  I(MENTIER(1/10=0.0100000000); JJ=MENTIER(1/20=0.0100000000);
  GK1#1/10=FLOAT(11); GK2#1/20=FLOAT(12);
  I(GK2)=0.01000000000AGK2<0.0100000000 then CRLF(1);
end;

CHLF(1);
- 270-
if NFPXY=NP2 v NMPXY=NP2 then
begin CRLF(3);
  OUT0(4.'****[awoundre if you are a man of sense.***');
  CRLF(1); go to L39;
end;
if PPP=1 then go to L310;
L2: for i=1 step 1 until NP2 do
begin
  for j=1 step 1 until NP2 do
begin
  MAT(1,J)=0.000000000; end;
- 280-
end;
for i=1 step 1 until NP do
begin
  A1#2=I-1;
  A2#2=I;
  for j=1 step 1 until 8 do
begin
  if ENP(1,J)=0 then go to L3 else go to L4;
  KK#ENP(1,J);
- 290-
  AA1#2#ENP(1,J)-1;
  AA2#2#ENP(1,J);
  for l=1 step 1 until NP do
begin
  A3#2=I-1;
  A2#2=I;
  for JJ=1 step 1 until 8 do
begin
  if ENP(1,JJ)=0 then go to L5 else go to L6;
  if ENP(1,JJ)=KK then go to L7 else go to L8;
- 300-

```

```

L7:      BB1=2*CNPE[11,JJ]-1;
          BB2=2*CNPE[11,JJ];
          MAT[A1,B1]=MAT[A1,B1]+K[KK,AA1,BB1];
          MAT[A1,B2]=MAT[A1,B2]+K[KK,AA1,BB2];
          MAT[A2,B1]=MAT[A2,B1]+K[KK,AA2,BB1];
          MAT[A2,B2]=MAT[A2,B2]+K[KK,AA2,BB2];
L8:      end;
L6:      end;
L4:      end;
L910:   end;
if D=0 then go to L99;
for I=1 step 1 until NP do
begin
  A2=2-I;
  for J=1 step 1 until 8 do
begin
  if CNPE[I,J]=0 then go to L9 else go to L10;
  L9:  KK=CNPE[I,J]; LOAD[A2]=LOAD[A2]+WE[KK];
  end;
  L10:
end;
end;
L99:   for I=1 step 1 until NP2 do
begin
  for J=1 step 1 until NP2 do
begin
    LOAD[I]=LOAD[I]-MAT[I,J]*DISPC[J]; end;
  end;
end;
if PPP=1 then go to L920;
for I=1 step 1 until NMPXY do
begin
  for J=1 step 1 until NMPXY do
begin
    I1=NMPXY[I];
    J1=NMPXY[J];
    MAT[I1,J1]=MAT[I1,JJ];
  end;
end;
end;
for I=1 step 1 until NMPXY do
begin
  for J=1 step 1 until NMPXY do
begin
    if I=J then MAT[I1,NMPXY+J]=1.0000000000 else MAT[I1,NMPXY+J]=0.0000000000;
  end;
end;
end;
for NK=1 step 1 until NMPXY do

```

- 310-

- 320-

- 330-

- 340-

- 350-

```

begin
  Q12=NMPXY;
  If MATM[NK,NK]=0.00000000 then go to L11 else P=NK+1;
  If P=NMPXY then go to L12;
L13: If MATM[P,NK]=0.00000000 then go to L14 else P=P+1;
  If P=NMPXY then go to L12 else go to L13;
L14: for J=1 step 1 until Q do
  begin MATM[NK,J]=MATM[NK,J]+MATM[P,J]; end;
L11: RM=MATM[NK,NK];
  for J=1 step 1 until Q do

    begin MATM[NK,J]=MATM[NK,J]/RM; end;
    for I=1 step 1 until NMPXY do
    begin
      RM=MATM[I,NK];
      for J=1 step 1 until Q do
      begin If I=NK then MATM[I,J]=(MATM[I,J]-RM*MATM[NK,J]) else MATM[I,J]=MATM[NK,J]; end;
      end;
      If NKNMPXY then go to L40;
      for I=1 step 1 until NMPXY do
      begin
        for J=1 step 1 until NMPXY do
          begin MATM[I,J]=MATM[I,NMPXY+J]; end;
        end;
      end;
    end;
  end;
L40:
end;
L920:
for I=1 step 1 until NMPXY do
begin
  I=NMPXY+1; S=DISP[I];
  for J=1 step 1 until NMPXY do
  begin
    begin JJ=NPNXY[J]; S=S+MATM[I,J]*LOAD[J];
    end;
    DISP[I]=S;
  end;
  for I=1 step 1 until NMPXY do
  begin I=NPNXY[J]; RF[I]=0.000000000; end;
  for I=1 step 1 until NMPXY do
  begin
    I=NPNXY[J];
    RF[I]=LOAD[I];
    for J=1 step 1 until NMPXY do
    begin JJ=NPNXY[J]; RF[I]=RF[I]+MAT[I,J]*DISP[J]; end;
  end;
  EJECT; CRLF(1);
  OUT(6,'113','PAGE#2'); CRLF(1);
  OUT(6,'**RESULT=1*(#NODAL*POINT*INFORMATION#)**');
  If (AS1-1)=0 then OUT(6,'718','CYCLE=',32D,AS1-1);
  CRLF(3);
  OUT(6,'#NO',AB,'LOAD(X)',BB,'LOAD(Y)',9B,'REAC(X)',9B,'REAC(Y)',
    9B,'DISP(X)',5B,'DISP(Y)',7B,'X',12B,'Y');

```

- 360-

- 370-

- 380-

- 390-

- 400-

```

CRLF(2);
GK1= GK2= GK3= GK4= 0.00000000;
for i=1 step 1 until NP do
begin
  I1=2-1-1 I2=2-1;
  GK4= GK1+ CAD(I1); GK2= GK2+ LOAD(I2);
  GK3= GK3+ RF(I1); GK4= GK4+ RF(I2);
  OUT1(6, '2ZDB', I1);
  OUT4(6, '2(-6ZD, 6D), 2(-7ZD, 6D), 2B, 4(-6ZD, 4D)', LOAD(I1), LOAD(I2), RF(I1), RF(I2),
    DISPL(I1), DISPL(I2), X(I1)+DISPL(I1), Y(I1)+DISPL(I2));
  CRLF(1);
end;
CRLF(1);
OUT0(6, 'GK', 9);
OUT4(6, '2(-64D, 6D), 2(-7ZD, 6D)', GK1, GK2, GK3, GK4); CRLF(1);
EJECT; CRLF(1);
OUT0(6, '1124, 'PAGE#3'); CRLF(1);
OUT0(6, 'RESULT-2*(#ELEMENT#INFORMATION#)');
if (AS1-1)=C then OUT1(6, '75B, 'CYCLE='32D', AS1-1);
CRLF(3);
OUT0(6, 'NE', 9B, 'SIGMA=X', 7B, 'SIGMA=Y', 7B, 'TAU=XY', 8B, 'TAU=MAX',
  7B, 'SIGMA=1', 7B, 'SIGMA=2', 9B, 'THETA', 7B, 'X*(X/Y)');
CRLF(2);
for i=1 step 1 until NE do
begin
  I1=JF(I1);
  JJ=JF(I1);
  MM=JF(I1);
  X1=X(I1);
  XJ=X(JJ);
  XM=X(MM);
  Y1=Y(I1);
  YJ=Y(JJ);
  YM=Y(MM);
  UV(I1)=DISPL(I1-1);
  UV(I2)=DISPL(I2-1);
  UV(I3)=DISPL(I2+JJ-1);
  UV(I4)=DISPL(I2+JJ);
  UV(I5)=DISPL(I2+MM-1);
  UV(I6)=DISPL(I2+MM);
  if PS=1 then begin C1=EE(I1)/(1.000000000-V(I1)*V(I1)); C2=V(I1); end
  else begin C1=EE(I1)/(1.000000000-V(I1))/(1.000000000+V(I1)+(1.000000000-2.000000000*V(I1)));
    C2=V(I1)/(1.000000000-V(I1)); end;
  F1=(1.000000000-C2)/2.000000000;
  CO=C1/(XJ-X1)*YM+(X1-XM)*YJ+(XM-XJ)*Y1;
  MA(I, 1)=(YM-YJ)*CO;
  MA(I, 2)=C2*(XJ-XM)*CO;
  MA(I, 3)=(Y1-YM)*CO;
  MA(I, 4)=(-C2)*(X1-XM)*CO;
  MA(I, 5)=(Y1-YJ)*(-CO);

```

- 410-

- 420-

- 430-

- 440-

- 450-

```

MAL1, A1=C2*(X1-XJ)*C01
MAL2, A1=C2*(Y1-YJ)*C01
MAL2, A1=C2*(X1-XJ)*C01
MAL2, A1=C2*(Y1-YJ)*C01
MAL2, A1=C2*(X1-XJ)*C01
MAL2, A1=C2*(Y1-YJ)*C01
MAL2, A1=C2*(X1-XJ)*C01
MAL2, A1=C2*(Y1-YJ)*C01
MAL3, A1=C2*(X1-XJ)*C01
MAL3, A1=C2*(Y1-YJ)*C01
MAL3, A1=C2*(X1-XJ)*C01
MAL3, A1=C2*(Y1-YJ)*C01

```

- 460 -

```

MAL3, A1=C2*(Y1-YJ)*C01
MAL3, A1=C2*(X1-XJ)*C01
MAL3, A1=C2*(Y1-YJ)*C01
FOR J=1 STEP 1 UNTIL 3 DO
  BEGIN
    SUM=0.0000000000
    FOR L=1 STEP 1 UNTIL 6 DO
      BEGIN SUM=SUM+AL(J,L)*UV(L); END;
    SIGMAL(J)=SUM;
  END;

```

- 470 -

```

S1=(SIGMA[1]+SIGMA[2])/2.000000000;
S2=500*((SIGMA[1]-SIGMA[2])/2.000000000)*((SIGMA[1]-SIGMA[2])/2.000000000)+SIGMA[3]*SIGMA[3];
SIG1=S1-S2;
SIG2=S1+S2;
THETA=2.64788975654*(ARCTAN(2.000000000*SIGMA[3]/(SIGMA[1]-SIGMA[2])));
IF (SIGMA[1]-SIGMA[2])>0.000000000 THEN THETA=THETA+90.00000000;

```

L20:

```

IF SIGMA[2]<0.000000000 THEN GK1=9999.9999 ELSE GK1=SIGMA[1]/SIGMA[2];
OUT1(4,'22038',1);
OUT1(4,'472B-32D.4D',5B-22D.3D,4B-32D.4D',SIGMA[1],SIGMA[2],SIGMA[3],S2,SIG1,SIG2,THETA,GK1);

```

- 480 -

```

CRLF(1);
G1[1]=SIGMA[1]; G2[1]=SIGMA[2];
G3[1]=SIGMA[3]; G4[1]=S2;
G5[1]=SIG1; G6[1]=SIG2;
G7[1]=THETA; G8[1]=GK1;
GK1=GK2+0.01; GK1=1/(C1*(1-C2*C2)); GK2=1/(C1*F1);
EPSIR[1]=EPSIR[2]*EPSIR[3]/0.01;
EPSIR[1]=GK1*(G1[1]-C2*G2[1]);
EPSIR[2]=GK1*(G2[1]-C2*G1[1]);
EPSIR[3]=GK2*(G3[1]);

```

- 490 -

```

ES1=(EPSIR[1]+EPSIR[2])/2;
FS2=500*((EPSIR[1]-EPSIR[2])/2+EPSIR[3]*2);
EPSIR1=ES1+ES2;
EPSIR2=ES1-ES2;
IF EPSIR[2]>EPSIR[1] THEN GO TO EHHH1;
ETHETA=2.64788975654*(ARCTAN(2.0*EPSIR[3]/(EPSIR[1]-EPSIR[2])));
IF EPSIR[1]>EPSIR[2] THEN ETHETA=ETHETA+90.001 GO TO EHH1;
IF EPSIR[1]>EPSIR[2] ^ EPSIR[3]>0 THEN GO TO EHH1;
IF EPSIR[1]>EPSIR[2] ^ EPSIR[3]<0 THEN ETHETA=ETHETA+180.001 GO TO EHH1;
EHHH1:

```

- 500 -

```

if EPSIR[1]=EPSIR[2] ^ EPSIR[3]>0 then ETHETA=45.00; go to EHH1;
if EPSIR[1]=EPSIR[2] ^ EPSIR[3]<0 then ETHETA=135.00; go to EHH1;
if EPSIR[1]=EPSIR[2] ^ EPSIR[3]=0 then ETHETA=360.00; go to EHH1;
EHH1:
EG1[1]=EPSIR[1]; EG2[1]=EPSIR[2];
EG3[1]=EPSIR[3]; EG4[1]=ES2;
EG5[1]=EPSIR1; EG6[1]=EPSIR2;
EG7[1]=ETHETA;
end;
EJECT: CRLF(1);

```

- 510-

```

OUTO(6,'113R','PAGE#4'); CRLF(1);
OUTO(6,'**RESULT-3#(ELEMENT#INFORMATION--EPUSIRON#)#*');
if (AS1-1)=0 then OUT1(6,'65B','CYCLE=',32D',AS1-1);
CRLF(3);
OUTO(6,'NE',9B,'HIZUM=X',7B,'HIZUM=Y',7B,'SEN=XY',8B,'SEN=MAX');
OUTO(6,'7B','HIZUM=1',7B,'HIZUM=2',9B,'THETA');
CRLF(2);
for I=1 step 1 until NE do
begin
OUT1(6,'27D3B',I);

```

- 520-

```

OUT7(A,'6(7B=37D.6D),5B=27D.3D',EG1[1],EG2[1],EG3[1],2*EG4[1],EG5[1],EG6[1],EG7[1]);
CRLF(1);
end;
EJECT: CRLF(1);
OUTO(6,'113R','PAGE#5'); CRLF(1);
OUTO(6,'**RESULT-4#(MEAN#VALUE#AROUND#NODAL#POINT#)#*');
if (AS1-1)=0 then OUT1(6,'65B','CYCLE=',32D',AS1-1);
CRLF(3);
OUTO(6,'#NP',9B,'SIGMA=X',7B,'SIGMA=Y',7B,'TAU=XY',8B,'TAU=MAX',7B,'SIGMA=1',
7B,'SIGMA=2',9B,'THETA',7B,'K#(X/Y)');

```

- 530-

```

CRLF(2);
for I=1 step 1 until NP do
begin
SIGMA[1]=SIGMA[2]=SIGMA[3]=S2=SIG1=SIG2=THETA=GK1=0.000000000;
JJ=0;
for J=1 step 1 until 8 do
begin
KK=ENP[I,J];
if KK=0 then go to L50;
JJ=JJ+1;

SIGMA[1]=SIGMA[1]+G1[KK];
SIGMA[2]=SIGMA[2]+G2[KK];
SIGMA[3]=SIGMA[3]+G3[KK];
S2=S2+G4[KK];
SIG1=SIG1+G5[KK];
SIG2=SIG2+G6[KK];
THETA=THETA+G7[KK];
GK1=GK1+G8[KK];
end;
L50:

```

- 540-

- 550-

```

GK2=F(QAT(JJ))
SIGMAF1=SIGMA[1]/GK2
SIGMAF2=SIGMA[2]/GK2
SIGMAF3=SIGMA[3]/GK2
S2=S2/GK2
SIG1=SIG1/GK2
SIG2=SIG2/GK2
THETA=THETA/GK2
GK1=GK1/GK2
OUT1(4,'22038',1)

OUT1(4,'4(28-32D,40),58-22D,35,48-32D,4D',
      SIGMAF1,SIGMAF2,SIGMAF3,S2,
      SIG1,SIG2,THETA,GK1)
CRLF(1)
und:
EJECT: CRLF(1)
OUTO(6,'114N','PAGE#6'): CRLF(1)
OUTO(6,'**RESULTS-5*(#MEAN*VALUE#AROUND#NODAL#POINT--EPSIRON#)#**'):
IF (AS1-1) < 0 THEN OUT1(4,'538','CYCLE='32D',AS1-1)
CRLF(3)

OUTO(6,'**NP',98,'HIZUM=X',78,'HIZUM=Y',78,'SEN=XY',88,'SEN=MAX'):
OUTO(6,'78','HIZUM=1',78,'HIZUM=2',98,'THETA'):
CRLF(2)
for [n] step 1 until NP do
begin
EPSIR[1]=EPSIR[2]*EPSIR[3]*ES2*EPSIR1*EPSIR2*ETHETA=0.0
JJ=0
for J=1 step 1 until 8 do
begin
KK=FNPF1(J)
if KK=0 then go to LL50
JJ=JJ+1
EPSIR[1]=EPSIR[1]*EG1[KK]
EPSIR[2]=EPSIR[2]*EG2[KK]
EPSIR[3]=EPSIR[3]*EG3[KK]
ES2=ES2*EG4[KK]
EPSIR[1]=EPSIR[1]*EK5[KK]
EPSIR[2]=EPSIR[2]*EG6[KK]
ETHETA=ETHETA*EG7[KK]
end:
LL50:
GK2=F(QAT(JJ))
GK2=1/GK2
EPSIR[1]=EPSIR[1]*GK2
EPSIR[2]=EPSIR[2]*GK2
EPSIR[3]=EPSIR[3]*GK2
ES2=ES2*GK2
ETHETA=ETHETA*GK2
EPSIR[1]=EPSIR[1]*GK2
EPSIR[2]=EPSIR[2]*GK2

```

- 560-

- 570-

- 580-

- 590-

- 600-

```

OUT1(6,'2D3R',1);
OUT7(6,'(2d-3D.4D).5B-2D.3D';
      EPS1(11)=EPSIN(2);EPS1(13)=2*ES2;
      EPS1(1)=EPS1(2)*ETHETA);
CRLF(1);
end;
go to L79;
L12: CRLF(5); OUT0(6,'CAN*NOT*CARRY*ON'); CRLF(1);
L39: EJECT;
      If AS1=1 then go to L820;
- 610-
CRLF(3);
OUT0(6,'10R.'*BEGINING*OF*NON*LINEAR*ANALYSIS*'2/');
READ(R1,AS2);
OUT2(6,'N1='*2D.4D.5B.*N='*2D.9B.*STRESS'*9B.*STRAIN'/'*R1,AS2);
for I=1 step 1 until AS2 do
begin
      READ(GK1,GK2);
      Z1[I]=GK1;
      Z2[I]=GK2;
      OUT2(6,'23b.2(-62D.6D)/*Z1[I],Z2[I]);
- 620-
end;
HEAD(XYZ);
EJECT;
L820:
      If AS1=1 then go to L830;
      If AD$DISP(AS4)-R4>R1 then go to L850;
      R4=DISP(AS4);
      go to L840;
L830:
      R2=0.00000000;
- 630-
for I=1 step 1 until NP2 do
begin
      R3=ANS(DISP(I));
      If R3>R2 then begin R4=R3; AS4=1; R2=R3; end;
end;
L840:
      If (AS1-1)<XYZ then go to XY1 else go to XY2;
      XY1:
      for I=1 step 1 until NE do
begin
- 640-
          for J=1 step 1 until (AS2-1) do
begin
              If G2[I]>Z1[J] & G2[I]<Z1[J+1] then
begin
                  R5=(Z2[J+1]-Z2[J])/(Z1[J+1]-Z1[J]);
                  R6=R3+G2[I]+(R5*(-Z1[J])+Z2[J]);
              end;
              end;
              EE[I]=G2[I]/R6;
end;
- 650-

```

```

go to XY3;
XY2:
for I=1 step 1 until NE do
begin
  for J= 1 step 1 until (AS2-1) do
  begin
    if EG2[I]>Z2[J] ^ EG2[I]<Z2[J+1] then
    begin
      R5=(Z1[J+1]-Z1[J])/(Z2[J+1]-Z2[J]);
      P6=R5*EG2[I]+(R5*(-Z2[J])+Z1[J]);

      end;
    end;
    EF[I]=R6/EG2[I];
  end;
XY3:
if AS1<100 then go to L810;
L850:
CRLF(3); OUTD(6,'10B','OWARIMASHITAYO'); EJECT;
end;
end;
end; if KEY(8) then go to START;
end;

```

- 660-

- 670-