674
12-12-79
LA-7963

LG. 402

Thermal Fracture Effects in Geothermal Energy Extraction

MASTER

University of California

LOS ALAMOS SCIENTIFIC LABORATORY

Post Office Box 1663 Los Alamos, New Mexico 87545

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Work supported by the US Department of Energy,
Division of Geothermal Energy.

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

LA-7963

UC-66a

Issued: November 1979

Thermal Fracture Effects in Geothermal Energy Extraction

Ruth B. Demuth
Francis H. Harlow

DISCLAIMER

This book was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

fly

CONTENTS

ABSTRACT	1
I. INTRODUCTION	1
II. THE PHYSICAL PROBLEM AND ITS IDEALIZATIONS	2
III. THE DIFFERENTIAL EQUATIONS	3
IV. THE SOLUTION TECHNIQUE	5
V. BOUNDARY CONDITIONS	18
VI. NUMERICAL STABILITY	20
VII. MODIFIED VERSIONS	21
VIII. CALCULATIONAL EXAMPLES	23
IX. THE COMPUTER PROGRAM	37
A. General Structure	37
B. The Indexing Notation	41
C. The Input and Derived Data	43
D. The Common Blocks, VAR and ARRAYS	45
E. Output - Prints, Plots	46
F. Operating the Terminal	47
APPENDIX A. CODE LISTING FOR GEØ-I	48
APPENDIX B. CODE LISTING FOR GEØ-II	76
APPENDIX C. CODE LISTING FOR GEØ-III	102
APPENDIX D. OUTPUT EXAMPLES	130
I. GEØ-I	131
II. GEØ-II	135
III. GEØ-III	139
ACKNOWLEDGMENTS	148
BIBLIOGRAPHY	148

THERMAL FRACTURE EFFECTS IN GEOTHERMAL ENERGY EXTRACTION

by

Ruth B. Demuth and Francis H. Harlow

ABSTRACT

Heat and water exchange between the primary crack and a lateral thermal fracture are analyzed by means of numerical modeling on a high-speed computer. Three-dimensional heat flow in the rock is coupled to a two-dimensional approximation for the time-varying convective patterns that result from both forced flow and buoyancy. The physical and mathematical bases for the study are described together with the numerical technique for solution. The report includes listings of three versions of the computer code and some results of representative calculations.

I. INTRODUCTION

The extraction of geothermal energy from hot dry rock requires drilling to a region of competent material that has sufficiently great temperature for the efficient extraction of heat. The rock must then be fractured, a process accomplished by means of elevated hydraulic pressure, which under idealized circumstances produces a vertically oriented circular crack that is hundreds of meters in radius but only a few millimeters in width. Cold water is then pumped into the lower part of the crack and hot water is withdrawn from the upper part in a nearly closed circuit that delivers its energy at the surface by means of a heat exchanger.

Many considerations, both practical and theoretical, contribute to departure from this idealized picture (see Bibliography); some are detrimental but some are highly advantageous. In this latter category is the expected tendency for thermal stress in the cooling rock to result in additional fractures.

These thermal cracks tend also to be vertically oriented and to extend in a direction normal to the surfaces of the primary fracture. Their presence enlarges significantly the volume of rock accessible to heat extraction by the circulating fluid.

Our goal here is to model mathematically the exchange of water and heat between the primary fracture and a single growing lateral thermal fracture. We use the differential equations for fluid flow and heat transfer in the formulation of finite difference equations that are solved by means of a high-speed computer. The resulting computer code, described in this report in three versions, is called GEO.

II. THE PHYSICAL PROBLEM AND ITS IDEALIZATIONS

In reality, the primary crack is likely to have a tortuous configuration determined by the initial distribution of relatively weak regions in the rock. Its width may vary, not just from center to edge, but over relatively short distances, and it may contain fragments of rock that have broken loose and been transported by gravity or fluid forces.

Thermal fractures branching from this primary crack may likewise be distorted from the idealized configuration of vertical orientation and smoothly varying width.

Nevertheless, in this phase of the study, we assume that the part of the primary crack resolved by the calculation has smooth, parallel sides, vertical in orientation, composed of rock that is homogeneous in all properties except the distribution of temperature. From one side, there grows a single thermal fracture with smooth vertical walls and a width that varies with position and time.

Within the two intersecting cracks, there is a complicated flow pattern that varies in three space dimensions and time. Even before the thermal crack starts opening, the water flow pattern is influenced by variations in buoyancy across the gap, by the wall drag, and possibly by the effects of turbulence. When the lateral fracture opens as a result of thermal stress effects, the potentiality for complex flow structure increases. Near the intersection of the two cracks, the flow streamlines may be distorted by swirling and the formation of local shears. Within the fracture, strong gradients in water viscosity and buoyancy result from the higher temperature of the rock that is being opened,

generating a likelihood for secondary recirculating flow patterns that can significantly affect the rate of heat transfer to the primary crack.

The limitations of computer time, however, preclude a detailed resolution of these complex flow configurations, forcing us to idealize the dynamics in several important respects. In particular, the flow field is constrained by our coordinate mesh to be purely two-dimensional.

In the primary crack, the flow is resolved in the vertical direction and in the horizontal direction across the width of the gap. In the lateral crack, resolution extends in two dimensions parallel to the bounding surfaces of the fracture but not in the direction across the gap.

An alternative to this resolution would employ two intersecting coordinate systems, one in each crack, each in the plane of its crack, thereby orthogonal to each other where they meet.

Neither layout is capable of showing the three-dimensional flow-field contortions near the intersection. The importance of these contortions must remain for now a matter for speculation. Each layout has advantages and disadvantages as a two-dimensional compromise. Our choice for this present stage of the work is based on the conjecture that the presence of a thermal fracture has little effect on the flow field in the primary crack on either side of the intersection region.

Both coordinate layouts, however, emphasize what we believe to be the most significant aspects of the fluid flow, namely the complex circulation patterns expected in the lateral fracture as a result of various competing mechanisms. Principal among the driving influences are the effects of forced convection in the primary crack and the communication of stress by flow penetration and viscous drag on the fluid in the lateral crack. Even more important is the natural convection induced by buoyancy. None of these is lost by the constraints of our analysis.

III. THE DIFFERENTIAL EQUATIONS

We use the Boussinesq approximation for a fluid with density that varies by a small fraction of its nominal value, ρ_0 , as a result of thermal expansion. Compared to sound speed, the fluid velocity is small enough for the material to be considered otherwise incompressible. Accordingly, the conservation of mass in three space dimensions can be described by vanishing divergence of the fluid velocity, $\vec{u}$, so that

$$\nabla \cdot \vec{u} = 0 \quad (1)$$

in the absence of volumetric sources and sinks. For the momentum equation, the Boussinesq form is

$$\frac{\partial \vec{u}}{\partial t} + \nabla \cdot (\vec{u}\vec{u}) = -\nabla \bar{\phi} + (\nabla \cdot \nu \nabla) \vec{u} + \vec{g} \frac{\rho}{\rho_0} \quad (2)$$

in which $\bar{\phi}$ is the ratio of pressure to nominal density, ν is the kinematic viscosity coefficient, and $\vec{g}$ is the acceleration of gravity. Incorporating Eq. (1), we transformed the usual convective term, $\vec{u} \cdot \nabla \vec{u}$, into the divergence of a dyadic in order to produce a conservative form as a basis for the finite difference equations. The ratio of actual to nominal density, ρ/ρ_0 , can be described in terms of water temperature, T , a reference temperature T_0 , and volumetric coefficient of expansion, β , by the relation

$$\frac{\rho}{\rho_0} = 1 - \beta(T - T_0) \quad (3)$$

With T in degrees Celsius and ν in units of cm^2/sec , it is sufficiently accurate for our purposes to write

$$\nu = \frac{0.279}{T - 3.8} \quad .$$

The variations in water temperature are described by

$$\frac{\partial T}{\partial t} + \nabla \cdot (\vec{u}T) = \nabla \cdot \kappa_W \nabla T \quad (4)$$

in which κ_W is the thermometric conductivity (molecular plus turbulent) of the water. Within the rock, the equation for rock temperature, ξ , is

$$\frac{\partial \xi}{\partial t} = \nabla \cdot \kappa_R \nabla \xi \quad , \quad (5)$$

with κ_R denoting the thermometric conductivity in the rock. In each case, the thermometric conductivity is the ratio of heat conduction coefficient to the product of density and specific heat coefficient. We shall also need a hybrid thermometric conductivity, κ_0 , defined as the ratio of heat conduction coefficient in rock to the product of density and specific heat coefficient in the water.

IV. THE SOLUTION TECHNIQUE

The ideal procedure would be to solve the equations for time variations throughout the entire three-dimensional region of interest. Instead, we introduce at this stage the two-dimensional approximations that are required for the numerical solution, as described above. The coordinate system, as shown in Fig. 1, consists of the x axis in the horizontal direction, with origin at the face of the primary crack opposite its intersection with the thermal fracture, extending across the primary crack, into the thermal fracture along its mid-plane, and into the unfractured rock. The y axis extends in a vertical direction, with its origin at or below the bottom of the thermal fracture. The z axis is orthogonal to the other two, thus extending in a horizontal direction normal to the plane of the thermal fracture.

We now describe the code version, GEO-I, in which there is resolution of the dynamics and temperature variations in the primary crack. As a function of x, y, and t, we define W as the crack width. Within the thermal fracture, W is the true crack width, whereas in the primary crack W denotes a distance that is roughly proportional to the distance in the z direction to the neighboring thermal fracture. Because of the relatively poorly known nature of this inter-crack spacing, we perform the transformation to two space dimensions by the same method of integration across both the primary and lateral cracks. W_p , the value of W in the primary crack, must accordingly be considered adjustable in its interpretation, as described below.

The region of interest is divided into finite difference computational cells. Within the cracks, these cells are of width δx and δy in the plane of resolution, and extend a distance $W/2$ in both the positive and negative z directions away from the plane of symmetry through the center of the thermal fracture. Figure 2 shows the locations in the x - y plane to which our subscript notation will refer. In particular, cell centers are labeled with integers i and j , with i increasing along the x axis from 2 in the leftmost cell to i_1 in the last cell of the primary crack, thence to $\bar{i}+1$ at the far right. Likewise, j increases in the positive y direction from 2 in the bottom row of cells to $\bar{j}+1$ at the top. Thus the cell-number dimensions of the finite difference mesh resolving the system are $\bar{i}$ by $\bar{j}$.

The x and y components of water velocity, $u_{i+\frac{1}{2}j}$ and $v_{ij+\frac{1}{2}}$, refer to appropriate spatial averages around the cell-edge positions described by the subscripts. Likewise, $\bar{\phi}_{ij}$, W_{ij} and T_{ij} refer to cell-centered quantities.

Within the rock, the finite difference cells are three-dimensional. With W in the thermal fracture always much less than W_p , we measure distance in the rock from a face of the thermal fracture as though that face were at $z = 0$. The rock cells also have dimensions δx and δy in the x - y plane, with positions and subscript notations coincident with those for the crack cells. In the z direction, the cell dimension is δz . As shown in Fig. 3, the integer subscript k , extending from 1 to $\bar{k}$, counts positions that are cell-centered in the x - y plane, but face-centered in the z direction, with the face at $k = 1$ lying on the face of the thermal fracture ($z = 0$) and $k = \bar{k}$ lying at $z = W_p/2$, so that $W_p = 2(\bar{k} - 1)\delta z$. Thus, with ξ_{ijk} representing an appropriate average of rock temperature around the position designated by the subscripts, it follows that ξ_{ij1} is the rock temperature at the face of the thermal crack in cell i, j , provided $i > i_1$. Likewise, $\xi_{i_1+\frac{1}{2}jk}$ denotes rock temperature at the face of the primary crack away from the thermal fracture. In both cases, our boundary conditions equate these rock temperatures at the faces of the cracks to the adjacent water temperature. Since half-integer i values for rock temperature do not correspond to positions for which rock temperature is stored in the computer, the concept of $\xi_{i_1+\frac{1}{2}jk}$ requires further clarification. Usually the value of any variable for which the subscripts indicate a position away from the computer storage position is formed of appropriate averages of the surrounding values, but a boundary value like that of $\xi_{i_1+\frac{1}{2}jk}$ requires an extrapolative

Fig. 1. Side view in the region of a single thermal crack.

Fig. 2. Subscript notation for positions relative to a computational cell in the x-y plane.

Fig. 3. Subscript notation for positions relative to computational cells in one corner of the x-z plane.

procedure, described below. Also, cell-edge and cell-corner values of W require special consideration.

To derive the finite difference equations for heat and mass transfer in the cracks, we integrate the equations over the three-dimensional volume contained in a cell of dimensions δx by δy by W . In this process, we assume that the velocity components vary with z in a Poiseuille profile. Relatively greater buoyancy near the edges of the crack, together with the possible turbulence and the effects of wall irregularity, would induce departures from this idealization, but because of our lack of precise information about these effects we write

$$u_{i+\frac{1}{2}j} \left[1 - \left(\frac{2z}{W_{i+\frac{1}{2}j}} \right)^2 \right]$$

and

$$v_{ij+\frac{1}{2}} \left[1 - \left(\frac{2z}{W_{ij+\frac{1}{2}}} \right)^2 \right]$$

for the two components of velocity in the localities designated by the subscripts. Allowing for W to vary with time, we can arrange the integral of Eq. (1) to the form

$$D_{ij} \equiv \frac{\partial W_{ij}}{\partial t} + \frac{2}{3} \left[\frac{(Wu)_{i+\frac{1}{2}j} - (Wu)_{i-\frac{1}{2}j}}{\delta x} + \frac{(Wv)_{ij+\frac{1}{2}} - (Wv)_{ij-\frac{1}{2}}}{\delta y} \right] = 0 . \quad (6)$$

Similar but lengthy calculations can be carried out for the two components of the momentum equation, Eq. (2). In the process, we equate the x and y components of $\vec{g}$ to the values 0 and $-g$, respectively. In addition we define rescaled pressure-to-density ratios, ϕ , such that

$$\phi_{i+1j} - \phi_{ij} \equiv \bar{\phi}_{i+1j} - \bar{\phi}_{ij}$$

and

$$\phi_{ij+1} - \phi_{ij} = \bar{\phi}_{ij+1} - \bar{\phi}_{ij} + g\delta y[1 - \beta(\bar{T}_{j+\frac{1}{2}} - T_0)] \quad , \quad (7)$$

where $\bar{T}_{j+\frac{1}{2}}$ is a reference temperature chosen to equal the mean temperature at that level across the primary crack. This choice of reference temperature for the buoyancy force is made in order to increase the efficiency of our iterative solution procedure, as discussed by Harlow and Pracht (see Bibliography). This transformation is the origin of our modified pressure, as presented in the output of the calculations.

In the resulting momentum equations, we use donor-cell notation for the convective fluxes. Thus, for any quantities A and B,

$$\langle Au:B \rangle_{ij} \equiv (Au)_{ij} \cdot \begin{cases} B_{i-\frac{1}{2}j} & \text{if } u_{ij} > 0 \\ B_{i+\frac{1}{2}j} & \text{if } u_{ij} < 0 \end{cases} \quad . \quad (8)$$

In general, the part before the colon is written with the subscripts of the whole term. The part following the colon bears the subscripts denoting a half cell behind the direction of the velocity that occurs in the part before the colon. Except as noted, wherever a quantity like u_{ij} is indexed in reference to a position other than its primary locality, the nomenclature denotes an average of the values at the adjacent true localities.

With these definitions, the integrals of the momentum equation components become

$$\begin{aligned}
\frac{\partial(uW)_{i+\frac{1}{2}j}}{\partial t} = & \frac{8}{15} \left[\frac{\langle Wu:u \rangle_{ij} - \langle Wu:u \rangle_{i+1j}}{\delta x} \right. \\
& + \left. \frac{\langle Wv:u \rangle_{i+\frac{1}{2}j-\frac{1}{2}} - \langle Wv:u \rangle_{i+\frac{1}{2}j+\frac{1}{2}}}{\delta y} \right] \\
& + W_{i+\frac{1}{2}j} \frac{\phi_{ij} - \phi_{i+1j}}{\delta x} - \frac{8v_{i+\frac{1}{2}j}}{W_{i+\frac{1}{2}j}} u_{i+\frac{1}{2}j} \\
& + (\nabla_2 \cdot v_a W \nabla_2 u)_{i+\frac{1}{2}j} , \tag{9}
\end{aligned}$$

and

$$\begin{aligned}
\frac{\partial(vW)_{ij+\frac{1}{2}}}{\partial t} = & \frac{8}{15} \left[\frac{\langle Wv:v \rangle_{ij} - \langle Wv:v \rangle_{ij+1}}{\delta y} \right. \\
& + \left. \frac{\langle Wu:v \rangle_{i-\frac{1}{2}j+\frac{1}{2}} - \langle Wu:v \rangle_{i+\frac{1}{2}j+\frac{1}{2}}}{\delta x} \right] \\
& + W_{ij+\frac{1}{2}} \frac{\phi_{ij} - \phi_{ij+1}}{\delta y} - \frac{8v_{ij+\frac{1}{2}}}{W_{ij+\frac{1}{2}}} v_{ij+\frac{1}{2}} \\
& + W_{ij+\frac{1}{2}} \beta g(T_{ij+\frac{1}{2}} - \bar{T}_{j+\frac{1}{2}}) + (\nabla_2 \cdot v_a W \nabla_2 v)_{ij+\frac{1}{2}} . \tag{10}
\end{aligned}$$

The two-dimensional finite difference gradient operator, ∇_2 , is explicitly defined in the terms for $\overline{Wu}$ and $\overline{Wv}$, below, where the distinction between v and v_a is also discussed.

The next step is to perform this same integration for the water temperature equation, Eq. (4). Within the thermal fracture, that is for $i > i_1$, the integration leads to a requirement for the temperature gradient at the walls of the crack. To accomplish this specification, we invoke the principle of heat flux continuity, which leads to the requirement for a hybrid thermometric conductivity, κ_0 , defined above. Thus

$$\kappa_W \left(\frac{\partial T}{\partial z} \right)_{\text{edge}} = \kappa_0 \left(\frac{\partial \xi}{\partial z} \right)_z = 0$$

The result, after some algebraic manipulation and the introduction of donor-cell convection, can be written

$$\begin{aligned} \frac{\partial (WT)_{ij}}{\partial t} = & \frac{2}{3} \left[\frac{\langle Wu:T \rangle_{i-\frac{1}{2}j} - \langle Wu:T \rangle_{i+\frac{1}{2}j}}{\delta x} \right. \\ & \left. + \frac{\langle Wv:T \rangle_{ij-\frac{1}{2}} - \langle Wv:T \rangle_{ij+\frac{1}{2}}}{\delta y} \right] \\ & + T_{ij} D_{ij} + 2\kappa_0 \left(\frac{\partial \xi_{ijk}}{\partial z} \right)_{z=0} + (\nabla_2 \cdot \kappa_a \nabla_2 T)_{ij} \end{aligned} \quad (11)$$

The last term, in analogy to similar terms in Eqs. (9) and (10), describes heat mixing from molecular diffusion or else either as a crude representation of turbulence or simply as an artifice to enhance numerical stability. The factor 2 in the wall-heat term describes the presence of two walls in the thermal crack. The addition of the term $T_{ij} D_{ij}$, which vanishes according to Eq. (6), ensures that our iterative solution technique, which never precisely accomplishes the vanishing of D_{ij} , will not result in spurious sources for the temperature equation. To see that the extra term is effective for this purpose, note that for circumstances in which the water temperature is constant in space, T can be factored from the donor-cell terms in Eq. (6) and the D_{ij} term cancels with much of the residuum.

For $2 < i < i_1$, the wall-heat term is omitted from Eq. (11). For $i = 2$ and $i = i_1$, that term is replaced by the effect of heat flux from the rock into the primary crack, as discussed below.

Derivation of the finite difference equation for the rock temperature requires integration over a fixed rectangular parallelepiped cell, and is accordingly much simpler than the previous derivations. In the result, we introduce discrete time steps of duration δt , through which the initial-value problem develops. These steps are numbered by superscripts $n+1$ and n , denoting the

new and old values, respectively, in a computational cycle. Wherever this cycle count superscript is omitted, the n^{th} time step value is implied. With this nomenclature, the equation for rock temperature, Eq. (5), becomes

$$\begin{aligned} \xi_{ijk}^{n+1} = & \xi_{ijk}^n + \kappa_R \delta t \left[\frac{1}{\delta x^2} (\xi_{i+1jk} + \xi_{i-1jk} - 2\xi_{ijk}) \right. \\ & \left. + \frac{1}{\delta y^2} (\xi_{ij+1k} + \xi_{ij-1k} - 2\xi_{ijk}) + \frac{1}{\delta z^2} (\xi_{ijk+1} + \xi_{ijk-1} - 2\xi_{ijk}) \right] . \end{aligned} \quad (12)$$

With the calculation of ξ_{ijk} specified, we can now show how the water temperature source terms and variations in W are to be calculated. Thus, for Eq. (11) we can put

$$2\kappa_0 \left(\frac{\partial \xi_{ijk}}{\partial z} \right)_{z=0} = 2\kappa_0 \frac{\xi_{ij2}^{n+1} - T_{ij}^{n+1}}{\delta z} . \quad (13)$$

For $i = 2$ and $i = i_1$, this term in Eq. (11) is replaced by

$$\begin{aligned} \frac{\kappa_0 \delta z}{4\delta x^2} \sum_{k=1}^{\bar{k}-1} \left(\xi_{i_1+1jk}^{n+1} + \xi_{i_1+1jk+1}^{n+1} + \xi_{i_1+2jk}^{n+1} + \xi_{i_1+2jk+1}^{n+1} \right) \\ - \frac{\kappa_0^W}{\delta x^2} T_{i_1j}^{n+1} \end{aligned} \quad (14)$$

which sums the fluxes into cell i_1, j from all the cells in the z direction lying on its rock face. Setting this same source into the cell $i = 2$ does not imply symmetry across the midplane of the primary crack, except insofar as it indicates that a similar set of thermal fractures lies on both sides, but not with symmetric orientation.

For the calculation of W , we hold $W_{ij} = W_p$ constant in space and time for $i < i_1$. For $i > i_1$, we allow for all the thermal contraction along a z direction line of depth $W_p/2$ in each direction from the thermal fracture to determine the width of the crack. Thus

$$W_{ij}^{n+1} = -2\beta_R \delta z \left[\frac{1}{2} (\xi_{ij1}^{n+1} - \xi_\infty) + \sum_{k=2}^{\bar{k}} (\xi_{ijk}^{n+1} - \xi_\infty) \right], \quad (15)$$

in which β_R is the coefficient of linear expansion for rock, and ξ_∞ is the rock temperature far from the cracks. If, however, Eq. (15) results in $W_{ij} < W_0$, where W_0 is a specified threshold value, we set $W_{ij}^{n+1} = 0$.

In proceeding, it is convenient to give a shorthand name to those parts of the momentum equations that require only n -time quantities for their calculation. These defined quantities exhibit the precise form of the ∇_2 terms in Eqs. (9) and (10), in which v has been replaced by v_a , a constant coefficient that may either approximate the true viscosity of the fluid or be used as "artificial" viscosity to enhance the stability of the numerical procedure if such should be required. For those examples calculated to date, there was no evidence that this coefficient was required for stability, so we simply used the nominal value for water, with negligible effect on the results. Note that in the wall-drag terms of Eqs. (9) and (10), v continues without subscript to denote the true water viscosity, which varies with temperature.

With these changes, we define the quantities

$$\begin{aligned} (\overline{uW})_{i+\frac{1}{2}j} = & (uW)_{i+\frac{1}{2}j}^n + \delta t \left\{ \frac{8}{15} \left[\frac{1}{\delta x} (< Wu:u >_{ij} - < Wu:u >_{i+1j}) \right. \right. \\ & + \frac{1}{\delta y} (< W'v:u >_{i+\frac{1}{2}j-\frac{1}{2}} - < W'v:u >_{i+\frac{1}{2}j+\frac{1}{2}}) \Big] \\ & + \frac{v_a}{\delta x^2} \left[W_{i+1j} (u_{i+3/2j} - u_{i+\frac{1}{2}j}) - W_{ij} (u_{i+\frac{1}{2}j} - u_{i-\frac{1}{2}j}) \right] \\ & \left. + \frac{v_a}{\delta y^2} \left[W'_{i+\frac{1}{2}j+\frac{1}{2}} (u_{i+\frac{1}{2}j+1} - u_{i+\frac{1}{2}j}) - W'_{i+\frac{1}{2}j-\frac{1}{2}} (u_{i+\frac{1}{2}j} - u_{i+\frac{1}{2}j-1}) \right] \right\}, \quad (16) \end{aligned}$$

and

$$\begin{aligned}
 (\overline{vW})_{ij+\frac{1}{2}} = & \left(vW \right)_{ij+\frac{1}{2}}^n + \delta t \left\{ \frac{8}{15} \left[\frac{1}{\delta y} (\langle Wv:v \rangle_{ij} - \langle Wv:v \rangle_{ij+1}) \right. \right. \\
 & + \frac{1}{\delta x} (\langle W'u:v \rangle_{i-\frac{1}{2}j+\frac{1}{2}} - \langle W'u:v \rangle_{i+\frac{1}{2}j+\frac{1}{2}}) \left. \right] \\
 & + W_{ij+\frac{1}{2}} \beta g \left[\frac{T_{ij} + T_{ij+1}}{2} - \bar{T}_{j+\frac{1}{2}} \right] + \frac{v}{\delta x^2} \left[W'_{i+\frac{1}{2}j+\frac{1}{2}} (v_{i+1j+\frac{1}{2}} - v_{ij+\frac{1}{2}}) \right. \\
 & - W'_{i-\frac{1}{2}j+\frac{1}{2}} (v_{ij+\frac{1}{2}} - v_{i-1j+\frac{1}{2}}) \left. \right] + \frac{v}{\delta y^2} \left[W_{ij+1} (v_{ij+3/2} - v_{ij+\frac{1}{2}}) \right. \\
 & \left. \left. - W_{ij} (v_{ij+\frac{1}{2}} - v_{ij-\frac{1}{2}}) \right] \right\} . \quad (17)
 \end{aligned}$$

With these definitions, Eqs. (9) and (10) become

$$(uW)_{i+\frac{1}{2}j}^{n+1} = (\overline{uW})_{i+\frac{1}{2}j} + W'_{i+\frac{1}{2}j} \frac{\phi_{ij}^{n+1} - \phi_{i+1j}^{n+1}}{\delta x} \delta t - \frac{8v_{i+\frac{1}{2}j} \delta t}{W'_{i+\frac{1}{2}j}} u_{i+\frac{1}{2}j}^{n+1} , \quad (18)$$

$$(vW)_{ij+\frac{1}{2}}^{n+1} = (\overline{vW})_{ij+\frac{1}{2}} + W'_{ij+\frac{1}{2}} \frac{\phi_{ij}^{n+1} - \phi_{ij+1}^{n+1}}{\delta y} \delta t - \frac{8v_{ij+\frac{1}{2}} \delta t}{W'_{ij+\frac{1}{2}}} v_{ij+\frac{1}{2}}^{n+1} . \quad (19)$$

Among the values of W that occur at cell edges or cell corners, some are designated with a prime and some are not. In the absence of a prime the value of W is to be calculated as a simple average of the two or four adjacent cell-center values. The prime denotes the discrimination of a lesser of two values. On a cell face, it is the lesser of the two cell-center values. On a cell corner, for which the variables occur to the left of the colon in certain donor-cell terms and always in association with a velocity, the prime denotes the lesser of two cell-center averages: one from the two cells behind the direction of the velocity, the other from the two cells ahead of the direction of the velocity. Cell-corner W' values also occur in the v_a terms of Eqs. (16)

and (17), where the lesser of two averages is also denoted, in this case behind and ahead of the direction of the velocities they multiply. For the donor-cell and pressure gradient terms, the use of the lesser principle is based on the idea that crack width is constant over a whole cell, changing discontinuously at the edges, where we need to know the effective open area. In the v_a terms, the lesser principle is required to prevent excess artificial diffusive flux in regions of sharp gradients, where the excess flux could introduce numerical instability.

The solutions of Eqs. (18) and (19) are

$$\tilde{u}_{i+\frac{1}{2}j} = W_{i+\frac{1}{2}j}^{n+1} \left[\frac{(\overline{uW})_{i+\frac{1}{2}j} + W_{i+\frac{1}{2}j}^{n+1} \frac{\delta t}{\delta x} (\tilde{\phi}_{ij} - \tilde{\phi}_{i+1j})}{W_{i+\frac{1}{2}j}^{n+1} W_{i+\frac{1}{2}j}^{n+1} + v_{i+\frac{1}{2}j} \delta t} \right], \quad (20)$$

$$\tilde{v}_{ij+\frac{1}{2}} = W_{ij+\frac{1}{2}}^{n+1} \left[\frac{(\overline{vW})_{ij+\frac{1}{2}} + W_{ij+\frac{1}{2}}^{n+1} \frac{\delta t}{\delta y} (\tilde{\phi}_{ij} - \tilde{\phi}_{ij+1})}{W_{ij+\frac{1}{2}}^{n+1} W_{ij+\frac{1}{2}}^{n+1} + v_{ij+\frac{1}{2}} \delta t} \right]. \quad (21)$$

The u , v , and ϕ terms carry a tilde instead of an $n+1$ to show that these variables change through repetition of the iterative solution procedure each cycle, converging after sufficient iterations to the final values, for which $n+1$ then becomes the appropriate designation. The D equation likewise carries the tilde notation, with the definition in Eq. (6) transformed to the following:

$$\tilde{D}_{ij} = \frac{W_{ij}^{n+1} - W_{ij}^n}{\delta t} + \frac{2}{3} \left[\frac{(W\tilde{u})_{i+\frac{1}{2}j} - (W\tilde{u})_{i-\frac{1}{2}j}}{\delta x} + \frac{(W\tilde{v})_{ij+\frac{1}{2}} - (W\tilde{v})_{ij-\frac{1}{2}}}{\delta y} \right]. \quad (22)$$

Our problem is to solve for the zeros of $\tilde{D}_{ij}$ subject to the constraints of Eqs. (20) and (21). For each cell there are three equations in three unknowns, $\tilde{u}$, $\tilde{v}$, and $\tilde{\phi}$. Because these variables are mixed together in an implicit formulation, a direct solution is not feasible and we employ a Newton-Raphson iteration procedure. Each step of the iteration makes use of an equation for the change in $\tilde{D}_{ij}$ that results from a change in $\tilde{\phi}_{ij}$, namely,

$$\delta \tilde{D}_{ij} = \frac{\partial \tilde{D}_{ij}}{\partial \tilde{\phi}_{ij}} \delta \tilde{\phi}_{ij} \quad .$$

With the requirement that $\delta \tilde{D}_{ij}$ describe the change from a current value to zero, we can replace $\delta \tilde{D}_{ij}$ by $-\tilde{D}_{ij}$, solve for $\delta \tilde{\phi}_{ij}$, and thus derive

$$\tilde{\phi}_{ij}(\text{new}) = \tilde{\phi}_{ij}(\text{old}) - \gamma_{ij} \tilde{D}_{ij} \quad , \quad (23)$$

where

$$\frac{1}{\gamma_{ij}} \equiv \frac{\partial \tilde{D}_{ij}}{\partial \tilde{\phi}_{ij}} \quad . \quad (24)$$

The magnitude of γ_{ij} determines the rate of convergence of the iterative process but otherwise has no effect on the results. There are three ways to evaluate γ_{ij} , exact, approximate, and numerical. The exact analytical expression is easily derived but the formula is long and likely to consume more computer time for its calculation than is worthwhile. An approximate analytical formulation can be derived by ignoring the distinction among the various values of W and v , replacing them all by the cell-centered values. In this case we derive

$$\gamma_{ij} = \frac{3 \left[\left(W_{ij}^{n+1} \right)^2 + v_{ij} \delta t \right]}{4 \left(W_{ij}^{n+1} \right)^3 \delta t \left(\frac{1}{\delta x^2} \right) + \left(\frac{1}{\delta y^2} \right)} \quad . \quad (25)$$

Experience has shown that where the crack width is very small this approximation form can overestimate the value of γ . A satisfactory correction is achieved by omitting the $v_{ij} \delta t$ term from Eq. (25).

With these equations we can describe the sequence of logic in the iteration loop.

1. Calculate D_{ij} values, Eq. (22).
2. Calculate γ_{ij} and new $\tilde{\phi}_{ij}$ values, Eqs. (25) and (23).
3. Test for convergence. Convergence means that the maximum magnitude of D_{ij} is less than some preassigned quantity for which we choose $\epsilon W_p v_{in} / \delta y$, in which v_{in} is the inlet velocity at the bottom of the primary crack and ϵ is a small dimensionless number. If convergence has occurred and if this is not the first iteration, then the iteration is complete; otherwise go to step number 4.
4. Calculate $\tilde{u}$ and $\tilde{v}$ values, Eqs. (20) and (21).
5. Go back to step number 1.

The test for convergence and exit from iteration occurs just after the $\tilde{D}_{ij}$, γ_{ij} , and ϕ_{ij} calculations to ensure compatibility between D_{ij} and the velocities used in the equation for water temperature.

To calculate the changes in water temperature, we first compute

$$\begin{aligned}
 HC_{ij} = & (WT)_{ij}^n + T_{ij} D_{ij} \delta t + \frac{2\delta t}{3} \left[\frac{\langle uW':T \rangle_{i-\frac{1}{2}j} - \langle uW':T \rangle_{i+\frac{1}{2}j}}{\delta x} \right. \\
 & \left. + \frac{\langle vW':T \rangle_{ij-\frac{1}{2}} - \langle vW':T \rangle_{ij+\frac{1}{2}}}{\delta y} \right] \\
 & + \frac{\kappa_a \delta t}{\delta x^2} \left[W'_{i+\frac{1}{2}j} (T_{i+1j} - T_{ij}) - W'_{i-\frac{1}{2}j} (T_{ij} - T_{i-1j}) \right] \\
 & + \frac{\kappa_a \delta t}{\delta y^2} \left[W'_{ij+\frac{1}{2}} (T_{ij+1} - T_{ij}) - W'_{ij-\frac{1}{2}} (T_{ij} - T_{ij-1}) \right] . \quad (26)
 \end{aligned}$$

For the calculations described in this report, the diffusion terms, proportional to κ_a in Eq. (26), were not required and have been omitted.

For Eq. (11), we make the same type of finite difference approximation that was used in Eq. (12) for the time derivative. Then, for $i > i_1$ the simultaneous solution of Eqs. (11) and (13) gives

$$T_{ij}^{n+1} = \frac{HC_{ij} + \frac{2\kappa_0 \delta t}{\delta z} \xi_{ij2}}{W_{ij}^{n+1} + \frac{2\kappa_0 \delta t}{\delta z}} \quad (27)$$

For $2 < i < i_1$, we have, simply

$$T_{ij}^{n+1} = \frac{HC_{ij}}{W_{ij}^{n+1}} \quad (28)$$

For $i = 2$ and $i = i_1$, Eqs. (11) and (14) are solved simultaneously to give

$$T_{ij}^{n+1} = \frac{1}{W_p \left(1 + \frac{\kappa_0 \delta t}{\delta x^2}\right)} \left[HC_{ij} + \frac{\kappa_0 \delta z \delta t}{4\delta x^2} \sum_{k=1}^{\bar{k}-1} \left(\xi_{i_1+1jk}^{n+1} + \xi_{i_1+1jk+1}^{n+1} + \xi_{i_1+2jk}^{n+1} + \xi_{i_1+2jk+1}^{n+1} \right) \right] \quad (29)$$

V. BOUNDARY CONDITIONS

The boundary conditions for the calculation translate to finite difference expressions, as follows.

On most sides of the calculational mesh, the normal velocity and the normal gradient of tangential velocity vanish. For $2 \leq i \leq i_1$ and $j = 1$, $v_{ij+\frac{1}{2}} = v(\text{inlet}) \equiv VIN$, describing a constant inflow velocity across the bottom of the primary crack. The outlet velocity at the top of the primary crack is calculated each cycle in such a way as to account for changes in volume of the space open to flow. Thus, for $2 \leq i \leq i_1$ and $j = \bar{j}+1$, $v_{ij+\frac{1}{2}} = v(\text{outlet}) \equiv VO$, where

$$VO = VIN - \frac{3\delta y \sum (W_{ij}^{n+1} - W_{ij}^n)}{2(i_1 - 1)W_p \delta t} \quad (30)$$

The sum extends over all cells in the computational mesh. The water temperature is specified at the inlet, such that for $2 \leq i \leq i_1$, $T_{i1} = T(\text{inlet})$; otherwise the temperature in a cell just outside the mesh equals the adjacent interior temperature. The boundary conditions on the rock temperature, relevant only for $i \geq i_1$, are

$$\xi_{ij1} = T_{ij}$$

$$\xi_{ij\bar{k}+1} = \xi_{ij\bar{k}}$$

$$\xi_{i+2jk} = \xi_{\infty}$$

$$\xi_{ij+2k} = \xi_{\infty}$$

$$\xi_{i1k} = \xi_{\infty}$$

$$\xi_{i_1jk} = 2T_{i_1j} - \xi_{i_1+1jk}$$

The energy inlet and outlet rates are calculated each cycle and compared. The inlet rate is

$$\dot{E}(\text{inlet}) = (i_1 - 1) \delta x W_p \rho_o b_o T(\text{inlet}) v(\text{inlet}) \quad (31)$$

and the outlet rate is

$$\dot{E}(\text{outlet}) = \delta x W_p \rho_o b_o v(\text{outlet}) \sum_{i=2}^{i_1} T_{ij+1}$$

in which b_o is the specific heat of the water.

VI. NUMERICAL STABILITY

Meaningful calculations require input specifications that recognize the potentiality for numerical instability. Too large a time step per cycle, for example, can reduce the results to nonsense.

Two of the conditions for numerical stability that are often required in computations are eliminated in the present code by the implicit formulation of the difference equations. One of these would limit the propagation of sound signals to no more than the span of a computational cell per cycle. The other would limit the amount of momentum exchange with the wall per cycle as a result of viscous stress.

Several restrictions do occur, however, in the techniques we employ. One of these restricts the distance that fluid can travel in a computational cycle to less than the span of a cell. Thus $u_m \delta t / \delta x$ and $v_m \delta t / \delta y$ must both be less than unity at all times and positions, where u_m and v_m are the maximum magnitudes of velocity in the x and y directions. The techniques previously used for this type of study (see Bibliography) eliminated this restriction, thereby allowing the calculation to proceed through a succession of very large time steps, the results at each cycle describing the steady state flow pattern corresponding to the rock temperature profile at that moment. The disadvantage of such a procedure is that it becomes numerically unstable when the effects of buoyancy are dominant, which we expect in the case of relatively hot thermal fractures.

Another stability requirement comes from the explicit treatment of the heat diffusion in the rock. The classical restriction applies here, namely,

$$\kappa_R \delta t \left(\frac{1}{\delta x^2} + \frac{1}{\delta y^2} + \frac{1}{\delta z^2} \right) < \frac{1}{2} .$$

A similar restriction relates to the explicit treatment of the artificial viscosity, namely,

$$v_a \delta t \left(\frac{1}{\delta x^2} + \frac{1}{\delta y^2} \right) < \frac{1}{2} .$$

If the heat conduction term is incorporated into the calculation, then the same restriction occurs, with κ_a replacing v_a .

As an empirical property of the code version, GEO-I, in which the dynamics in the primary crack are resolved, we found that large input velocities tend to produce numerical instability unless the time step per cycle is reduced to a very small magnitude. This difficulty furnished part of the incentive for the development of GEO-II, in which the primary-crack dynamics are not resolved, and the difficulty associated with large input velocity is thereby eliminated.

To automatically ensure numerical stability and to furnish a basis for computing a criterion for convergence of the iterative solution, the computer program forms a quantity called v_{test} . The magnitude is determined by balancing the buoyancy and wall-friction terms in an analytical approximation from which this velocity can be estimated to a degree of accuracy that has proven adequate when the input velocity in the primary crack is small. Even when the input velocity is large we can roughly estimate the magnitude of v_{test} by including the effects of momentum convection. From v_{test} , δt is computed by the requirement that $v_{\text{test}} \delta t / \sqrt{\delta x^2 + \delta y^2}$ be equal to a specified fraction, for which 0.2 has usually been appropriate. The associated estimate for a convergence criterion for D is calculated to equal a small fraction (say 10^{-3}) of the ratio of v_{test} to cell size. Details of these estimate calculations are presented in the full listings of the computer codes.

VII. MODIFIED VERSIONS

In GEO-I, water flow is resolved in the region of the primary crack. As discussed above, the flow in the region of intersection between the cracks is potentially quite complicated, with motion taking place in a fully three-dimensional manner. Any two-dimensional version of the code is necessarily only an approximation, but we believe that the inaccuracy is a relatively small price to pay at this stage for the tremendous increase in computer efficiency afforded by the approximation.

A further step in the approximation is here described for the code version called GEO-II. For this we eliminate entirely the resolution in the primary crack, and obtain, in exchange, a much greater degree of computer efficiency for a wider scope of physical circumstances. In particular, great depths of secondary crack penetration can be calculated without requiring the small computer cells that were necessary for resolution of the narrow primary crack.

The details of GEO-II can be discerned in comparison of the two code listings, one for each version. The essential differences are described as follows:

1. The velocity and temperature of the water in every part of the primary crack remain at the specified inlet values throughout the calculation.

2. Fictitious computing cells just outside of the secondary crack are created with the same dimensions as those of the cells in the secondary crack. These carry boundary conditions that are not representative of the physical state of the primary crack, but describe instead the necessary fluid, temperature, and physical conditions for the intersection region. Thus we set in those fictitious cells a crack width equal to that of the adjacent true cell in the secondary crack, and the quantity W_p used in GEO-I is replaced by DC , the spacing between adjacent secondary cracks. We also specify a value for the modified pressure, ϕ , that represents an extrapolation from a prescribed constant value (usually arbitrarily scaled to equal zero) at the intersection. This choice of constant pressure is based upon the anticipated ease of modified pressure equilibration with the lateral parts of the primary crack.

3. Whereas GEO-I accounts for overall energy balance by differencing the flux between input and output positions, GEO-II performs the accounting by summing the net flux of energy at the opening edge of the secondary crack.

In addition, GEO-III has been written in such a way as to separate the advancement in rock temperature from the calculation of the water dynamics. The purpose is to allow calculations to proceed through large amounts of elapsed time. The computer code is split into two different types of update cycles. After the initialization, a prescribed number of fluid flow cycles are calculated until a steady state with the constant rock-temperature profile is achieved. Then, with unchanging fluid-flow profile and a much larger time step per cycle, the rock temperature is advanced for a prescribed number of cycles. A return to the fluid-flow calculation then advances the water velocities and temperatures to equilibrium with this new configuration of rock temperature. The calculation continues to alternate in this fashion between the advancement of rock temperature (through large increments of time) and the achievement of a new fluid-flow equilibrium (through a small increment of time).

VIII. CALCULATIONAL EXAMPLES

The examples that we describe are representative of the three versions of the computer code presented in this report.

The first three calculations, characterized by the input data in Tables I and II, use code version GEO-I, in which the primary crack is resolved. All three of these examples study the early stages of a secondary fracture, in which the depth of penetration is less than 10 cm, and the width of opening is less than 0.1 cm.

Example 1, Fig. 4. In this case, the calculation is continued over a long period of elapsed time (3100 sec) to demonstrate the persistence of a quasi-steady fluid flow configuration. After a relatively rapid adjustment from the initial conditions, the variations with time result principally from the slow variations of temperature in the surrounding rock.

The results of this calculation confirm the existence of two very different scales of time in these problems. One is the time scale for the adjustment of the fluid dynamics to the temperature configuration in the surrounding rock, which takes place within a few tens or hundreds of seconds. The second is the time scale for adjustment of the rock temperature to the effects of altered

TABLE I
INPUT CONSTANTS FOR THE EXAMPLE CALCULATIONS, GEO-I, GEO-II
(All values are in cgs units,
and temperatures are in degrees Celsius.)

BW	= 1.0
BR	= 0.25
RHOW	= 1.0
RHOR	= 2.65
AKR	= 6.2×10^{-3}
BETA	= 2.07×10^{-4}
BETAR	= 2.5×10^{-3}
GRAV	= 980.0
KBAR	= 4
TIN	= 65.0
XII	= 300.0

TABLE II
INPUT VARIABLES FOR THE EXAMPLE CALCULATIONS, GEO-I, GEO-II
(All values are in cgs units,
and temperatures are in degrees Celsius.)

Code	GEO-I	GEO-I	GEO-I	GEO-II	GEO-II
Example Number	1	2	3	4	5
Figure Number	4	5	6	7	8
Variables					
DC	-	-	-	30.0	60.0
DT	0.2	0.1	0.1	0.0	0.0
WP	10.0	10.0	10.0	-	-
DX	1.0	0.5	0.5	10.0	10.0
DY	1.0	0.5	0.5	10.0	10.0
EPSILON	1.0×10^{-3}	2.0×10^{-4}	1.0×10^{-3}	2.0×10^{-3}	2.0×10^{-3}
HEIGHT	10.0×10^{20}	5.0	5.0	-	-
IL	2	2	2	-	-
IBAR	10	20	20	20	20
JBAR	10	20	20	20	20
RNUA	0.0	0.0	0.0	0.01	0.01
THICK	4.0	5.0	5.0	50.0	100.0
VIN	1.0	1.0	0.1	10.0	10.0
WO	0.01	0.0	0.0	0.0	0.0

heat removal resulting from extension of the secondary fracture. This longer time scale varies with the evolutionary stage of the secondary fracture, except for early times it will be on the scale of hours, days, weeks, and even months.

Example 2, Fig. 5. This calculation is much like that of example 1, the principal difference being the fineness of resolution. In addition, we set $W_o = 0$, so that any degree of lateral contraction of the rock produces at least a small amount of crack opening, and we changed the initial crack configuration, as shown in the contour plots at early times. The effect of changing W_o is not great as long as W_o is small. The reason is that even with W_o equal to zero, the circulation is impeded almost as effectively in the regions with small crack opening as if the crack were not open at all.

Fig. 4. Example 1.

The notable result from this calculation is that the fluid motion in the secondary crack shows downward circulation near the intersection with the primary crack. It appears that the counterclockwise motion is sufficiently enhanced by both buoyancy and forced motion from water injection to produce the downward-directed current.

Example 3, Fig. 6. The principal change from example 2 is the large cut in fluid speed in the primary crack. Circulation in the secondary crack then results almost entirely from buoyancy and is not sufficiently strong to produce the downward motion near the intersection with the primary crack.

In both example 2 and example 3, the calculation had reached a quasi-steady state within 100 sec.

The next two examples were run with the code version GEO-II, with input data also shown in Tables I and II. Tests of similar calculations run with both versions show little difference between results. The two cases illustrated here, however, were performed with configurations much different from those of the first three examples. In both cases, the secondary cracks are much larger, and the spacing between them is also larger, so that there is much greater initial opening to the secondary crack.

Example 4, Fig. 7. In this case, the crack is tall and shallow, extending about 70 cm into the rock and initially open by a maximum gap size of 0.186 cm. A quasi-steady state is achieved within 50 sec. The flow configuration is like that of examples 1 and 3. The energy extraction rate attributable to this secondary crack is about 2.0×10^3 cal/sec.

Example 5, Fig. 8A-B. The initial secondary crack differs from that of example 4 by being deeper and having a greater spacing to the next adjacent secondary crack, so that the initial maximum gap opening is 0.391 cm. In this calculation we see considerable contortion of the flow field during the transition from initial conditions to the quasi-steady state, which is also achieved in about 50 sec. The reason for the transient variations can be traced to the ease of the rapid and deep circulation that occurs as soon as the calculation begins. Water is swept into a complex pattern of secondary circulation, which distorts the temperature field and with it the distribution of buoyancy forces. Despite this interesting sequence of intermediate states, the only relevant part of the calculation is the ultimately achieved quasi-steady state, from which all traces of the contortions have been lost. It was this realization that led to the double-time-step modification to GEO-II, resulting in GEO-III,

Fig. 5. Example 2.

Fig. 6. Example 3.

Fig. 7. Example 4.

in which the initial conditions for a sequence of water circulation calculations are generated by interspersed rock-temperature calculations.

Example 6, Fig. 9. Typical results from GEO-III, characterized by the input data in Table III, are presented to illustrate the crack penetration carried through a large extent of time.

All of the calculations show that the vertical extent of a secondary fracture grows very rapidly in the absence of any mitigating circumstances. In reality, the growth in that direction would likely terminate as soon as the fracture intrudes into the space between two other fractures. The calculations represent this effect by placing a restriction on growth at the top and bottom boundaries of the computational region. As a result the sequence of stages for a particular crack takes the form shown in Fig. 9, essentially independent of the initial crack configuration. A few representative calculations in which the crack height is held constant at several different dimensions show that the energy extraction rate is almost exactly proportional to that dimension. Accordingly, for the following studies, all performed with GEO-III, we held the crack height fixed at 200 cm.

Calculations were performed with water velocity in the primary crack varying from -20 cm/sec to +20 cm/sec, with results that are remarkably insensitive to that variation. These results support the contention that the water flow in the secondary crack is buoyancy-dominated for this velocity range. Thus for the analysis of trends, we held the fluid in the primary crack at rest, except insofar as it is influenced by the natural convection induced within the lateral fracture.

Each calculation commences with an initially small thermal fracture and follows the development through time until the results show a nearly steady rate of penetration and energy extraction. Two sets of studies are reported, one with various temperatures in the primary fracture and another with various lateral distances between secondary fractures. Typical results from the first set are shown in Figs. 10-13. For each case the lateral distance between thermal fractures is 50 cm. Figure 10 shows the maximum crack width as a function of time for water temperatures in the primary crack ranging from 65°C to 250°C, with a deep-rock temperature in each case of 300°C. In Fig. 11 are shown the crack penetration depths as functions of time. For the purposes of measurement, penetration depth is defined as the deepest extension of the crack-width contour corresponding to 20% of the maximum crack width. From each curve a

Fig. 8-A. Example 5.

Fig. 8-B. Example 5 (Con't).

Fig. 9. Example 6. The second number after T is the total elapsed time.

TABLE III
INPUT CONSTANTS AND VARIABLES FOR EXAMPLE CALCULATION, GEO-III
(All values are in cgs units,
and temperatures are in degrees Celsius.)

Example Number 6

Figures 9-19

BW	= 1.0
BR	= 0.25
RHOW	= 1.0
RHOR	= 2.65
AKR	= 6.2×10^{-3}
BETA	= 2.07×10^{-4}
BETAR	= 2.5×10^{-3}
DC	= Variable
DX	= Variable
DY	= 10.0
EPSILON	= 2.0×10^{-3}
GRAV	= 980.0
KBAR	= 4
HEIGHT	= 1.0×10^{10}
IBAR	= 20
JBAR	= 20
RNUA	= 0.01
THICK	= 50.0
TIN	= Variable
VIN	= 0.0
WO	= 0.0
XII	= 300.0

Fig. 10. Maximum crack width as a function of time for various temperatures in the primary crack.

Fig. 11. Crack penetration depth as a function of time for various temperatures in the primary crack.

penetration speed is calculated. Figure 12 shows the maximum speed of water flow in the secondary crack. The total heat energy extraction rates as functions of time are presented in Fig. 13.

Similar calculations were performed with various lateral spacings between the thermal fractures, all with 150°C in the primary crack. The results are described in Figs. 14-17.

Fig. 12. Maximum water speed in the thermal fracture as a function of time for various temperatures in the primary crack.

Fig. 13. Total heat energy extraction rate as a function of time for various temperatures in the primary crack.

Figure 18 shows the total actual volume in the thermal fracture as a function of time for various temperatures in the primary crack and lateral spacings between thermal fractures.

The late-time results from these calculations are summarized in Fig. 19 as functions of temperature in the primary crack, and as functions of lateral spacing between thermal fractures. The correlation of these results will be presented elsewhere* together with a discussion of their significance to the study of multiple thermal fracture growth and the enhancement of heat extraction from the entire geothermal system.

* Ruth B. Demuth and Francis Harlow, "Geothermal Energy Enhancement by Thermal Fracture," Los Alamos Scientific Laboratory report in preparation.

Fig. 14. Maximum crack width as a function of time for various lateral spacings between thermal fractures.

Fig. 15. Crack penetration depth as a function of time for various lateral spacings between thermal fractures.

Fig. 16. Maximum water speed in the thermal fracture as a function of time for various lateral spacings between thermal fractures.

Fig. 17. Total heat energy extraction rate as a function of time for various lateral spacings between thermal fractures.

Fig. 18. Total volume of the thermal-fracture crack as a function of time for various temperatures in the primary crack and lateral spacings between thermal fractures.

IX. THE COMPUTER PROGRAM

The numerical procedure for solving the equations is implemented by a computer code called GEO. We describe the procedure in detail for the version, GEO-I. The structures of GEO-II and GEO-III are closely similar, the differences being exhibited by full listings of all versions. The flow of logic for a computational cycle is shown in Fig. 20A-B. Full listings of the computer programs are given in Appendix A-C. Calculations with GEO-III, for a problem with 5290 cells, typically require 7 min of CDC-7600 CPU time, utilizing 125,325 octal words of large core memory (LCM) storage. Here we describe the input and running procedure.

A. General Structure

GEO is a computer code written to operate under the Livermore Time-Sharing System (LTSS) on the CDC-7600. The program is designed with one main subroutine to handle the central flow of the logic and several other subroutines, including those required for output in the form of printing and plotting. The steps in a cycle are as follows.

Fig. 19. Summary of the late-time results as functions of primary-crack temperature and of the lateral spacing between thermal fractures.

Fig. 20-A. Flow diagram for the computer codes GEØ-I and GEØ-II.

1. The first subroutine called is a library routine UNQCHG, which changes the name of the drop file to a unique name. This prevents the original file from being continuously overwritten with the current image of the executing program, and allows the drop file to be available for repeated execution with different input data and for use with DBCTRL, the debugging program on LTSS. A call to UNDRØP follows. This subroutine is a GEØ routine which contains a series of LTSS library system-interface routines that create output and film files required by the code.
2. The program then proceeds to the main subroutine, which contains an initial variable input list, initial conditions, and boundary conditions before the computational cycle begins. At this point, the code

Fig. 20-B. Flow diagram for the computer code GEØ-III.

- calls the print and plot routines in order to print and plot at a time of zero, so that initial conditions may be checked.
3. The computational cycle begins with additional boundary conditions on ξ , the rock temperature, followed by calculations of the new values of ξ . The new values of W , the crack width, are stored and the updated W values are then calculated. Next it is necessary to establish boundary conditions on the velocity and the water temperature.
 4. The bar quantities are initialized for iteration.
 5. The iteration process is a subcycling within the main cycle. D , the mass equation discrepancy, is calculated; γ and the new value of ϕ are calculated; and tests are made for convergence. If convergence has been obtained, the flow leaves the iteration loop and continues through the cycle. If convergence has not been obtained, the new

velocities are then calculated and the flow returns to the beginning of the iteration loop.

6. The new water temperatures are calculated.
7. The GEØ print (PRINT) and plotting (PLØTPKG) subroutines are now called. These produce output in the form of microfiche plots and printed information on either microfiche or fanfold paper. PLØTPKG in turn calls the modified LTSS library routines PLØJB, CØNTRJB, and TRICJB. A test is made to see if the finish time has been reached.
8. When the printing and plotting are finished for the last cycle, it is time to terminate the program. This is done by calling the routine DRØP, which calls several system interface routines to close job files. The library routine ØOTIM is called in order to obtain the integer microseconds charged for the run, which are then converted to seconds and printed out at the end of the output. Subroutine EXIT now terminates the run. To leave a drop file in the user file space, CALL EXIT(n) must be used, where n is a nonzero integer. This form is recommended for debugging. The library routines ALLØUT and GIVE may now be used to dump the output as explained in Sec. IX-E.

B. The Indexing Notation

Figure 21 shows that some variables are located at cell centers and some at cell edges, a typical arrangement for many Eulerian codes. FØRTRAN representation of these values is complicated by the half-integer indices, as T_{ij} can be represented as $T(I,J)$, but $u_{i+\frac{1}{2}j}$ can not be represented as $U(I + .5,J)$. Our convention is shown in Fig. 21, such, for example, that $u_{i+\frac{1}{2}j}$ becomes in FØRTRAN $U(I,J)$ and $v_{ij+\frac{1}{2}}$ becomes $V(I,J)$.

The computer storage grid must be at least $(\bar{I}+1)$ by $(\bar{J}+1)$, as the number of cell edges is one greater than the number of cells in each direction. In our FØRTRAN index notation, the letter P is used to represent +, and M represents -. For example,

$$IBP1 \equiv IBAR + 1$$

$$JBM1 \equiv JBAR - 1$$

Fig. 21. Location of the principal variables in a GEØ cell.

In addition, it is necessary to surround the computing mesh with an additional set of cells on all four sides, which gives an effective $(\bar{I}+2)$ by $(\bar{J}+2)$ mesh as seen in Fig. 22. These cells are needed to handle the boundary conditions.

Fig. 22. FORTRAN computing mesh used in GEØ. Dashed lines indicate fictitious cells.

C. The Input and Derived Data

The data are input directly from initial variable values at the beginning of the code. These values can be changed easily if new problem conditions are desired. The following list gives the names and a brief description of these variables.

<u>NAME</u>	<u>DESCRIPTION</u>
AKAPO	κ_0 , ratio of thermal conductivity in rock to the product of the density of water and specific heat coefficient for water.
AKAPR	κ_R , thermometric conductivity of rock.
AKR	k_R , coefficient of heat conduction in rock.
BETA	β_w , volumetric expansion coefficient for water.
BETAR	β_R , linear expansion coefficient for rock.
BR	b_R , specific heat of rock.
BW	b_w , specific heat of water.
DC	lateral distance between secondary cracks.
DT	δt , the time step per hydrodynamic cycle.
DTR	$1/\delta t$.
DTXI	δt_ξ , the time step per each temperature cycle.
DX	δx , the cell size in the horizontal direction.
DXR	$1/\delta x$.
DXSQ	δx^2 .
DXSQR	$1/\delta x^2$.
DY	δy , the cell size in the vertical direction.
DYR	$1/\delta y$.
DYSQ	δy^2 .
DYSQR	$1/\delta y^2$.
DZ	δz , the cell size in the direction normal to the secondary crack.
DZR	$1/\delta z$.
DZSQ	δz^2 .
DZSQR	$1/\delta z^2$.
EIGFIF	8/15.
EPSILON	ϵ , the convergence criterion test number for the pressure iteration.
GRAV	g , acceleration of gravity.

HEIGHT	A measure of initial crack height: if height is a large number, then the lateral crack is of uniform thickness and extends from the bottom to the top of the mesh.
II	i_1 , one more than the number of computational cells across the primary crack.
IIP1	i_1+1 .
IIP2	i_1+2 .
IIM1	i_1-1 .
IBAR	$\bar{i}$, the total number of calculational cells in the horizontal direction.
IBP1	$\bar{i}+1$
IBP2	$\bar{i}+2$.
JBAR	$\bar{j}$, the total number of calculational cells in the vertical direction.
JBP1	$\bar{j}+1$.
JBP2	$\bar{j}+2$.
KBAR	$\bar{k}$, the total number of computational cells in the direction normal to the secondary crack.
KBP1	$\bar{k}+1$.
KBM1	$\bar{k}-1$.
NEW	Current number of the hydrodynamic-cycle group.
NEWF	Total required number of hydrodynamic-cycle groups.
NXIF	Number of cycles per group of rock-temperature updates.
NCYF	Number of cycles per group of hydrodynamic updates.
NCYI	Number of cycles between hydrodynamic prints.
NCYP	Print interval counter.
NCYTER	Terminal interval counter.
NCYTERI	Number of hydrodynamic cycles between terminal messages.
RHØR	ρ_R , density of rock.
RHØW	ρ_w , density of water.
RNUA	ν_a , coefficient of artificial kinematic viscosity.
THICK	Thickness of initial cooling zone in the rock.
TI	t, problem time within each set of hydro cycles.
TIN	Water temperature at inlet.
TF	Time to finish.
TP	Time to print.

TPC	Time print constant.
TPI	Time print interval.
TTI	Total elapsed problem time.
TWØTHI	2/3.
VIN	Velocity at inlet.
WO	W_0 , yield width for crack production.
WP	W_p , width of primary crack.
XII	ϵ_∞ , rock temperature far from the crack.

D. The Common Blocks, VAR and ARRAYS

In addition to the variables contained in the input data (Sec. IX-C), which can be set initially, the SCM common block VAR also contains

DSUM	Measure of total nonconvergence discrepancy.
NCYCLE	Number of cycles in the problem at a given time.
NIT	Number of iterations within a given cycle.
VØUT	Velocity at outlet.
WMAX	Largest width within the crack.

The common block, ARRAYS, contains the cell storage information below.

D(I,J)	D, mass equation discrepancy.
GAM(I,J)	γ , Newton-Raphson relaxation coefficient.
HC(I,J)	Heat convection term.
PHI(I,J)	ϕ , modified pressure/density.
T(I,J)	T, water temperature.
TBAR(J)	$\bar{T}$, mean temperature for each level.
TNP1(I,J)	T^{n+1} , new water temperature.
U(I,J)	u, velocity in horizontal direction.
UWBAR(I,J)	$\bar{uW}$, initialization quantity for iteration.
V(I,J)	v, velocity in vertical direction.
VWBAR(I,J)	$\bar{vW}$, initialization quantity for iteration.
W(I,J)	W, crack width.
WNP1(I,J)	W^{n+1} , new crack width.
XI(I,J)	ξ , rock temperature.
XINP1(I,J)	ξ^{n+1} , new rock temperature.

E. Output-Prints, Plots

The code GEØ provides output in two forms, microfiche plots and printed output information on either microfiche or fanfold paper. The microfiche plots are created by using the library GIVE program, followed by the GEØ-created FX10512XXX file. The printed output can be dumped on microfiche by using the ALLØUT FR80 GEØØUT option or on paper with ALLØUT GEØØUT. This output is created automatically at a time $t = 0$. Thereafter, the plot and print times are controlled by setting TPI (time print interval) in the initial variable list.

The plot package (subroutine PLØTPKG) allows the user to create three contour plots and one vector plot. Contour plots are given for the arrays WNPI (crack width), T (water temperature), and PHI (pressure, as modified by the removal of the hydrostatic part and the normalization of buoyancy force to zero in the primary crack). These contour plots are created by connected vector segments, which join points of equal value. The contour increment is linear.

The vector plot is drawn with each vector originating from a cell center, and the vector length and direction are determined by the x and y velocity components. The vectors are normalized such that the maximum magnitude lies within one cell size.

In addition to the plot capability, GEØ also provides for printed numerical data. The long print is a complete numerical listing of the major field variables in each cell. The columns in a listing contain the I and J of each cell, followed by seven of the two-dimensional field variables, and rock temperatures in the first three cells of the three-dimensional array:

$$i, j, u_{i+\frac{1}{2},j}, v_{i,j+\frac{1}{2}}, \phi_{ij}, T_{ij}, W_{ij}^{n+1}, \gamma_{ij}, D_{ij}, \xi_{ij2}, \xi_{ij3}, \xi_{ij4} \quad .$$

Preceding each long print is a short print, which gives the time, number of cycles, and number of iterations. It also gives the energy inflow rate through the inlet (EDØTIN), energy outflow rate through the outlet (EDØTØUT), the difference between these outflow and inflow rates (EDØTDIF), and DSUM, the sum of all the D_{ij} values in the mesh. DSUM is used as a check on the calculation; its magnitude should always be very small compared with DEPS, which

is calculated each cycle as a function of EPSILON. In addition, all of the initial input variables are printed between the short and long prints at $t = 0$.

Typical printed listings of output are given in Appendix D.

F. Operating the Terminal

The GEØ code is run from the remote terminal using the LTSS operating system. The program is currently resident on CFS and can be accessed upon request. There is a LIX file containing the source code (GEØSØ), the load module (GEØLØ), the map (GEØMA), the symbol table (GEØSY), and a utility program (UTIL) for ease of compiling and running. The compile file (GEØLS), binary file (GEØBI), and executable file (GEØ) are created at compile and load time and the output (GEØØUT) and plot (FX10512XXX) files are available after running.

After getting the code from CFS with MASS, it is necessary to use LIX to establish the individual files in the working space. Perform a GR ALL under LIX, and the files will be available for individual use. Next, in TRIX, read the utilities (RD!UTIL!). Then using the CHAT compiler and these utilities, compile as follows:

```
compl(GEØ) and after the prompt,  
load(GEØ).
```

The program may also be run without the utility program by using conventional CHAT compile and load lines.

After leaving TRIX, the program may then be run by simply typing GEØ. The current running time, cycle number, iteration number, volume of the secondary crack, and energy difference will be typed on the terminal as the program runs. After the program finishes running, the output and plots may be dumped using ALLØUT and GIVE as explained in Sec. IX-E.

APPENDIX A
CODE LISTING FOR GEØ-I
Example Numbers 1-3
Figures 4-6

LASL Identification Number LP-1092

```

1*****
000001 000001 C
000002 000002 C  VARIABLE DEFINITIONS
000003 000003 C
000004 000004 C
000005 000005 C  AKAPR - KAPPA ROCK THERMOMETRIC CONDUCTIVITY OF ROCK
000006 000006 C  AKAPD - RATIO OF THERMAL CONDUCTIVITY IN ROCK TO THE
000007 000007 C          PRODUCT OF THE DENSITY OF WATER AND VOLUMETRIC
000008 000008 C          EXPANSION COEFFICIENT FOR WATER
000009 000009 C  BETA  - BETA WATER- VOLUMETRIC EXPANSION COEFFICIENT FOR WATER
000010 000010 C  HETAR - BETA ROCK-      LINEAR EXPANSION COEFFICIENT FOR ROCK
000011 000011 C  DT    - DELTA T, CHANGE IN THE ELAPSED TIME
000012 000012 C  DX    - CHANGE IN THE DISTANCE IN X DIRECTION
000013 000013 C  DY    - CHANGE IN THE DISTANCE IN Y DIRECTION
000014 000014 C  DZ    - CHANGE IN THE DISTANCE IN Z DIRECTION
000015 000015 C  GAM   - NEWTON RAPHSON RELAXATION COEFFICIENT
000016 000016 C  GRAV  - ACCELERATION OF GRAVITY
000017 000017 C  HC    - HEAT CONVECTION TERMS
000018 000018 C  IL    - RIGHT MESH BOUNDARY OF PRIMARY FRACTURE
000019 000019 C  IBAR  - RIGHT BOUNDARY OF TOTAL MESH IN X DIRECTION
000020 000020 C  JBAR  - TOP BOUNDARY OF MESH IN Y DIRECTION
000021 000021 C  KBAR  - TOP BOUNDARY OF MESH IN Z DIRECTION
000022 000022 C  PHI   - PRESSURE/ DENSITY
000023 000023 C  RNU   - COEFFICIENT OF KINEMATIC VISCOSITY
000024 000024 C  RNUA  - COEFFICIENT OF ARTIFICIAL KINEMATIC VISCOSITY
000025 000025 C  T     - WATER TEMPERATURE
000026 000026 C  TF    - TOTAL CALCULATION TIME
000027 000027 C  THICK - THICKNESS OF INITIAL COOLING ZONE IN THE ROCK
000028 000028 C  TI    - RUNNING PROBLEM TIME
000029 000029 C  TIN   - WATER TEMPERATURE AT INLET
000030 000030 C  TP    - TIME TO PRINT
000031 000031 C  TPC   - TIME PRINT CONSTANT
000032 000032 C  TPI   - TIME PRINT INTERVAL
000033 000033 C  U     - VELOCITY IN X DIRECTION
000034 000034 C  V     - VELOCITY IN Y DIRECTION

```

```

000035 000035 C VIN - VELOCITY AT INLET
000036 000036 C VOUT - VELOCITY AT OUTLET
000037 000037 C W - CRACK WIDTH
000038 000038 C W0 - YIELD WIDTH FOR CRACK PRODUCTION
000039 000039 C WP - WIDTH OF PRIMARY CRACK
000040 000040 C XI - ROCK TEMPERATURE
000041 000041 C XII - ROCK TEMPERATURE AT INFINITY
000042 000042 C
000043 000043 C
000044 000044 C
000045 000045 C PROGRAM GEN(OUTPUT,GENOUT,TAPE4=GEOUT,TAPE6=100B)
000046 000046 C
000047 000047 C
000048 000048 C PARAMETER (NI=23)
000049 000049 C PARAMETER (NJ=23)
000050 000050 C PARAMETER (NK=10)
000051 000051 C COMMON/ARAYS/GAM(NI,NJ),HC(NI,NJ),PHI(NI,NJ),T(NI,NJ),
000052 000052 1 UMBAR(NI,NJ),VMBAR(NI,NJ),U(NI,NJ),
000053 000053 2 V(NI,NJ),W(NI,NJ),WNP1(NI,NJ),XI(NI,NJ,NK),
000054 000054 3 XINP1(NI,NJ,NK),D(NI,NJ),INP1(NI,NJ),TRAH(NJ)
000055 000055 C
000056 000056 C
000057 000057 C COMMON/VAR/AKAP0,AKAPR,HETA,HETAR,DT,DTM,DX,DY,DZ,DAR,DYR,DZR,
000058 000058 1 DXSQ,DYSQ,DZSQ,DXSOR,DYSOR,DZSOR,EIGFIF,EPSI,ON,GRAV,
000059 000059 2 I1,I1P1,I1P2,I1M1,IRAR,IRP1,IRP2,JRAR,JBP1,JRP2,
000060 000060 3 KBAR,KRP1,KRM1,NCYCLE,NIT,RNUA,TI,TF,TP,
000061 000061 4 TPC,TPI,THICK,TIN,TWOTH1,VIN,VOUT,W0,WP,XII,
000062 000062 5 BR,BW,RHOR,RHOW,AKR,HEIGHT,DSUM
000063 000063 C
000064 000064 C
000065 000065 C CALL INQCMB
000066 000066 C CALL UNDRAP
000067 000067 C
000068 000068 C
000069 000069 C TIMING STUDIES
000070 000070 C
000071 000071 C CALL TOTIM
000072 000072 C ISEC = 0
000073 000073 C
000074 000074 C
000075 000075 C
000076 000076 C VARIABLES
000077 000077 C
000078 000078 C BW = 1.0
000079 000079 C BR = 0.25
000080 000080 C RHOW = 1.0
000081 000081 C RHOR = 2.65
000082 000082 C AKR = 6.2E-3
000083 000083 C AKAP0 = AKR / (RHOW*HW)
000084 000084 C AKAPR = AKR / (RHOR*BR)

```

000085	000085	BETA = 2.07E-4
000086	000086	BETAR = 2.5E-5
000087	000087	DT = 0.1
000088	000088	DTR = 1./DT
000089	000089	KBAR = 4
000090	000090	KBP1 = KBAR+1
000091	000091	KBM1 = KBAR-1
000092	000092	WP = 10.0
000093	000093	DX = 0.5
000094	000094	DY = 0.5
000095	000095	DZ = 0.5*WP/FLOAT(KBM1)
000096	000096	DXR = 1./DX
000097	000097	DYR = 1./DY
000098	000098	DZR = 1./DZ
000099	000099	DXSQ = DX**2
000100	000100	DYSQ = DY**2
000101	000101	DZSQ = DZ**2
000102	000102	DXSQR = 1./DXSQ
000103	000103	DYSQR = 1./DYSQ
000104	000104	DZSQR = 1./DZSQ
000105	000105	EIGFIF = 0./15.
000106	000106	EPSILON = 1.E-3
000107	000107	GRAV = 900.0
000108	000108	HEIGHT = 5.0
000109	000109	II = 2
000110	000110	IIP1 = II-1
000111	000111	IIP2 = II-2
000112	000112	IIM1 = II-1
000113	000113	IHAR = 20
000114	000114	IHP1 = IHAR+1
000115	000115	IHP2 = IHAR+2
000116	000116	JBAR = 20
000117	000117	JBP1 = JBAR+1
000118	000118	JBP2 = JBAR+2
000119	000119	RNUA = 0.0
000120	000120	TI = 0.0
000121	000121	TF = 100.0
000122	000122	TP = 0.0
000123	000123	TPC = (1.E-3)*DT
000124	000124	TP1 = 10.0
000125	000125	THICK = 5.0
000126	000126	TIN = 65.0
000127	000127	TWOHT = 2./3.
000128	000128	VIN = 0.1
000129	000129	W0 = 0.0
000130	000130	XII = 300.0

000131	000131	C
000132	000132	C
000133	000133	C INITIAL CONDITIONS
000134	000134	C
000135	000135	DO 199 I = 2,JBAR
000136	000136	DO 198 J = 2,JBP1
000137	000137	U(I,J) = 0.0
000138	000138	198 CONTINUE
000139	000139	199 CONTINUE
000140	000140	C
000141	000141	DO 299 I = 2,JBP1
000142	000142	DO 298 J = 2,JBAR
000143	000143	V(I,J) = 0.0
000144	000144	298 CONTINUE
000145	000145	299 CONTINUE
000146	000146	C
000147	000147	C
000148	000148	DO 359 I = 2,II
000149	000149	V(I,1) = VIN
000150	000150	V(I,JBP1) = VIN
000151	000151	T(I,1) = TIN
000152	000152	359 CONTINUE
000153	000153	C
000154	000154	C CALCULATE INDEX FOR THE DOWN SIDE OF MIDDLE
000155	000155	C
000156	000156	HJBAR = FLOAT(JBAR)*.5 + 1.01
000157	000157	JM = HJBAR
000158	000158	DO 379 I = IIP1,IBP1
000159	000159	DO 378 J = 2,JBP1
000160	000160	DO 377 K = 1,KBAR
000161	000161	XRAR = (FLOAT(I-II) - 0.51*DX)/THICK
000162	000162	YBAR = (FLOAT(J-JM)*DY)/HEIGHT
000163	000163	RADIUS = SQRT(XBAR**2 + YBAR**2)
000164	000164	XI(I,J,K) = TIN + (XII-TIN)*ERF(RADIUS)
000165	000165	377 CONTINUE
000166	000166	378 CONTINUE
000167	000167	379 CONTINUE
000168	000168	C
000169	000169	DO 399 I = 2,II
000170	000170	DO 398 J = 2,JBP1
000171	000171	T(I,J) = TIN
000172	000172	PHI(I,J) = 0.0
000173	000173	398 CONTINUE
000174	000174	399 CONTINUE
000175	000175	C
000176	000176	DO 439 I = IIP1,IBP1
000177	000177	DO 438 J = 2,JBP1
000178	000178	T(I,J) = XI(I,J,1)
000179	000179	PHI(I,J) = 0.0
000180	000180	438 CONTINUE
000181	000181	439 CONTINUE

```

000182 000182 C
000183 000183 DO 599 I = 1,IP1,IBP1
000184 000184 DO 598 J = 2,JBP1
000185 000185 SUM1 = 0.0
000186 000186 DO 597 K = 2,KBP1
000187 000187 SUM1 = SUM1 + (XI(I,J,K) - XII)
000188 000188 597 CONTINUE
000189 000189 WNP1(I,J) = -2.0*RETA*0.7*(.5*(XI(I,J,1)-XII)+SUM1)
000190 000190 IF(WNP1(I,J).LE.W0) WNP1(I,J) = 0.0
000191 000191 W(I,J) = WNP1(I,J)
000192 000192 598 CONTINUE
000193 000193 599 CONTINUE
000194 000194 C
000195 000195 DO 659 I = 2,II
000196 000196 DO 658 J = 1,JBP2
000197 000197 W(I,J) = WP
000198 000198 WNP1(I,J) = WP
000199 000199 658 CONTINUE
000200 000200 659 CONTINUE
000201 000201 C
000202 000202 C
000203 000203 C BOUNDARY CONDITIONS ON XI
000204 000204 C
000205 000205 DO 698 J = 2,JRP1
000206 000206 DO 697 K = 1,KBP1
000207 000207 XI(1BP1,J,K) = XII
000208 000208 697 CONTINUE
000209 000209 698 CONTINUE
000210 000210 C
000211 000211 C
000212 000212 DO 799 I = 1,IP1,IBP1
000213 000213 DO 797 K = 1,KBP1
000214 000214 XI(1,JBP2,K) = XII
000215 000215 XI(1,1,K1) = XII
000216 000216 797 CONTINUE
000217 000217 799 CONTINUE
000218 000218 C
000219 000219 C SET CYCLE COUNTERS
000220 000220 C
000221 000221 NCYCIF = 0
000222 000222 900 CONTINUE
000223 000223 C
000224 000224 C
000225 000225 C CHECK FOR PRINTING INTERVAL
000226 000226 C
000227 000227 IF(II + TPC.GE.TP) GO TO 5100

```

```

000228 000228 C
000229 000229 C*****
000230 000230 C***** BEGIN CYCLE *****
000231 000231 C*****
000232 000232 C
000233 000233 C
000234 000234 C
000235 000235 1100 CONTINUE
000236 000236 C
000237 000237 C BOUNDARY CONDITIONS ON XI
000238 000238 C
000239 000239 DO 1199 I = 1,IP1,IRP1
000240 000240 DO 1199 J = 2,JRP1
000241 000241 XI(I,J,1) = T(I,J)
000242 000242 XI(I,J,KRP1) = XI(I,J,KBAR)
000243 000243 1198 CONTINUE
000244 000244 1199 CONTINUE
000245 000245 C
000246 000246 DO 1299 J = 2,JBP1
000247 000247 DO 1299 K = 1,KBP1
000248 000248 XI(I,J,K) = 2.0*T(I,J) - XI(I,JP1,J,K)
000249 000249 1297 CONTINUE
000250 000250 1298 CONTINUE
000251 000251 C
000252 000252 C CALCULATE NEW XI
000253 000253 C
000254 000254 DO 1399 I = 1,IP1,IRP1
000255 000255 DO 1399 J = 2,JBP1
000256 000256 DO 1397 K = 2,KBAR
000257 000257 XINP1(I,J,K) = XI(I,J,K) + AKAPR*DT*
000258 000258 1 (DXSQ*(XI(I+1,J,K)-XI(I-1,J,K))-2.0*XI(I,J,K))
000259 000259 2 + DYSQ*(XI(I,J+1,K)-XI(I,J-1,K))-2.0*XI(I,J,K))
000260 000260 3 + DZSQ*(XI(I,J,K+1)-XI(I,J,K-1))-2.0*XI(I,J,K))
000261 000261 1397 CONTINUE
000262 000262 1398 CONTINUE
000263 000263 1399 CONTINUE
000264 000264 C
000265 000265 C STORE NEW XI INTO OLD
000266 000266 C
000267 000267 DO 1499 I = 1,IP1,IRP1
000268 000268 DO 1499 J = 2,JBP1
000269 000268 DO 1497 K = 2,KBAR
000270 000270 XI(I,J,K) = XINP1(I,J,K)
000271 000271 1497 CONTINUE
000272 000272 1498 CONTINUE
000273 000273 1499 CONTINUE

```

```

000274 000274 C
000275 000275 C STORE NEW W INTO OLD W AND CALCULATE NEW W
000276 000276 C
000277 000277 DO 1599 I = 1,IP1,IRP1
000278 000278 DO 1599 J = 2,JBP1
000279 000279 W(I,J) = WNP1(I,J)
000280 000280 SUM1 = 0.0
000281 000281 DO 1597 K = 2,KRAP1
000282 000282 SUM1 = SUM1 + (X1(I,J,K) - X1I)
000283 000283 1597 CONTINUE
000284 000284 WNP1(I,J) = -2.0*RETAR*D7*(.5*(X1(I,J,1)-X1I) + SUM1)
000285 000285 IF(WNP1(I,J).LE.W0) WNP1(I,J) = 0.0
000286 000286 IF(WNP1(I,J).NE.0.0.AND.W(I,J).EQ.0.) T(I,J)=XINP1(I,J,1)
000287 000287 1598 CONTINUE
000288 000288 1599 CONTINUE
000289 000289 C
000290 000290 C BOUNDARY CONDITIONS ON VELOCITY AND TEMPERATURE
000291 000291 C
000292 000292 FIIM1 = FLOAT(I1)-1.0
000293 000293 SUM2 = 0.0
000294 000294 DO 1699 I = 1,IP1,IRP1
000295 000295 DO 1699 J = 2,JBP1
000296 000296 SUM2 = SUM2 + WNP1(I,J) - W(I,J)
000297 000297 1698 CONTINUE
000298 000298 1699 CONTINUE
000299 000299 VOUT = VIN - 1.5*DY*SUM2/(FIIM1*WP*DT)
000300 000300 DO 1799 I = 2,I1
000301 000301 V(I,1) = VIN
000302 000302 V(I,JBP1) = VOUT
000303 000303 T(I,1) = TIN
000304 000304 1799 CONTINUE
000305 000305 C
000306 000306 C
000307 000307 C INITIALIZE FOR ITERATION
000308 000308 C
000309 000309 C
000310 000310 DO 2199 I = 2,IBAR
000311 000311 DO 2199 J = 2,JBP1
000312 000312 UC = .5*(U(I-1,J) + U(I,J))
000313 000313 IF(UC.GE.0.) A = W(I,J)*UC*U(I-1,J)
000314 000314 IF(UC.LT.0.) A = W(I,J)*UC*U(I,J)
000315 000315 UC = .5*(U(I,J) + U(I+1,J))
000316 000316 IF(UC.GE.0.) B = W(I+1,J)*UC*U(I,J)
000317 000317 IF(UC.LT.0.) B = W(I+1,J)*UC*U(I+1,J)
000318 000318 WFAC = .5*AMIN1(W(I,J-1)*W(I+1,J-1),W(I,J)*W(I+1,J))
000319 000319 VC = .5*(V(I,J-1) + V(I+1,J-1))
000320 000320 IF(VC.GE.0.) C = WFAC*VC*U(I,J-1)
000321 000321 IF(VC.LT.0.) C = WFAC*VC*U(I,J)
000322 000322 WFAC = .5*AMIN1(W(I,J)*W(I+1,J),W(I,J+1)*W(I+1,J+1))
000323 000323 VC = .5*(V(I,J) + V(I+1,J))
000324 000324 IF(VC.GE.0.) D1 = WFAC*VC*U(I,J)
000325 000325 IF(VC.LT.0.) D1 = WFAC*VC*U(I,J+1)

```

000326 000326
 000327 000327
 000328 000328
 000329 000329
 000330 000330
 000331 000331
 000332 000332
 000333 000333
 000334 000334
 000335 000335
 000336 000336
 000337 000337
 000338 000338
 000339 000339
 000340 000340
 000341 000341
 000342 000342
 000343 000343
 000344 000344
 000345 000345
 000346 000346
 000347 000347
 000348 000348
 000349 000349
 000350 000350
 000351 000351
 000352 000352
 000353 000353
 000354 000354
 000355 000355
 000356 000356
 000357 000357
 000358 000358
 000359 000359
 000360 000360
 000361 000361
 000362 000362
 000363 000363
 000364 000364
 000365 000365
 000366 000366
 000367 000367
 000368 000368
 000369 000369
 000370 000370
 000371 000371
 000372 000372
 000373 000373
 000374 000374
 000375 000375
 000376 000376
 000377 000377
 000378 000378
 000379 000379
 000380 000380

WFAA = .5*AMIN1(W(I,J+1)+W(I+1,J+1)+W(I,J)+W(I+1,J))
 WFAA = .5*AMIN1(W(I,J)+W(I+1,J)+W(I,J-1)+W(I+1,J-1))
 C
 UWBAR(I,J) = (U(I,J)+.5*(W(I,J)+W(I+1,J)))
 1 * DT*(EIGFIF*(DXR*(A-R) + DYR*(C-D))
 2 * RNUA*NXSOR*(W(I+1,J)+U(I+1,J)-U(I,J))
 3 * W(I,J)*(U(I,J)-W(I-1,J)))
 4 * RNUA*NYSOR*(WFAA*(U(I,J)-U(I,J))
 5 * WFAA*(U(I,J)-U(I,J-1)))
 2198 CONTINUE
 2199 CONTINUE
 C
 C
 C
 C CALCULATE TRAR
 C
 DO 2259 J = 2,JBAR
 TRS = 0.
 DO 2254 I = 2,I1
 TRS = T(I,J) + T(I,J+1) + TRS
 2258 CONTINUE
 TRAR(J) = TRS/(2.*(-1.0*FLOAT(I1)))
 2259 CONTINUE
 C
 C
 C
 DO 2299 I = 2,IHP1
 DO 2294 J = 2,JBAR
 VC = .5*(V(I,J-1) + V(I,J))
 IF(VC.GE.0.) E = W(I,J)+VC*V(I,J-1)
 IF(VC.LT.0.) E = W(I,J)+VC*V(I,J)
 VC = .5*(V(I,J) + V(I,J+1))
 IF(VC.GE.0.) F = W(I,J+1)+VC*V(I,J)
 IF(VC.LT.0.) F = W(I,J+1)+VC*V(I,J+1)
 WFAA = .5*AMIN1(W(I-1,J)+W(I-1,J+1)+W(I,J)+W(I,J+1))
 UC = .5*(U(I-1,J) + U(I-1,J+1))
 IF(UC.GE.0.) G = WFAA+UC*V(I-1,J)
 IF(UC.LT.0.) G = WFAA+UC*V(I,J)
 WFAA = .5*AMIN1(W(I,J)+W(I,J+1)+W(I+1,J)+W(I+1,J+1))
 UC = .5*(U(I,J) + U(I,J+1))
 IF(UC.GE.0.) H = WFAA+UC*V(I,J)
 IF(UC.LT.0.) H = WFAA+UC*V(I+1,J)
 WFAAC = .5*AMIN1(W(I,J)+W(I,J+1)+W(I+1,J)+W(I+1,J+1))
 WFAA = .5*AMIN1(W(I,J)+W(I,J+1)+W(I-1,J)+W(I-1,J+1))
 C
 VWBAR(I,J) = (V(I,J)+.5*(W(I,J)+W(I,J+1)))
 1 * DT*(EIGFIF*(DYR*(E-F) + DXR*(G-H))
 2 * .5*(W(I,J)+W(I,J+1))+BETA*GRAV
 3 * (.5*(T(I,J) + T(I,J+1)) - TBAH(J))
 4 * RNUA*NXSOR*(WFAAC*(V(I,J)-V(I,J))
 5 * WFAA*(V(I,J) - V(I-1,J)))
 6 * RNUA*NYSOR*(W(I,J+1)*(V(I,J+1) - V(I,J))
 7 * W(I,J)*(V(I,J) - V(I,J-1)))
 2298 CONTINUE
 2299 CONTINUE

```

000381 000381 C
000382 000382 C
000383 000383 C ***** BEGIN ITERATION *****
000384 000384 C
000385 000385 C
000386 000386 C CALCULATE D (MASS EQUATION)
000387 000387 C
000388 000388     NIT = 0
000389 000389     2500 CONTINUE
000390 000390     DTEST = 0.0
000391 000391     DSUM = 0.0
000392 000392     DO 2599 I = 2,IBP1
000393 000393         DO 2598 J = 2,JRP1
000394 000394             WFACA = AMIN1(WNP1(I,J),WNP1(I+1,J))
000395 000395             WFACB = AMIN1(WNP1(I-1,J),WNP1(I,J))
000396 000396             WFACC = AMIN1(WNP1(I,J),WNP1(I,J+1))
000397 000397             WFACD = AMIN1(WNP1(I,J),WNP1(I,J-1))
000398 000398 C
000399 000399             D(I,J) = TWO*H1*(WFACA*U(I,J)-WFACB*U(I-1,J))
000400 000400             1             *DXR + (WFACC*V(I,J) - WFACD*V(I,J-1))*DYR)
000401 000401             2             + (WNP1(I,J) - W(I,J))*DTH
000402 000402             DARS = ABS(D(I,J))
000403 000403             DTEST = AMAX1(DARS,DTEST)
000404 000404             DSUM = DSUM + D(I,J)
000405 000405 C
000406 000406 C CALCULATE GAMMA
000407 000407 C
000408 000408             GTERM = (2.232*DT)/(T(I,J)-3.8)
000409 000409             IF(WNP1(I,J).EQ.0.0) GO TO 2550
000410 000410             GAM(I,J) = 0.75*(WNP1(I,J)**2) /
000411 000411             1             (WNP1(I,J)**3*DT*(DXSOR*DYSOR))
000412 000412             GO TO 2570
000413 000413     2550 CONTINUE
000414 000414             GAM(I,J) = 0.0
000415 000415     2570 CONTINUE
000416 000416 C
000417 000417 C CALCULATE PHI
000418 000418 C
000419 000419             PHI(I,J) = PHI(I,J) - (GAM(I,J)*D(I,J))
000420 000420     2598 CONTINUE
000421 000421     2599 CONTINUE
000422 000422 C
000423 000423 C TEST FOR CONVERGENCE
000424 000424 C
000425 000425             IF(NIT.EQ.0) GO TO 2650
000426 000426             DEPS = EPSILON*W*V*IN*DYR
000427 000427             IF(DTEST.I. DEPS) GO TO 3000
000428 000428             IF(NIT.GE.1000) GO TO 3000
000429 000429     2650 CONTINUE

```

000430
 000431
 000432
 000433
 000434
 000435
 000436
 000437
 000438
 000439
 000440
 000441
 000442
 000443
 000444
 000445
 000446
 000447
 000448
 000449
 000450
 000451
 000452
 000453
 000454
 000455
 000456
 000457
 000458
 000459
 000460
 000461
 000462
 000463
 000464
 000465
 000466
 000467
 000468
 000469
 000470
 000471
 000472
 000473
 000474
 000475
 000476
 000477
 000478
 000479
 000480
 000481
 000482

C
 C NO CONVERGENCE
 C
 C
 C
 C CALCULATE U AND V
 C
 DO 2499 I = 2,IBAR
 DO 2499 J = 2,JBP1
 GTERMU = 2.232*NT / (.5*(T(I,J)+T(I+1,J))-3.8)
 WFACU = .5*(WNP1(I,J)+WNP1(I+1,J))
 WMINU = AMIN1(WNP1(I,J),WNP1(I+1,J))
 U(I,J) = WMINU*(UWBAR(I,J)+WMINU*DT*DX*(PHI(I,J)
 -PHI(I+1,J))) / (WMINU*WFACU+GTERMU)
 1
 2698 CONTINUE
 2699 CONTINUE
 C
 DO 2799 I = 2,IBP1
 DO 2799 J = 2,JBAR
 GTERMV = 2.232*NT / (.5*(T(I,J)+T(I,J+1))-3.8)
 WFACV = .5*(WNP1(I,J)+WNP1(I,J+1))
 WMINV = AMIN1(WNP1(I,J),WNP1(I,J+1))
 V(I,J) = WMINV*(VWBAR(I,J)+WMINV*DT*DY*(PHI(I,J)
 -PHI(I,J+1))) / (WMINV*WFACV+GTERMV)
 1
 2798 CONTINUE
 2799 CONTINUE
 NIT = NIT + 1
 GO TO 2500
 C
 C ***** END ITERATION *****
 C
 C
 C CONVERGENCE ATTAINED, CONTINUE CYCLE
 C
 3000 CONTINUE
 C
 C CALCULATE NEW TEMPERATURES (WATER TEMPERATURE CONVECTION)
 C
 DO 4199 I = 2,IBP1
 DO 4199 J = 2,JBP1
 WFAC = AMIN1(WNP1(I,J),WNP1(I-1,J))
 UC = U(I-1,J)
 IF(UC.GE.0.0) AA = WFAC*UC*T(I-1,J)
 IF(UC.LT.0.0) AA = WFAC*UC*T(I,J)
 WFAC = AMIN1(WNP1(I,J),WNP1(I+1,J))
 UC = U(I,J)
 IF(UC.GE.0.0) BB = WFAC*UC*T(I,J)
 IF(UC.LT.0.0) BB = WFAC*UC*T(I+1,J)
 WFAC = AMIN1(WNP1(I,J),WNP1(I,J-1))
 VC = V(I,J-1)
 IF(VC.GE.0.0) CC = WFAC*VC*T(I,J-1)
 IF(VC.LT.0.0) CC = WFAC*VC*T(I,J)
 WFAC = AMIN1(WNP1(I,J),WNP1(I,J+1))
 VC = V(I,J)

```

000483 000483 IF (VC.GE.0.0) Dn = WFAC*VC*T(I,J)
000484 000484 IF (VC.LT.0.0) Dn = WFAC*VC*T(I,J+1)
000485 000485 C
000486 000486 HC(I,J) = DT*(TWO*THI*DXR*(AA-BB) + TWO*THI*DYR*(CC-DD))
000487 000487 I * W(I,J)*T(I,J) + T(I,J)*D(I,J)*DT
000488 000488 4198 CONTINUE
000489 000489 4199 CONTINUE
000490 000490 C
000491 000491 C
000492 000492 AKAPTERM = 2.0*AKAP0*DT*DZR
000493 000493 DO 4299 I = IIP1,IRP1
000494 000494 DO 4299 J = 2,JBPI
000495 000495 T(I,J) = 1.0 / (WNP1(I,J) + AKAPTERM)*(HC(I,J) + AKAPTERM
000496 000496 *XI(I,J,2))
000497 000497 4298 CONTINUE
000498 000498 4299 CONTINUE
000499 000499 C
000500 000500 C
000501 000501 IF (II.LT.4) GO TO 4400
000502 000502 DO 4399 I = 3,IIPI
000503 000503 DO 4399 J = 2,JBPI
000504 000504 T(I,J) = HC(I,J) / WNP1(I,J)
000505 000505 4398 CONTINUE
000506 000506 4399 CONTINUE
000507 000507 4400 CONTINUE
000508 000508 C
000509 000509 C
000510 000510 AKTERM1 = 1.0 + AKAP0*DT*DXSQ
000511 000511 AKTERM2 = AKAP0*DT*DY*DSQ
000512 000512 DO 4499 J = 2,JBPI
000513 000513 SUMT = 0.0
000514 000514 DO 4499 K = 1,KRPI
000515 000515 SUMT = SUMT + XI(IIP1,J,K) + XI(IIP1,J,K+1) + XI(IIP2,J,K)
000516 000516 I * XI(IIP2,J,K+1)
000517 000517 4498 CONTINUE
000518 000518 T(2,J) = 1.0 / (WNP*AKTERM1)*(HC(2,J) + AKTERM2*SUMT)
000519 000519 T(11,J) = 1.0 / (WNP*AKTERM1)*(HC(11,J) + AKTERM2*SUMT)
000520 000520 4499 CONTINUE
000521 000521 C
000522 000522 C INCREMENT TIME AND CYCLE
000523 000523 C
000524 000524 NCYCLE = NCYCLE + 1
000525 000525 TI = TI + DT
000526 000526 GO TO 900

```


000527	000527	C
000528	000528	C END CYCLE
000529	000529	C
000530	000530	C
000531	000531	C
000532	000532	C
000533	000533	5100 CONTINUE
000534	000534	C
000535	000535	C CALL PRINT AND PLOT ROUTINES
000536	000536	C
000537	000537	CALL PRINT
000538	000538	CALL PLOTPKG
000539	000539	C
000540	000540	C CHECK FOR DONE
000541	000541	C
000542	000542	IF (T1.0E+TF) GO TO 7000
000543	000543	TP = TP + TPI
000544	000544	GO TO 1100
000545	000545	C
000546	000546	C
000547	000547	7000 CONTINUE
000548	000548	CALL DROP
000549	000549	CALL NNTIM(ISEC)
000550	000550	CPTIME = FLOAT(ISEC) / 1.E6
000551	000551	WRITE (6,9100) CPTIME
000552	000552	CALL EXIT(1)
000553	000553	C
000554	000554	C
000555	000555	9100 FORMAT(17.10HSECONDS USED =.1PE14.5)
000556	000556	C
000557	000557	END

CHAT 1670 09159124 504/02/79

```

000001      000558      C
000002      000559      C
000003      000560      C
000004      000561
000005      000562      C
000006      000563      C
000007      000564
000008      000565
000009      000566
000010      000567      COMMON/ARRAYS/GAM(NI,NJ),HC(NI,NJ),PHI(NI,NJ),T(NI,NJ),
000011      000568      1      UWBAR(NI,NJ),VWBAR(NI,NJ),H(NI,NJ),
000012      000569      2      VINI,NJ),WINI,NJ),WNP1(NI,NJ),XI(NI,NJ,NK),
000013      000570      3      XINP1(NI,NJ,NK),D(NI,NJ),INP1(NI,NJ),TRAN(NJ)
000014      000571      C
000015      000572      C
000016      000573      COMMON/VAR/AKAPO,AKAPR,BETA,BETAR,DT,DTR,DX,DY,DZ,DXR,DYR,DZR,
000017      000574      1      NXSQ,DYSQ,NZSQ,NXSQR,DYSQR,NZSQR,FIGFIF,EPSILON,GRAV,
000018      000575      2      IL,IIP1,IIP2,IIM1,IRAR,IRP1,IRP2,JRAR,JBP1,JBP2,
000019      000576      3      KBAR,KBP1,KRM1,NCYCLE,NIT,RNUA,TI,TF,TP,
000020      000577      4      TPC,TPI,THICK,TIN,TWOHI,VIN,VOUT,W0,WP,XII,
000021      000578      5      BR,RW,RHOR,RHOW,AKR,HEIGHT,DSUM
000022      000579      C
000023      000580      C
000024      000581      C
000025      000582      C CALCULATE ENERGY INFLOW RATE THROUGH OUTLET
000026      000583      C
000027      000584      EDOTIN = FLOAT(II-1)*DX*WP*RHOW*RW*VIN
000028      000585      C
000029      000586      C
000030      000587      C CALCULATE ENERGY OUTFLOW RATE THROUGH THE OUTLET
000031      000588      C
000032      000589      SUM = 0.0
000033      000590      DO 5299 I = 2,II
000034      000591      SUM = SUM + T(II,JBP1)
000035      000592      5299 CONTINUE
000036      000593      EDOTOUT = WP*RHOW*BW*NX*VOUT*SUM
000037      000594      EDOTDIF = EDOTOUT - EDOTIN
000038      000595      C
000039      000596      C PRINT STATEMENTS
000040      000597      C
000041      000598      WRITE(6,9000) TI,NCYCLE,NIT
000042      000599      WRITE(6,9002)
000043      000600      WRITE(6,9005) EDOTIN,EDOTOUT,EDOTDIF,DSUM
000044      000601      WRITE(6,9002)
000045      000602      IF(NCYCLE.GT.0.0) GO TO 5310
000046      000603      WRITE(6,9010) RW,BR,RHOW,RHOR,AKR
000047      000604      WRITE(6,9012) AKAPR,AKAPO,BETA,BETAR,DT,DTR
000048      000605      WRITE(6,9015) NX,DY,DZ,DXR,DYR,DZR
000049      000606      WRITE(6,9017) NXSQ,DYSQ,NXSQR,DYSQR,NZSQ
000050      000607      WRITE(6,9020) FIGFIF,EPSILON,GRAV,IL,IBAR,JBAR,KBAR
000051      000608      WRITE(6,9022) RNUA,TI,TF,TP,TPC,TPI

```

000052 000609
 000053 000610
 000054 000611
 000055 000612
 000056 000613
 000057 000614
 000058 000615
 000059 000616
 000060 000617
 000061 000618
 000062 000619
 000063 000620
 000064 000621
 000065 000622
 000066 000623
 000067 000624
 000068 000625
 000069 000626
 000070 000627
 000071 000628
 000072 000629
 000073 000630
 000074 000631
 000075 000632
 000076 000633
 000077 000634
 000078 000635
 000079 000636
 000080 000637
 000081 000638
 000082 000639
 000083 000640
 000084 000641
 000085 000642
 000086 000643
 000087 000644
 000088 000645
 000089 000646
 000090 000647
 000091 000648
 000092 000649
 000093 000650
 000094 000651
 000095 000652
 000096 000653
 000097 000654
 000098 000655
 000099 000656
 000100 000657
 000101 000658
 000102 000659
 000103 000660
 000104 000661

```

WRITE(6,9025) THICK,TIN,TWOTH,VIN,HFIGHT
WRITE(6,9027) W0WP,X11,NI,NJ,NK
5310 CONTINUE
C
WRITE(6,9040)
WRITE(6,9035)
WRITE(6,9040)
DO 5399 I = 1,IP2
  DO 5399 J = 1,JBP2
    WRITE(6,9045) I,J,U(I,J),V(I,J),PHI(I,J),T(I,J),WNP1(I,J),
      GAM(I,J),D(I,J),XI(I,J,2),XI(I,J,3),XI(I,J,4)
  1
5398 CONTINUE
WRITE(6,9040)
5399 CONTINUE
C
C PRINT OUT CURRENT TIME ON TELETYPE WHEN RUNNING
C
PRINT 9110,TI
RETURN
C
C
C
C
C FORMATS
C
9000 FORMAT(1H1,25X,6HTIME=.1PF12.5,10X,8HNCYCLE=.16,10X,5HUNIT=.16)
9005 FORMAT(7HEDOTIN=.E9,2,5X,7HEDOTOUT=.E9,2,5X,7HEDOTDIF=.E9,2,
  1 5X,4HDSUM=.E9,2)
9010 FORMAT(3HRAW=.F9,2,5X,1HBR=.E9,2,5X,5HROW=.E9,2,3X,5HROW=.
  1 E9,2,3X,4HAKR=.E9,2)
9012 FORMAT(6HAKAPR=.E9,2,2X,6HAKAPR=.E9,2,2X,5HBTAR=.E9,2,3X,
  1 6HBTAR=.E9,2,5X,3HNT=.E9,2,5X,4HUTR=.F9,2)
9015 FORMAT(3HDX=.F9,2,5X,3HDX=.E9,2,5X,3HDX=.E9,2,5X,4HDXR=.F9,2,
  1 4X,4HDXR=.E9,2,4X,4HDXR=.E9,2)
9017 FORMAT(5HDXSQ=.E9,2,3X,5HDXSQ=.E9,2,3X,6HDXSQ=.E9,2,2X,
  1 6HDXSQ=.E9,2,2X,6HDXSQ=.E9,2)
9020 FORMAT(7HIFIF=.E9,2,1X,8HPSILON=.E9,2,1X,5HGRAV=.E9,2,
  1 2X,1HIL=.14,10X,5HIRAR=.14,8X,5HJBAR=.14,2X,5HKBAR=.14)
9022 FORMAT(5HNUA=.E9,2,3X,3HTJ=.E9,2,5X,3HTF=.E9,2,5X,3HTP=.
  1 E9,2,5X,4HTPC=.E9,2,4X,4HTPI=.F9,2)
9025 FORMAT(6HTHICK=.E9,2,2X,4HTIN=.E9,2,4X,7HTWOTH=.E9,2,1X,
  1 4HTIN=.E9,2,4X,7HWEIGH=.E9,2)
9027 FORMAT(3HWN=.E9,2,5X,3HWP=.E9,2,5X,4HXI=.E9,2,4X,3HNI=.16,
  1 8X,3HNJ=.16,8X,3HNK=.16)
9035 FORMAT(1HACELL I J,8X,1HU,10X,1HV,9X,3HPI,9X,1HT,8X,4HWNPI,
  1 1X,1HGAM,9X,1HD,4X,9HXI(I,J,2),2X,9HXI(I,J,3),2X,
  2 9HXI(I,J,4))
9040 FORMAT(//)
9042 FORMAT(///)
9045 FORMAT(2I4,2X,10(1PE10.3,1X))
9110 FORMAT(5HCURRENT TIME IS E,1PE12.5,/)
C
END

```

1*****

000001 0006A2
 000002 0006A3
 000003 0006A4
 000004 0006A5
 000005 0006A6
 000006 0006A7
 000007 0006A8
 000008 0006A9
 000009 000670
 000010 000671
 000011 000672
 000012 000673
 000013 000674
 000014 000675
 000015 000676
 000016 000677
 000017 000678
 000018 000679
 000019 0006A0
 000020 0006A1
 000021 0006A2

C

C

C

CHAT 167D 09159124 504/02/79

SUBROUTINE UNDROP

LCM (BINJIN) = (FLMBUFC)
 COMMON /BINJLN/ IBSIZE,IBUFF(5000)
 COMMON /FIMBUFC/ FLMBUF(5000)
 DIMENSION IHEAD(6)
 IBSIZE = 4000
 CALL FILMRO
 CALL IDENTRO (12,FLMBUF,5000,1100000,105,0)
 IHEAD(1) = 10W T3 RUTH
 IHEAD(2) = 10HREO 48
 IHEAD(3) = 10HPSIL=1.-3
 IHEAD(4) = 10HWO=0.0
 IHEAD(5) = 10HNT=.1
 IHEAD(6) = 10HVIN = 0.1
 CALL HEADRO (12,60,IHEAD,10HBOX T3RUTH)
 CALL KEEPFLM (12)
 RETURN
 END

00001 0006A3
 00002 0006A4
 00003 0006A5
 00004 0006A6
 00005 0006A7
 00006 0006A8
 00007 0006A9
 00008 000690
 00009 000691
 00010 000692
 00011 000693

C

C

C

CHAT 167D 09159124 504/02/79

SUBROUTINE DROP

LCM (FLMBUFC)
 COMMON /FIMBUFC/ FLMBUF(5000)
 CALL FILMRO
 CALL IDENTRO (12,FLMBUF,5000,1100000,105,0)
 CALL KEEPFLM(12)
 RETURN
 END

1*****

000001 000694
000002 000695
000003 000696
000004 000697
000005 000698
000006 000699
000007 000700
000008 000701
000009 000702
000010 000703
000011 000704
000012 000705
000013 000706
000014 000707
000015 000708
000016 000709
000017 000710
000018 000711
000019 000712
000020 000713
000021 000714
000022 000715
000023 000716
000024 000717
000025 000718
000026 000719
000027 000720
000028 000721
000029 000722
000030 000723
000031 000724
000032 000725
000033 000726
000034 000727
000035 000728
000036 000729
000037 000730
000038 000731
000039 000732
000040 000733
000041 000734
000042 000735
000043 000736
000044 000737
000045 000738
000046 000739
000047 000740
000048 000741
000049 000742
000050 000743

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

CHAT 167D 09159:24 504/02/79

SUBROUTINE PLOTPKG

PARAMETER (NI=23)

PARAMETER (NJ=23)

PARAMETER (NK=10)

COMMON/ARRAYS/RAH(NI,NJ),HC(NI,NJ),PHI(NI,NJ),T(NI,NJ),

1 UWBAR(NI,NJ),VWBAR(NI,NJ),U(NI,NJ),

2 V(NI,NJ),W(NI,NJ),WNP1(NI,NJ),XI(NI,NJ,NK),

3 XINP1(NI,NJ,NK),D(NI,NJ),INP1(NI,NJ),TBAR(NJ)

COMMON/VAR/AKAP0,AKAP0,BETA,BETAR,DT,OTR,DX,DY,DZ,DXR,DYR,DZR,

1 DXSQ,DYSQ,DZSQ,DXSQR,DYSQR,DZSQR,FIGFIF,EPSILON,GRAV,

2 I1,I1P1,I102,I1M1,IRAR,IRP1,IRP2,JRAR,JBP1,JRP2,

3 KBAR,KBP1,KRM1,NCYCLE,NIT,RNUA,TI,TF,TP,

4 TPC,TPI,THICK,TIN,TWOTH1,VIN,VOUT,W0,WP,XII,

5 RR,RW,RHOW,RHOW,AKR,HEIGHT,DSUM

COMMON/CJF07/IXL,IXR,IYT,IYR,XMN,XXK,YMA,YMN

DIMENSION CON(25),BCD(2),TIME(4),VAVE(2),X(1),Y(1)

EQUIVALENCE (UWBAR(1,1),X(1)),(VWBAR(1,1),Y(1))

C CONTOUR PLOTS

IPLT=i

IFINISH=1

C SET COORDINATES FOR CELL CENTERED QUANTITIES

C (THE BOUNDARIES ARE UNCHANGED.)

C

95 CONTINUE

X(1)=0.

Y(1)=0.

X(2) = .5*DX

Y(2) = .5*DY

DO 100 I = 2,IRAR

X(I+1) = X(I) + DX

100 CONTINUE

C

DO 120 J = 2,JRAR

Y(J+1) = Y(J) + DY

120 CONTINUE

X(IRP2)=X(IRP1)+.5*DX

Y(IRP2)=Y(IRP1)+.5*DY

GO TO (125,155,195,215),IPI,T

```

000051 000744 C
000052 000745   125 CONTINUE
000053 000746 C
000054 000747 C LABELS AND NUMBER OF CHARACTERS FOR AXES
000055 000748   BCD(1) = AHX-AXIS
000056 000749   BCD(2) = AHY-AXIS
000057 000750   NXN = 6
000058 000751   NYN = 6
000059 000752 C
000060 000753 C NO. OF CHARACTERS FOR FRAME TITLE
000061 000754   MZL = 40
000062 000755 C
000063 000756 C PLOT M
000064 000757 C
000065 000758 C WRITE LARGE CHARACTERS FOR TIME (REPLACES ENCODE)
000066 000759 C
000067 000760   WRITE(63,900) T1
000068 000761   REWIND 63
000069 000762   READ(63,901) (TIME(I),I=1,4)
000070 000763   901 FORMAT(4A10)
000071 000764   900 FORMAT(1X,=CRACK WIDTH AT T=,F12.5,10X)
000072 000765 C
000073 000766 C STORE CRACK WIDTH IN DUMMY ARRAY HC SUCH THAT THE PRIMARY CRACK
000074 000767 C IS EXCLUDED. (IT IS THE LAST COLUMN IN THE PRIMARY CRACK.)
000075 000768   JM = 5*JRP
000076 000769   DO 150 I = 2,JRP
000077 000770   DO 150 J = 2,JRP
000078 000771   HC(I,J) = WNP1(I,J)
000079 000772   IF(I,LF,1) HC(I,J) = WNP1(IIP1,JM)
000080 000773   150 CONTINUE
000081 000774 C
000082 000775 C SET UPPER AND LOWER BOUNDARIES IN THE PRIMARY CRACK TO
000083 000776 C ADJACENT INNER VALUES
000084 000777 C SET UPPER AND LOWER BOUNDARIES TO ZERO OUTSIDE PRIMARY CRACK
000085 000778   DO 153 I = 1,IBP2
000086 000779   IF(I,GT,1) GO TO 152
000087 000780   HC(I,1) = WNP1(IIP1,JM)
000088 000781   HC(I,JRP2) = WNP1(IIP1,JM)
000089 000782   GO TO 153
000090 000783   152 CONTINUE
000091 000784   HC(I,1) = 0.
000092 000785   HC(I,JRP2) = 0.
000093 000786   153 CONTINUE
000094 000787 C
000095 000788 C SET LEFT BOUNDARY WIDTH TO ADJACENT INNER
000096 000789 C SET RIGHT BOUNDARY WIDTH TO 0.
000097 000790   DO 154 J = 1,JRP2
000098 000791   HC(1,J) = WNP1(IIP1,JM)
000099 000792   HC(IIP2,J) = 0.
000100 000793   154 CONTINUE
000101 000794   CALL CONTRJ(X,=IBP2,Y,=JRP2,HC(1,1),NI,NJ,10,0,0,0,=1,
000102 000795   1 CON,DX,DY,1,TIME,MZL,6HX=AXIS,6,AHY=AXIS,6)

```

```

000103 000796      IPLT=IPLT.I
000104 000797      GO TO 325
000105 000798      C
000106 000799      C PLOT T
000107 000800      C
000108 000801      155 CONTINUE
000109 000802      REWIND 63
000110 000803      WRITE(63,9010) TI
000111 000804      REWIND 63
000112 000805      READ(63,9011) (TIME(I),I=1,4)
000113 000806      910 FORMAT(1X,=WATER TEMP AT T= E.F)2.5,12X)

000114 000807      C
000115 000808      C STORE WATER TEMPERATURE IN DUMMY ARRAY
000116 000809      DO 180 I=2,IBP1
000117 000810      DO 180 J=2,JBP1
000118 000811      HC(I,J)=T(I,J)
000119 000812      180 CONTINUE
000120 000813      C
000121 000814      C SET UPPER AND LOWER BOUNDARY TEMPERATURES OF PRIMARY CRACKS TO
000122 000815      C ADJACENT INNER
000123 000816      C SET UPPER AND LOWER BOUNDARY TEMPERATURES OUTSIDE PRIMARY CRACK TO
000124 000817      C TEMP AT INFINITY
000125 000818      HC(1,2) = HC(2,2)
000126 000819      HC(1BP2,2) = XII
000127 000820      HC(1,JBP1) = HC(2,JBP1)
000128 000821      HC(1BP2,JBP1) = XII
000129 000822      DO 183 I=1,IBP2
000130 000823      IF (I.GT.1) GO TO 182
000131 000824      HC(I,1)=HC(I,2)
000132 000825      HC(I,JBP2)=HC(I,JBP1)
000133 000826      GO TO 183
000134 000827      182 CONTINUE
000135 000828      HC(I,1)=XII
000136 000829      HC(I,JBP2)=XII
000137 000830      183 CONTINUE
000138 000831      C
000139 000832      C SET LEFT BOUNDARY TEMPERATURES TO ADJACENT INNER
000140 000833      C SET RIGHT BOUNDARY TEMPERATURES TO TEMP AT INFINITY
000141 000834      DO 184 J=1,JBP2
000142 000835      HC(1,J)=HC(2,J)
000143 000836      HC(1BP2,J)=XII
000144 000837      184 CONTINUE
000145 000838      C
000146 000839      C SET SPECIAL CONTOUR MATRIX
000147 000840      NC = -10
000148 000841      NOC = -NC
000149 000842      CALL MINM(HC,N1,IBP2,JBP2,1,J,7MIN)
000150 000843      CALL MAXM(HC,N1,IBP2,JBP2,1,J,7MAX)
000151 000844      DELZ = (ZMAX - ZMIN) / (NOC-1.)
000152 000845      C
000153 000846      C REMOVE BOUNDARY VALUES
000154 000847      ZMIN = ZMIN + DELZ
000155 000848      ZMAX = ZMAX - DELZ

```

```

000156 000849      DELZ = (ZMAX-ZMIN) / (NOC-1.)
000157 000850      CON(1) = 7MIN
000158 000851      DO 189 I = 2,NOC
000159 000852          CON(I) = CON(I-1) + DELZ
000160 000853      189 CONTINUE
000161 000854      C
000162 000855      192 CONTINUE
000163 000856      CALL CONTRJR(X,IRP2,Y,JRP2,HC(1,1),NI,NJ,NC,n..0..DELZ,
000164 000857      1      CON,DX,DY,1,TIME,MZL,6HX-AXIS,6,6HY-AXIS,6)
000165 000858      IPLT=IPLT+1
000166 000859      GO TO 325
000167 000860      C
000168 000861      C PLOT PHI
000169 000862      C
000170 000863      195 CONTINUE
000171 000864      REWIND 63
000172 000865      WRITE(63,920) T1
000173 000866      REWIND 63
000174 000867      READ(63,901) (TIME(I),I=1,4)
000175 000868      920 FORMAT(1X,=MODIFIED PRESSURE AT T= E.E12.5+15X)
000176 000869      C
000177 000870      C
000178 000871      C USE HC ARRAY TO PLOT PHIS. THIS ELIMINATES PLOTTING
000179 000872      C ZERO PRESSURES
000180 000873      C
000181 000874      DO 200 I = 2,IRP1
000182 000875      DO 200 J = 2,JRP1
000183 000876          HC(I,J) = PHI(I,J)
000184 000877          IF (WNP1(I,J).LT.W0) HC(I,J)=HC(I-1,J)
000185 000878      200 CONTINUE
000186 000879      C
000187 000880      C SET UPPER AND LOWER BOUNDARY PRESSURES TO ADJACENT INNER
000188 000881      HC(1,2) = HC(2,2)
000189 000882      HC(IRP2,2) = HC(IRP1,2)
000190 000883      HC(1,JRP1) = HC(2,JRP1)
000191 000884      HC(IRP2,JRP1) = HC(IRP1,JRP1)
000192 000885      DO 202 I=1,IRP2
000193 000886          HC(I,1)=HC(I,2)
000194 000887          HC(I,JRP2)=HC(I,JRP1)
000195 000888      202 CONTINUE
000196 000889      C
000197 000890      C SET LEFT AND RIGHT BOUNDARY PRESSURES TO ADJACENT INNER
000198 000891      DO 203 J=3,JRP1
000199 000892          HC(1,J)=HC(2,J)
000200 000893          HC(IRP2,J)=HC(IRP1,J)
000201 000894      203 CONTINUE
000202 000895      CALL CONTRJR(X,-IRP2,Y,-JRP2,HC(1,1),NI,NJ,1n.0..0..-1..
000203 000896      1      CON,DX,DY,1,TIME,MZL,6HX-AXIS,6,6HY-AXIS,6)
000204 000897      IPLT=IPLT+1
000205 000898      GO TO 325

```



```

000206      000899      C
000207      000900      C
000208      000901      C
000209      000902      C
000210      000903      C VECTOR PLOTS
000211      000904      C
000212      000905      215 CONTINUE
000213      000906      IFINISH=2
000214      000907      C
000215      000908      C NORMALIZATION
000216      000909      C
000217      000910      UXAX = 0.
000218      000911      DO 300 I = 2,JRAR
000219      000912      DO 300 J = 2,JRAR
000220      000913      VAVE(1) = .5*(U(I+1,J)+U(I,J))
000221      000914      UTEST = ABS(VAVE(1))
000222      000915      IF(UTEST.GT.UMAX) UMAX = UTEST
000223      000916      300 CONTINUE
000224      000917      C
000225      000918      VMAX = 0.
000226      000919      DO 310 I = 2,JRAR
000227      000920      DO 310 J = 2,JRAR
000228      000921      VAVE(2) = .5*(V(I+1,J)+V(I,J))
000229      000922      VTEST = ABS(VAVE(2))
000230      000923      IF(VTEST.GT.VMAX) VMAX = VTEST
000231      000924      310 CONTINUE
000232      000925      C
000233      000926      C VELF IS MAXIMUM VELOCITY COMPONENT IN SYSTEM
000234      000927      C PICK OUT THE MAXIMUM VELOCITY COMPONENT IN SYSTEM
000235      000928      C
000236      000929      VELF = UMAX
000237      000930      IF(VMAX.GT.UMAX) VELF = VMAX
000238      000931      DF = DX
000239      000932      IF(DX.GT.DY) DF = DY
000240      000933      VNORM = .35
000241      000934      VELF = VNORM*DF/VELF
000242      000935      C
000243      000936      C SET UP FRAME AND LABEL
000244      000937      C
000245      000938      C
000246      000939      C WRITE LARGE CHARACTERS FOR TIME
000247      000940      C
000248      000941      REWIND 63
000249      000942      WRITE(63,930) TI
000250      000943      REWIND 63
000251      000944      READ(63,941) (TIME(I),I=1,4)
000252      000945      930 FORMAT(1X,=VELOCITY AT T= .E12.5.10X)
000253      000946      C
000254      000947      CALL ADV(T)
000255      000948      CALL FRAMP(IXL,IXR,IYT,IYR)
000256      000949      CALL DOA(IXL,IXR,IYT,IYB,XMN,XXM,YMX,MYN)

```

69

```

000257 000950
000258 000951 C CENTER THE TITLE AND THE LABELS FOR THE VECTOR PLOTS
000259 000952 IX=MAX0(4A,IXL,(IXR-IXL-12*MZL)/2)
000260 000953 LZL=MAX0(MZL-(1024-IX)/12,0)+1
000261 000954 IF(LZL.GT.1) LZL=(LZL+R3)/R5+1
000262 000955 CALL DLCH,IX,MAX0(10,ITY-24-16*LZL),MZL,TIME(1),2)
000263 000956 CALL DLCH,MAX0(54,IXL+(IXR-IXL-15*NXN)/2),IYR,26,NXN,BCD(1),2)
000264 000957 CALL DLCH,78,MIND(IYB,52,IYR-(IYR-ITY-12*NYN)/2),NYN,BCD(2),2)
000265 000958
000266 000959 C CALCULATE VELOCITY VECTORS
000267 000960 C
000268 000961 DO 320 J = 2,JRP1
000269 000962 Y1 = Y(J)
000270 000963 DO 320 I = 2,IRP1
000271 000964 X1 = X(I)
000272 000965 X2 = X1+VELF*(U(I-1,J)+U(I,J))
000273 000966 Y2 = Y1+VELF*(V(I,J-1)+V(I,J))
000274 000967 CALL CONVRT,X1,IX1,XMN,XX,XL,IXR)
000275 000968 CALL CONVRT,Y1,IY1,YMN,YM,X,YR,IYT)
000276 000969 CALL CONVRT,X2,IX2,XMN,XX,XL,IXR)
000277 000970 CALL CONVRT,Y2,IY2,YMN,YM,X,YR,IYT)
000278 000971 CALL DRV,IX1,IY1,IX2,IY2)
000279 000972 320 CONTINUE
000280 000973 C
000281 000974 C DETERMINE MAX SPEED AND ZONE
000282 000975 VMX = 0.
000283 000976 IMX = 1
000284 000977 JMX = 1
000285 000978 DO 342 I = 2,IRP1
000286 000979 DO 342 J = 2,JRP1
000287 000980 VAVE(1) = .5*(U(I-1,J) + U(I,J))
000288 000981 VAVE(2) = .5*(V(I,J-1) + V(I,J))
000289 000982 VAVE(1)=VAVE(1)+VAVE(1)
000290 000983 VAVE(2) = VAVE(2)+VAVE(2)
000291 000984 VTEST = SURT(VAVE(1)+VAVE(2))
000292 000985 IF(VMX.GT.VTEST) GO TO 342
000293 000986 VMX = VTEST
000294 000987 IMX = I
000295 000988 JMX = J
000296 000989 342 CONTINUE
000297 000990 C
000298 000991 C IDENTIFY MAX SPEED AND ZONE
000299 000992 REWIND 63
000300 000993 WRITE(63,940) VMX,IMX,JMX
000301 000994 REWIND 63
000302 000995 READ(A3,941) (TIME(I),I=1,4)
000303 000996 CALL DLCH,116,1000,60,TIME(1),2)
000304 000997 940 FORMAT(1X,19HMAXIMUM SPEED,I,J) ,1PE12.5,2I4)
000305 000998 941 FORMAT(1X,A10)

```

```

000306      000999      C
000307      001000      C DRAW DOTTED LINE BETWEEN PRIMARY FRACTURE AND
000308      001001      C THERMAL STRESS FRACTURE
000309      001002      C
000310      001003      C SET UP X COORDINATES
000311      001004      325 CONTINUE
000312      001005      XF = X(I1)*.50X
000313      001006      CALL CONVRT(XF,IX1,XMN,XX,IXL,IXR)
000314      001007      IX2 = IX1
000315      001008      C
000316      001009      C SET UP Y COORDINATES
000317      001010      JDOT = 5
000318      001011      DYDOT = DY/JDOT
000319      001012      YL = NYDOT*.5
000320      001013      JDOTOT = JNAR*JDOT+.1
000321      001014      Y(I1) = 0.
000322      001015      DO 330 J = 1,JDOTOT
000323      001016      Y(J+1) = Y(I1)+DYDOT
000324      001017      330 CONTINUE
000325      001018      C
000326      001019      C PLOT DOTS
000327      001020      DO 340 J = 1,JDOTOT
000328      001021      Y1 = Y(J)
000329      001022      Y2 = Y(J)+YI
000330      001023      CALL CONVRT(Y1,IY1,YMN,YMX,IYR,IYT)
000331      001024      CALL CONVRT(Y2,IY2,YMN,YMX,IYR,IYT)
000332      001025      CALL DRV(IX1,IY1,IX2,IY2)
000333      001026      340 CONTINUE
000334      001027      GO TO (95,145),IFINISH
000335      001028      C
000336      001029      345 RETURN
000337      001030      END

```

CHAT 167D 09159124 504/02/79

1*****

```

000001      001031      C
000002      001032      C
000003      001033      SUBROUTINE PLOJ(B(IX,Y,NPTS,INC,LNN,NSYM,C,XAA,YAA,LABELZ,NZL,LABELX
000004      001034      1,NXL,LABELY,NYL)
000005      001035      C
000006      001036      C LABEL AXES, WRITE TITLES, DRAW BORDERS AND CONNECT CONTOUR POINTS
000007      001037      C ESTABLISHED IN TRICJR
000008      001038      COMMON /CJEOT/ IXL,IXR,IYT,IYR,XMN,XX,XX,YMN,YMN
000009      001039      DIMENSION X(1), Y(1)
000010      001040      INTEGER GRIDF
000011      001041      B=AMAX1(AMAX1(C,0.1)*(LNN+1),0.)
000012      001042      LIN=LNN
000013      001043      KSYM=IARS(NSYM)
000014      001044      KINC=MAX0(IABS(INC),1)
000015      001045      C
000016      001046      C ESTABLISH NO. CHARACTERS AND NO. POINTS
000017      001047      MPTS=IARS(NPTS)
000018      001048      NZL=IABS(NZL)
000019      001049      XAA=ARS(XAA)
000020      001050      YAA=ARS(YAA)

```

```

000021 001051 NXN=YARS(NXL)
000022 001052 NYN=YARS(NYL)
000023 001053 GRIDF=AMAX1(1.,ABS(C))
000024 001054 IF (NSYM.LT.0) GO TO 130
000025 001055 C
000026 001056 C ADVANCE FRAME AND DETERMINE MAX AND MIN FOR AXES
000027 001057 CALL ADV (1)
000028 001058 CALL MAXV (X,KINC,MPTS,ISUR,XXM)
000029 001059 CALL MAXV (Y,KINC,MPTS,ISUR,XXM)
000030 001060 CALL MINV (X,KINC,MPTS,ISUR,XXM)
000031 001061 CALL MINV (Y,KINC,MPTS,ISUR,XXM)
000032 001062 C
000033 001063 C IF XXA AND/OR YYA ARE NON-ZERO THE LENGTHS
000034 001064 C WILL BE CONSIDERED AS RATIOS WHERE THE LONGEST
000035 001065 C SIDE IS FITTED ON A 90 DEGREE POINT LINE.
000036 001066 C DETERMINE ASPECT RATIO OF BORDERS
000037 001067 IF (XXA.EQ.0) XXA=6.
000038 001068 IF (YYA.EQ.0) YYA=10.
000039 001069 DXM=.001*ABS(XXM)
000040 001070 IF (DXM.EQ.0) DXM=.0001
000041 001071 XMN=XXM-DXM
000042 001072 XMX=XMX+DXM
000043 001073 60 IF (SIGN(1,NY),LT.0.AND,INC,GT.0) YYA=(YMX-VMN)/YYA
000044 001074 IF (SIGN(1,NXL),LT.0.AND,NPTS,GT.0) XXA=(XMX-XMN)/XXA
000045 001075 C
000046 001076 C DETERMINE 4020 COORDINATES OF BORDER
000047 001077 IXL=120
000048 001078 IYB=910
000049 001079 FACT=.60./AMAX1(XXA,YYA)
000050 001080 IXR=120+IFIX(FACT*XXA)
000051 001081 IYT=910-IFIX(FACT*YYA)
000052 001082 CALL FRAME (IXL,IXR,IYT,IYB)
000053 001083 IF (SIGN(1.,XAA),GT.0) GO TO 70
000054 001084 SWAP=XXM
000055 001085 XMN=XMX
000056 001086 XMX=SWAP
000057 001087 70 IF (SIGN(1.,YAA),GT.0) GO TO 80
000058 001088 SWAP=VMN
000059 001089 YMN=YMX
000060 001090 YMX=SWAP
000061 001091 80 CALL DGA (IXL,IXR,IYT,IYB,XMN,XMX,YMX,VMN)
000062 001092 C
000063 001093 C CENTER TITLE AND LABEL AXES
000064 001094 IF (NXN.NF.0) CALL DICH (MAX0(54,(IXR-IXL-12*NXN)/2),IYB+26
000065 001095 1,NXN,1,ARELY,5)
000066 001096 INCX=78
000067 001097 IF (NYN,GT.85) INCX=14
000068 001098 IF (NYN.NF.0) CALL DICH (INCX,MIN0(IYB+52,IYB-(IYB-IYT-12*NYN)/4)
000069 001099 1,NYN,1,ARELY,2)
000070 001100 IX=MAX0(44,(IXR-IXL-12*MZL)/2)
000071 001101 LZL=MAX0(MZL-(1024-IX)/12,0)+1
000072 001102 IF (LZL,GT.1) LZL=(LZL+3)/4+1

```

000073	001103	IF (N7L.NF.0) CALL DLCH (IX.MAX0(10.IYT-24-16*LZL),MZL,LABELZ,2)
000074	001104	C
000075	001105	C INCREASE EXPOSURE
000076	001106	CALL FXH
000077	001107	IF (N7L.LT.0) GO TO 220
000078	001108	C
000079	001109	C PLOT POINTS AND/OR LINE
000080	001110	130 MPTS=MPTS+KINC
000081	001111	DO 210 NXP=1,MPTS,KINC
000082	001112	XTWO=X(NXP)
000083	001113	YTWO=Y(NXP)
000084	001114	IF (MPTS.IT.0) XTWO=ALOG10(XTWO)
000085	001115	IF (INP.LT.0) YTWO=ALOG10(YTWO)
000086	001116	CALL CONVRT (XTWO,NXTWO,XMN,XMX,XL,XR)
000087	001117	CALL CONVRT (YTWO,NYTWO,YMN,YMX,YB,YT)
000088	001118	IF (NXP.EQ.1) GO TO 100
000089	001119	IF (LIN.GF.0) GO TO 100
000090	001120	100 IF (MOD((NXP-1)/KINC),IARS(LIN),NE.0) GO TO 150
000091	001121	C
000092	001122	C LIGHTEN EXPOSURE
000093	001123	CALL FXL
000094	001124	C
000095	001125	C REMOVES CONTOUR LABELS
000096	001126	C CALL DLCH (NXTWO,NYTWO,0,KSYM,1)
000097	001127	CALL FXH
000098	001128	GO TO 200
000099	001129	150 IF (R.EQ.0) GO TO 200
000100	001130	160 DO 170 IB=1,4
000101	001131	170 CALL PLT (NXTWO,NYTWO,42)
000102	001132	GO TO 200
000103	001133	180 IF (R.EQ.0) CALL DRV (NXONE,NYONE,NXTWO,NYTWO)
000104	001134	190 IF (LIN.NF.0) GO TO 140
000105	001135	IF (R.NE.0) GO TO 160
000106	001136	200 NYONE=NYTWO
000107	001137	NXONE=NXTWO
000108	001138	210 CONTINUE
000109	001139	220 RETURN
000110	001140	END

CMAI 1670 07107144 004/02/77

```

000001 001141 C
000002 001142 C
000003 001143 SUBROUTINE CONTRJB(X,NNX,Y,NNY,Z,NZX,NZY,NC,7MN,ZMX,DLZ,ZPLAN,DMPX
000004 001144 1,DMPY,IGRN,ITITLE,NTITLE,XLABEL,NXLBL,YLABEL,NYLHL)
000005 001145 C
000006 001146 C
000007 001147 C ESTABLISH THE CONTOUR SCHEME TO BE USED BY IRICJR
000008 001148 COMMON /CNTRCOM/ ISYM(50)
000009 001149 DIMENSION XSCALE(2), YSCALE(2)
000010 001150 EQUIVALENCE (XMIN,XSCALE(1)), (XMAX,XSCALE(2))
000011 001151 EQUIVALENCE (YMIN,YSCALE(1)), (YMAX,YSCALE(2))
000012 001152 DIMENSION X(1), Y(1), Z(NZX,1), ZPLAN(1)
000013 001153 DIMENSION ZT(4)
000014 001154 DIMENSION COM(6)
000015 001155 NOC=MINTO(7ABS(NC)+50)
000016 001156 ZMIN=7MN
000017 001157 ZMAX=7MX
000018 001158 DELZ=DLZ
000019 001159 DMAPX=DMPX
000020 001160 DMAPY=DMPY
000021 001161 NOX=7ABS(NNX)
000022 001162 NOY=7ABS(NNY)
000023 001163 DO 10 I=1,50
000024 001164 10 ISYM(I)=0
000025 001165 C
000026 001166 C ESTABLISH SCALES
000027 001167 CALL MINV (X=1,NOX,I,XMIN)
000028 001168 CALL MAXV (Y=1,NOY,I,YMAX)
000029 001169 CALL MAXV (X=1,NOX,I,XMAX)
000030 001170 CALL MINV (Y=1,NOY,I,YMIN)
000031 001171 FGRD=0
000032 001172 IF (IGRD.NE.0) FGRD=-IGRD
000033 001173 CALL PLOJR (XSCALE,YSCALE,2,1,1,1,FGRD,DMAPX,DMAPY,ITITLE,-NTITLE
000034 001174 1,XLABEL,NXLBL,YLABEL,NYLBL)
000035 001175 IF (NC.LT.0) GO TO 50
000036 001176 C
000037 001177 C DETERMINE MAX-MIN CONTOURS AND SPACING
000038 001178 C STORE IN ARRAY ZPLAN
000039 001179 IF (NNX.LF.0) CALL MINM (Z,NZX,NOX,NOY,I,J,ZMIN)
000040 001180 IF (NNY.LF.0) CALL MAXM (Z,NZX,NOX,NOY,I,J,ZMAX)
000041 001181 IF (DELZ.NE.0) GO TO 40
000042 001182 DELZ=(ZMAX-ZMIN)/(NOC-1.)
000043 001183 20 IF (NZY.GT.0) GO TO 30
000044 001184 ZMAX=7MAX-AMOD(ZMAX,DELZ)
000045 001185 ZMIN=7MIN-AMOD(ZMIN,DELZ)
000046 001186 NOC=MINTO(NOC+IFIX((ZMAX-ZMIN)/DELZ+1,01))
000047 001187 30 ZPLAN(I)=7MIN
000048 001188 DO 40 I=2,NOC
000049 001189 40 ZPLAN(I)=7PLAN(I-1)+DELZ

```

000050	001190	C
000051	001191	C STORE THE ARRAY BEING CONTOURED IN ZT AND CONTOUR ONE CELL AT A TIME
000052	001192	50 DO 90 NY=2,N0Y
000053	001193	IX=MOD(NY,2)
000054	001194	DY=Y(NY)-Y(NY-1)
000055	001195	DO 80 INX=2,N0X
000056	001196	NX=INX
000057	001197	IF (IX.NE.0) NX=N0X-INX+2
000058	001198	ZT(1)=Z(NY-1,NY-1)
000059	001199	ZT(2)=Z(NX,NY-1)
000060	001200	ZT(3)=Z(NX,NY)
000061	001201	ZT(4)=Z(NX-1,NY)
000062	001202	DX=X(NX)-X(NX-1)
000063	001203	IF (ABS(ZT(3)-ZT(1))-ABS(ZT(4)-ZT(2))) 70,60,60
000064	001204	60 CALL TRIC,B (X(NX),Y(NY),-DX,-DY,NOC,ZPLAN,ZT(4),ZT(3),ZT(2))
000065	001205	CALL TRIC,B (X(NX-1),Y(NY-1),DX,DY,NOC,ZPLAN,ZT(2),ZT(1),ZT(4))
000066	001206	GO TO 80
000067	001207	70 CALL TRIC,B (X(NX-1),Y(NY),DX,-DY,NOC,ZPLAN,ZT(3),ZT(4),ZT(1))
000068	001208	CALL TRIC,B (X(NX),Y(NY-1),-DX,DY,NOC,ZPLAN,ZT(1),ZT(2),ZT(3))
000069	001209	80 CONTINUE
000070	001210	90 CONTINUE
000071	001211	C
000072	001212	C IDENTIFY THE MAX, MIN, AND SPACING FOR CONTOURS
000073	001213	REWIND 63
000074	001214	WRITE(63,900) ZPLAN(1),ZPLAN(NOC),DELTA
000075	001215	REWIND 63
000076	001216	READ(63,901) (CON(I),I=1,6)
000077	001217	CALL NLCH(48,1000,60,CON(1),2)
000078	001218	900 FORMAT(1X,15HMIN,MAX,DELTA,1P3E12.4)
000079	001219	901 FORMAT(1X,4A10)
000080	001220	RETURN
000081	001221	C
000082	001222	END

```

000001 001223
000002 001224
000003 001225
000004 001226
000005 001227
000006 001228
000007 001229
000008 001230
000009 001231
000010 001232
000011 001233
000012 001234
000013 001235
000014 001236
000015 001237
000016 001238
000017 001239
000018 001240
000019 001241
000020 001242
000021 001243
000022 001244
000023 001245
000024 001246
000025 001247
000026 001248
000027 001249
000028 001250
000029 001251
000030 001252
000031 001253
000032 001254
000033 001255
000034 001256
000035 001257
000036 001258
000037 001259
000038 001260
000039 001261
000040 001262
000041 001263
000042 001264
000043 001265
000044 001266
000045 001267
000046 001268
000047 001269
000048 001270
000049 001271
000050 001272
000051 001273
000052 001274

```

CHAI 1070 0915V124 504/02/74

```

C
C
C SUBROUTINE TRICJB(X,Y,DX,DY,NOC,ZPLAN,ZX,ZV,ZY)
C
C PERFORM ACTUAL INTERPOLATIONS NEEDED TO PRODUCE CONTOURS
C DEFINED IN CONTRJB. PLOJR USED TO CONNECT POINTS.
C
COMMON /CNTRCOM/ ISYM(50)
DIMENSION XP(2,50), YP(2,50), ZT(4), ZPLAN(1)
ZT(1)=ZX
ZT(2)=ZV
ZT(3)=ZY
ZT(4)=ZX
ZTMIN=AMIN1(ZT(1),ZT(2),ZT(3))
ZTMAX=AMAX1(ZT(1),ZT(2),ZT(3))
IMIN=NOC+1
IMAX=0
DO 10 K=1,NOC
J=NOC-K+1
IF (ZPLAN(J).GE.ZTMIN) IMIN=J
IF (ZPLAN(K).LE.ZTMAX) IMAX=K
10 CONTINUE
INT=IMAX-IMIN
IF (INT.LE.0.OR.ZTMIN.EQ.ZTMAX) GO TO 130
I2=1
DO 110 K=1,3
ZTMAX=AMAX1(ZT(K),ZT(K+1))
ZPMIN=AMIN1(ZT(K),ZT(K+1))
MIN=NOC+1
MAX=0
DO 20 J=1,NOC
INZ=NOC-J+1
IF (ZPLAN(INZ).GT.ZPMIN.OR.(ZPLAN(INZ).EQ.ZPMIN.AND.ZTMIN.EQ.ZPMIN
1) MIN=INZ
IF (ZPLAN(J).LE.ZTMAX) MAX=J
20 CONTINUE
INZ=MAX-MIN
IF (INZ.LE.0.OR.ZTMAX.EQ.ZPMIN) GO TO 110
IF (INZ=INT) 40,30,40
30 GO TO (50,40), I2
40 I2=1
GO TO 60
50 I2=2
60 DO 100 J=MIN,MAX
GO TO (70,80,90), K
70 XP(I2,J)=X+DX*(ZPLAN(J)-ZT(2))/(ZT(1)-ZT(2))
YP(I2,J)=Y
GO TO 100
80 XP(I2,J)=X
YP(I2,J)=Y+DY*(ZPLAN(J)-ZT(2))/(ZT(3)-ZT(2))
GO TO 100

```


000053 001275
 000054 001276
 000055 001277
 000056 001278

 000057 001279
 000058 001280
 000059 001281
 000060 001282
 000061 001283
 000062 001284
 000063 001285
 000064 001286
 000065 001287

```

90 XP(12,J)=X*NX*(ZPLAN(J)-ZT(3))/(ZT(1)-ZT(3))
   YP(12,J)=Y*DY*(ZPLAN(J)-ZT(1))/(ZT(3)-ZT(1))
100 CONTINUE
110 CONTINUE
  

   DO 128 J=1,N,1MAX
     ISYM(J)=1+SYM(J)+1
     L=3
     IF (MOD(ISYM(J)+10),NF,1) .NE. 0
       CALL PLOJR (XP(1,J),YP(1,J),2,1,L=J,0,0,0,0,0,0,0,0,0,0)
120 CONTINUE
C
130 RETURN
   END

```

APPENDIX B
CODE LISTING FOR GEØ-II

Example Numbers 4-5

Figures 7-8

LASL Identification Number LP-1093

```
*****
000001 000001 C
000002 000002 C VARIABLE DEFINITIONS
000003 000003 C
000004 000004 C
000005 000005 C AKAPR - KAPPA ROCK THERMOELECTRIC CONDUCTIVITY OF ROCK
000006 000006 C AKAPD - RATIO OF THERMAL CONDUCTIVITY IN ROCK TO THE
000007 000007 C PRODUCT OF THE DENSITY OF WATER AND VOLUMETRIC
000008 000008 C EXPANSION COEFFICIENT FOR WATER
000009 000009 C BETA - BETA WATER- VOLUMETRIC EXPANSION COEFFICIENT FOR WATER
000010 000010 C BETAR - BETA ROCK- LINEAR EXPANSION COEFFICIENT FOR ROCK
000011 000011 C DC - DISTANCE BETWEEN SECONDARY CRACKS
000012 000012 C DT - DELTA T, CHANGE IN THE ELAPSED TIME
000013 000013 C DX - CHANGE IN THE DISTANCE IN X DIRECTION
000014 000014 C DY - CHANGE IN THE DISTANCE IN Y DIRECTION
000015 000015 C DZ - CHANGE IN THE DISTANCE IN Z DIRECTION
000016 000016 C RAM - NEWTON RAPHSOON RELAXATION COEFFICIENT
000017 000017 C GRAV - ACCELERATION OF GRAVITY
000018 000018 C WC - HEAT CONVECTION TERMS
000019 000019 C TRAR - RIGHT BOUNDARY OF TOTAL MESH IN Y DIRECTION
000020 000020 C LRAR - TOP BOUNDARY OF MESH IN Y DIRECTION
000021 000021 C KRAR - TOP BOUNDARY OF MESH IN Z DIRECTION
000022 000022 C PHI - PRESSURE/ DENSITY
000023 000023 C BNU - COEFFICIENT OF KINEMATIC VISCOSITY
000024 000024 C BNUA - COEFFICIENT OF ARTIFICIAL KINEMATIC VISCOSITY
000025 000025 C T - WATER TEMPERATURE
000026 000026 C TF - TOTAL CALCULATION TIME
000027 000027 C THICK - THICKNESS OF INITIAL COOLING ZONE IN THE ROCK
000028 000028 C TJ - RUNNING PROBLEM TIME
000029 000029 C TJN - WATER TEMPERATURE AT INLET
000030 000030 C TP - TIME TO PRINT
000031 000031 C TPC - TIME PRINT CONSTANT
000032 000032 C TPI - TIME PRINT INTERVAL
000033 000033 C U - VELOCITY IN X DIRECTION
000034 000034 C V - VELOCITY IN Y DIRECTION
```

000035	000035	C	VIN	=	VELOCITY AT INLET
000036	000036	C	VOUT	=	VELOCITY AT OUTLET
000037	000037	C	W	=	CRACK WIDTH
000038	000038	C	WO	=	YIELD WIDTH FOR CRACK PRODUCTION
000039	000039	C	VI	=	ROCK TEMPERATURE
000040	000040	C	VII	=	ROCK TEMPERATURE AT INFINITY
000041	000041	C			
000042	000042	C			
000043	000043	C			
000044	000044				PROGRAM 3=0 (OUTPUT, 6F0001, TAPE4=RE0001, TAPE63=1000)
000045	000045	C			
000046	000046	C			
000047	000047		PARAMETER	(NT=23)	
000048	000048		PARAMETER	(NJ=23)	
000049	000049		PARAMETER	(NK=10)	
000050	000050		COMMON/ARAYS/GAM	(NI,NJ,WC(NI,NJ),PHI(NI,NJ),T(NI,NJ),	
000051	000051	1		UWBAR(NI,NJ),VWBAR(NI,NJ),U(NI,NJ),	
000052	000052	2		V(NI,NJ),W(NI,NJ),WNP1(NI,NJ),XT(NI,NJ,NK),	
000053	000053	3		XINP1(NI,NJ,NK),J(NI,NJ),TNP1(NI,NJ),TBR(NI,NJ)	
000054	000054	C			
000055	000055	C			
000056	000056		COMMON/VAR/AKAP0,AKAP0,BETA,BETAR,DT,DTR,DY,DY,D7,DYR,DYR,D7R,		
000057	000057	1		DYS0,DYS0,D7C4,DYS00,DYS0R,D7S00,ET0FF,FPSTLON,GRAV,	
000058	000058	2		IRAR,IRP1,TR0C,JHAR,JHP1,JRP2,	
000059	000059	3		KAR,KRP1,KR01,NCYCLF,NIT,RH0A,TI,TF,TD,	
000060	000060	4		TRC,TPT,THICK,TIN,TJNTHY,VIN,VOUT,W0,DC,XT,	
000061	000061	5		BR,RW,RW0P,RW0,AKR,DSIM,WMAX	
000062	000062	C			
000063	000063	C			
000064	000064		CALL	UNQCHG	
000065	000065		CALL	UNDR0P	
000066	000066	C			
000067	000067	C			
000068	000068	C	TIMING	STUDIES	
000069	000069	C			
000070	000070		CALL	00TIM	
000071	000071		ISEC	=	0
000072	000072	C			
000073	000073	C			
000074	000074	C			
000075	000075	C	VARIABLES		
000076	000076	C			
000077	000077		BW	=	1.0
000078	000078		BR	=	0.25
000079	000079		RH0W	=	1.0
000080	000080		RH0R	=	2.65
000081	000081		AKR	=	6.2E-3
000082	000082		AKAP0	=	AKR / (RH0W*RW)
000083	000083		AKAPR	=	AKR / (RH0R*RW)
000084	000084		BETA	=	2.07E-3
000085	000085		BETAR	=	2.5E-3
000086	000086		DC	=	3A.0

00087	000087	DT	= 0.0
00088	000088	KBAR	= 4
00089	000089	KBP1	= KBAR*.1
00090	000090	KBM1	= KBAR-.1
00091	000091	DX	= 10.0
00092	000092	DY	= 10.0
00093	000093	DZ	= 0.5*DC/LOAT(KBM1)
00094	000094	DXR	= 1./DX
00095	000095	DYR	= 1./DY
00096	000096	DZR	= 1./DZ
00097	000097	DXSQ	= DY**2
00098	000098	DYSQ	= DX**2
00099	000099	DZSQ	= DZ**2
00100	000100	DXSQM	= 1./DXSQ
00101	000101	DYSQM	= 1./DYSQ
00102	000102	DZSQM	= 1./DZSQ
00103	000103	EIGEIF	= .0/15.
00104	000104	EPSILON	= 2.E-3
00105	000105	GRAV	= 900.0
00106	000106	IqAV	= 20
00107	000107	IqP1	= IqAR*.1
00108	000108	IqP2	= IqAR*.2
00109	000109	JqAR	= 20
00110	000110	JqP1	= JqAR*.1
00111	000111	JqP2	= JqAR*.2
00112	000112	RvHA	= 0.01
00113	000113	THICK	= 50.0
00114	000114	TIN	= 55.0
00115	000115	TWOTHI	= 2./1.
00116	000116	VIN	= 10.0
00117	000117	W0	= 0.0
00118	000118	XII	= 300.0
00119	000119		
00120	000120		
00121	000121		
00122	000122		
00123	000123		
00124	000124		
00125	000125		
00126	000126		
00127	000127		
00128	000128		
00129	000129		
00130	000130		
00131	000131		
00132	000132		
00133	000133		
00134	000134		

C
 C
 C INITIAL CONDITIONS
 C
 DO 199 I = 2,IBAR
 DU 198 J = 2,JBP1
 U(I,J) = 0.0
 198 CONTINUE
 199 CONTINUE
 C
 DO 299 I = 3,IBP1
 DU 298 J = 2,JBAR
 V(I,J) = 0.0
 V(2,J) = VIN
 298 CONTINUE
 299 CONTINUE

00135 000135
 00136 000136
 00137 000137
 00138 000138
 00139 000139
 00140 000140
 00141 000141
 00142 000142
 00143 000143
 00144 000144
 00145 000145
 00146 000146
 00147 000147
 00148 000148
 00149 000149
 00150 000150
 00151 000151
 00152 000152
 00153 000153
 00154 000154
 00155 000155
 00156 000156
 00157 000157
 00158 000158
 00159 000159
 00160 000160
 00161 000161
 00162 000162
 00163 000163
 00164 000164
 00165 000165
 00166 000166
 00167 000167
 00168 000168
 00169 000169
 00170 000170

000171 000171
 000172 000172
 000173 000173
 000174 000174
 000175 000175
 000176 000176
 000177 000177
 000178 000178
 000179 000179
 000180 000180
 000181 000181
 000182 000182
 000183 000183
 000184 000184
 000185 000185
 000186 000186

```

C
C      V(2,1) = VIN
      V(2,J901) = VIN
C
      DO 379 I = 3,IBP1
        DU 378 J = 2,JBP1
          DO 377 K = 1,KBP1
            VBAR = (FLOAT(I-2) - 0.5)*DX/THICK
            VI(I,J,K) = TIN + (XI(I,J,K)-TIN)*ERF(VBAR)
          377 CONTINUE
        378 CONTINUE
      379 CONTINUE
C
C      DO 439 I = 3,IBP1
        DU 438 J = 2,JBP1
          T(I,J) = XI(I,J,1)
          PHI(I,J) = 0.0
          T(2,J) = TIN
          PHI(2,J) = 0.0
        438 CONTINUE
      439 CONTINUE
C
      DO 599 I = 3,IBP1
        DU 598 J = 2,JBP1
          SUM1 = 0.0
          DO 597 K = 2,KBP1
            SUM1 = SUM1 + (XI(I,J,K) - XI(I))
          597 CONTINUE
          WNP1(I,J) = -2.0*DETAR*DT*(XI(I,J,1)-XI(I)*SUM1)
          IF(WNP1(I,J).LE.WN) WNP1(I,J) = 0.0
          W(I,J) = WNP1(I,J)
        598 CONTINUE
      599 CONTINUE
C
      WMAX = 0.0
      DO 658 J = 2,JBP1
        W(2,J) = W(3,J)
        WNP1(2,J) = WNP1(3,J)
        WTEST = WNP1(3,1)
        WMAX = AMAX1(WTEST,WMAX)
      658 CONTINUE
C
C      BOUNDARY CONDITIONS ON XI
C
      DO 698 J = 2,IBP1
        DU 697 K = 1,KBP1
          XI(IHP2,J,K) = XI(I)
        697 CONTINUE
      698 CONTINUE

```

08

```

000187 000187 C
000188 000188 QUEST = 3*AV*RETA*(TIN-7.9)*(XTI-TIN)/17.856
000189 000189 DEPSO = EPSILON*QUFST*DYH
000190 000190 VTESI = (WMAX**2)*QUFST
000191 000191 IF(DI.NE.4.0) GO TO 700
000192 000192 DT = (.2**SQRT(DX**2+DY**2))/VTESI
000193 000193 700 CONTINUE
000194 000194 DTR = 1./DT
000195 000195 TI = 0.0
000196 000196 TP = 0.0
000197 000197 TPC = (1.F-3)*DT
000198 000198 TF = 500.*DT
000199 000199 TPI = 50.*DT
000200 000200 C
000201 000201 C
000202 000202 C SET CYCLE COUNTERS
000203 000203 C
000204 000204 NCYCLE = 0
000205 000205 900 CONTINUE
000206 000206 C
000207 000207 C
000208 000208 C CHECK FOR PRINTING INTERVAL
000209 000209 C
000210 000210 IF(TI + TPC.GE.TP) GO TO 4100
000211 000211 C
000212 000212 C*****
000213 000213 C***** HE3IN CYCLE *****
000214 000214 C*****
000215 000215 C
000216 000216 C
000217 000217 C
000218 000218 1100 CONTINUE
000219 000219 C
000220 000220 C BOUNDARY CONDITIONS ON XT
000221 000221 C
000222 000222 DO 1150 I = 1,IHP1
000223 000223 DU 1157 K = 1,KRP1
000224 000224 XI(I,JRP2,K) = XI(1,JRP1,K)
000225 000225 XI(I+1,K) = XI(I,K)
000226 000226 1157 CONTINUE
000227 000227 1150 CONTINUE
000228 000228 C
000229 000229 DO 1190 I = 1,IHP1
000230 000230 DU 1190 J = 2,JRP1
000231 000231 XI(I,J,1) = T(I,J)
000232 000232 XI(I,J,KRP1) = XI(1,J,KBAR)
000233 000233 1190 CONTINUE
000234 000234 1190 CONTINUE
000235 000235 C
000236 000236 DO 1290 J = 2,JRP1
000237 000237 DU 1297 K = 1,KRP1
000238 000238 XI(2,J,K) = 2.0*TIN - XI(3,J,K)

```

000239	000239	1297 CONTINUE
000240	000240	129A CONTINUE
000241	000241	C
000242	000242	C CALCULATE NEW XI
000243	000243	C
000244	000244	DO 1399 I = 3,IBP1
000245	000245	DU 139A J = 2,JRP1
000246	000246	DO 1397 K = 2,KRAP
000247	000247	YINP1(I,J,K) = XI(I,J,K) + AKAPRINT*
000248	000248	(DXSQR*(XI(I+1,J,K)+XI(I-1,J,K)-2.*XI(I,J,K))
000249	000249	+ DYSQR*(XI(I,J+1,K)+XI(I,J-1,K)-2.*XI(I,J,K))
000250	000250	+ DZSQR*(XI(I,J,K+1)+XI(I,J,K-1)-2.*XI(I,J,K))
000251	000251	1397 CONTINUE
000252	000252	139A CONTINUE
000253	000253	1399 CONTINUE
000254	000254	C
000255	000255	C STORE NEW XI INTO OLD
000256	000256	C
000257	000257	DO 1499 I = 3,IBP1
000258	000258	DU 149A J = 2,JRP1
000259	000259	DO 1497 K = 2,KRAP
000260	000260	XI(I,J,K) = XINP1(I,J,K)
000261	000261	1497 CONTINUE
000262	000262	149A CONTINUE
000263	000263	1499 CONTINUE
000264	000264	C
000265	000265	C STORE NEW W INTO OLD W AND CALCULATE NEW W
000266	000266	C
000267	000267	DO 1599 I = 3,IBP1
000268	000268	DU 159A J = 2,JRP1
000269	000269	W(I,J) = WNP1(I,J)
000270	000270	SUM1 = 0.0
000271	000271	DO 1597 K = 2,KRAP
000272	000272	SUM1 = SUM1 + (XI(I,J,K) - XI(I,J,K-1))
000273	000273	1597 CONTINUE
000274	000274	WNP1(I,J) = -2.0*ACTAR*D7*(.5*(XI(I,I+1)-XI(I,I) + SUM1)
000275	000275	IF(WNP1(I,J).LE.WN) WNP1(I,J) = 0.0
000276	000276	IF(WNP1(I,J).GE.WN) WNP1(I,J) = WNP1(I,J)
000277	000277	159A CONTINUE
000278	000278	1599 CONTINUE
000279	000279	C
000280	000280	WMAX = 0.0
000281	000281	DO 165A J = 2,JRP1
000282	000282	WTEST = WNP1(3,J)
000283	000283	WMAX = AMAX1(WTEST,WMAX)
000284	000284	165A CONTINUE
000285	000285	C
000286	000286	C
000287	000287	DO 169A J = 2,JRP1
000288	000288	W(2,J) = WNP1(2,J)
000289	000289	WNP1(2,J) = WNP1(2,J)
000290	000290	169A CONTINUE

```

10291 000291 C
10292 000292 C INITIALIZE FDO ITERATION
10293 000293 C
10294 000294 C
10295 000295 DO 2199 I = 2,IBAP
10296 000296 DU 2199 J = 2,JBAP
10297 000297 UC = .5*(U(I-1,J) + U(I,J))
10298 000298 IF (UC.RE.0.) A = W(I,J)*UC*(I-1,J)
10299 000299 IF (UC.LT.0.) A = W(I,J)*UC*(I,J)
10300 000300 UC = .5*(U(I,J) + U(I+1,J))
10301 000301 IF (UC.RE.0.) B = W(I+1,J)*UC*(I,J)
10302 000302 IF (UC.LT.0.) B = W(I+1,J)*UC*(I+1,J)
10303 000303 WFAC = .5*AMIN1(W(I,J-1),W(I+1,J-1),W(I,J)+W(I+1,J))
10304 000304 VC = .5*(V(I,J-1) + V(I+1,J-1))
10305 000305 IF (VC.RE.0.) C = WFAC*VC*(I,J-1)
10306 000306 IF (VC.LT.0.) C = WFAC*VC*(I,J)
10307 000307 WFAC = .5*AMIN1(W(I,J),W(I+1,J),W(I,J+1)+W(I+1,J+1))
10308 000308 VC = .5*(V(I,J) + V(I+1,J))
10309 000309 IF (VC.RE.0.) D1 = WFAC*VC*(I,J)
10310 000310 IF (VC.LT.0.) D1 = WFAC*VC*(I,J+1)
10311 000311 WFACA = .5*AMIN1(W(I,J+1)+W(I+1,J+1),W(I,J)+W(I+1,J))
10312 000312 WFACB = .5*AMIN1(W(I,J)+W(I+1,J),W(I,J-1)+W(I+1,J-1))
10313 000313 C
10314 000314 UNBAR(I,J) = (U(I,J)+.5*(W(I,J)+W(I+1,J)))
10315 000315 1 + DT*(F1GIF*(NYR*(A-B) + NYR*(C-D))
10316 000316 2 + RN1A*DXSQ*(W(I+1,J)*(U(I+1,J)-U(I,J))
10317 000317 3 - W(I,J)*(U(I,J)-W(I-1,J))))
10318 000318 4 + RN1A*DYSQ*(WFACA*(U(I,J+1)-U(I,J))
10319 000319 5 - WFACB*(U(I,J)-U(I,J-1))))
10320 000320 2199 CONTINUE
10321 000321 2199 CONTINUE
10322 000322 C
10323 000323 C
10324 000324 C
10325 000325 DO 2299 I = 1,IBP1
10326 000326 DU 2299 J = 2,JBAP
10327 000327 VC = .5*(V(I,J-1) + V(I,J))
10328 000328 IF (VC.RE.0.) E = W(I,J)*VC*(I,J-1)
10329 000329 IF (VC.LT.0.) E = W(I,J)*VC*(I,J)
10330 000330 VC = .5*(V(I,J) + V(I+1,J))
10331 000331 IF (VC.RE.0.) F = W(I,J+1)*VC*(I,J)
10332 000332 IF (VC.LT.0.) F = W(I,J+1)*VC*(I,J+1)
10333 000333 WFAC = .5*AMIN1(W(I-1,J),W(I+1,J),W(I,J)+W(I+1,J))
10334 000334 UC = .5*(U(I-1,J) + U(I+1,J))
10335 000335 IF (UC.RE.0.) G = WFAC*UC*(I-1,J)
10336 000336 IF (UC.LT.0.) G = WFAC*UC*(I,J)
10337 000337 WFAC = .5*AMIN1(W(I,J)+W(I+1,J),W(I,J-1)+W(I+1,J-1))
10338 000338 UC = .5*(U(I,J) + U(I+1,J))
10339 000339 IF (UC.RE.0.) H = WFAC*UC*(I,J)
10340 000340 IF (UC.LT.0.) H = WFAC*UC*(I+1,J)
10341 000341 WFACC = .5*AMIN1(W(I,J)+W(I+1,J),W(I,J-1)+W(I+1,J-1))

```



```

000343 000343      WFAPI = .5*AMIN1(W(I,J)+W(I,J+1),W(I-1,J)+W(I-1,J+1))
000344 000344      C
000345 000345      WBAR(I,J) = (W(I,J)+.5*(W(I,J)+W(I,J+1)))
000346 000346      1      + NT*(F10F1F*(NYR*(F-F) + DXR*(Q-H))
000347 000347      2      + .5*(W(I,J)+W(I,J+1))*RET*GRAV
000348 000348      3      + (.5*(W(I,J) + W(I,J+1)) - TIN)
000349 000349      4      + AM1A*DXSQ*(WFACC*(V(I+1,J)-V(I,J))
000350 000350      5      - WFACT*(V(I,J) - V(I-1,J)))
000351 000351      6      + AM1A*DYSD*(W(I,J+1)*V(I,J+1) - V(I,J))
000352 000352      7      - W(I,J)*(V(I,J) - V(I,J-1)))
000353 000353      229A CONTINUE
000354 000354      229B CONTINUE
000355 000355      C
000356 000356      C ***** REIN ITERATION *****
000357 000357      C
000358 000358      C
000359 000359      C CALCULATE D (MASS EQUATION)
000360 000360      C
000361 000361      NIT = 0
000362 000362      250A CONTINUE
000363 000363      DTESI = 0.0
000364 000364      DSUM = 0.0
000365 000365      DO 2599 I = 3,IRP1
000366 000366      DU 259A J = 2,IRP1
000367 000367      WFAA = AMIN1(WNP1(I,J)+WNP1(I+1,J))
000368 000368      WFAB = AMIN1(WNP1(I-1,J)+WNP1(I,J))
000369 000369      WFAC = AMIN1(WNP1(I,J)+WNP1(I,J+1))
000370 000370      WFAD = AMIN1(WNP1(I,J)+WNP1(I,J-1))
000371 000371      C
000372 000372      D(I,J) = TWOTHY*(WFACA*(I,J)-WFACB*U(I-1,J))
000373 000373      1      + DXR*(WFACC*V(I,J) - WFACD*V(I,J-1)*NYR)
000374 000374      2      + (WNP1(I,J) - W(I,J))*DTR
000375 000375      DAGE = ABS(D(I,J))
000376 000376      DTEST = AMAX1(DAGE,DTEST)
000377 000377      DSUM = DSUM + D(I,J)
000378 000378      C
000379 000379      C CALCULATE GAMMA
000380 000380      C
000381 000381      GTEOM = (2.72*NT)/(T(I,J)-T,A)
000382 000382      IF(WNP1(I,J).EQ.0.0) GO TO 2550
000383 000383      GAM(I,J) = 0.75*(WNP1(I,J)*G) /
000384 000384      1      (WNP1(I,J)*3*NT*(DXSQ+DYSD))
000385 000385      GO TO 2570
000386 000386      255A CONTINUE
000387 000387      GAM(I,J) = 0.0
000388 000388      257A CONTINUE
000389 000389      C
000390 000390      C CALCULATE PHI
000391 000391      C
000392 000392      PHI(I,J) = PHI(I,J) - (GAM(I,J)*D(I,J))
000393 000393      259A CONTINUE
000394 000394      259B CONTINUE

```

```

000395 000395 C
000396 000396 C BOUNDARY CONDITIONS ON PHI
000397 000397 C
000398 000398 DO 265A J = 2,JRP1
000399 000399 PHI(2,1) = -PHI(3,1)
00400 000400 265A CONTINUE
00401 000401 C
00402 000402 C TEST FOR CONVERGENCE
00403 000403 C
00404 000404 IF(NIT,E2,0) GO TO 2450
00405 000405 DEPS = DEPS0*WMAX**3
00406 000406 IF(01EST,1,T,DEPS) GO TO 3000
00407 000407 IF(NIT,0E,1000) GO TO 3000
00408 000408 265A CONTINUE
00409 000409 C
00410 000410 C NO CONVERGENCE
00411 000411 C
00412 000412 C
00413 000413 C CALCULATE II AND V
00414 000414 C
00415 000415 DO 2699 I = 2,IBAP
00416 000416 DU 2699 J = 2,JRP1
00417 000417 GTERMU = 2.232*DT / (.5*(T(I,J)+T(I+1,J))-3.0)
00418 000418 WFACU = .5*(WNP1(I,J)+WNP1(I+1,J))
00419 000419 WMINU = AMIN1(WNP1(I,J),WNP1(I+1,J))
00420 000420 U(I,J) = WMINU*(UWBAR(I,J)+WMINU*NT*TXR*(PHI(I,J)
00421 000421 -PHI(I+1,J))) / (WMINU*WFACU+GTERMU)
00422 000422 1 CONTINUE
00423 000423 2699 CONTINUE
00424 000424 C
00425 000425 DO 2799 I = 3,IBP1
00426 000426 DU 2799 J = 2,JRAP
00427 000427 GTERMV = 2.232*DT / (.5*(T(I,J)+T(I,J+1))-3.0)
00428 000428 WFACV = .5*(WNP1(I,J)+WNP1(I,J+1))
00429 000429 WMINV = AMIN1(WNP1(I,J),WNP1(I,J+1))
00430 000430 V(I,J) = WMINV*(VWBAR(I,J)+WMINV*NT*TYR*(PHI(I,J)
00431 000431 -PHI(I,J+1))) / (WMINV*WFACV+GTERMV)
00432 000432 1 CONTINUE
00433 000433 2799 CONTINUE
00434 000434 NIT = NIT + 1
00435 000435 GO TO 250A
00436 000436 C
00437 000437 C ***** END ITERATION *****
00438 000438 C
00439 000439 C
00440 000440 C CONVERGENCE ATTAINED, CONTINUE CYCLE
00441 000441 C
00442 000442 300A CONTINUE

```

```

000443 000443 C CALCULATE NEW TEMPERATURES (WATER TEMPERATURE CONVECTION)
000444 000444 C
000445 000445 C
000446 000446 DO 4199 I = 2,IRP1
000447 000447 DU 4199 J = 2,IRP1
000448 000448 WFAC = AMIN1(WNP1(I,J),WNP1(I-1,J))
000449 000449 HC = U(I-1,J)
000450 000450 IF (HC.GE.0.0) AA = WFAC*HC*(I-1,J)
000451 000451 IF (HC.LT.0.0) AA = WFAC*HC*(I,J)
000452 000452 WFAC = AMIN1(WNP1(I,J),WNP1(I+1,J))
000453 000453 HC = U(I,J)
000454 000454 IF (HC.GE.0.0) RR = WFAC*HC*(I,J)
000455 000455 IF (HC.LT.0.0) RR = WFAC*HC*(I+1,J)
000456 000456 WFAC = AMIN1(WNP1(I,J),WNP1(I,J+1))
000457 000457 VC = V(I,J-1)
000458 000458 IF (VC.GE.0.0) CC = WFAC*VC*(I,J-1)
000459 000459 IF (VC.LT.0.0) CC = WFAC*VC*(I,J)
000460 000460 WFAC = AMIN1(WNP1(I,J),WNP1(I,J+1))
000461 000461 VC = V(I,J)
000462 000462 IF (VC.GE.0.0) DD = WFAC*VC*(I,J)
000463 000463 IF (VC.LT.0.0) DD = WFAC*VC*(I,J+1)
000464 000464 C
000465 000465 HC(I,J) = DT*(TWOTHT*DXR*(AA-RR) + TWOTHT*DYB*(CC-DD)
000466 000466 + W(I,J)*T(I,J) + T(I,J)*D(I,J)*DT
000467 000467 1
000468 000468 4199 CONTINUE
000469 000469 4199 CONTINUE
000470 000470 C
000471 000471 C
000472 000472 AKAPTERM = 2.0*AKAPD*NT*U7R
000473 000473 DO 4299 I = 1,IRP1
000474 000474 DU 4299 J = 2,IRP1
000475 000475 T(I,J) = 1.0 / (WNP1(I,J) + AKAPTERM*(HC(I,J) + AKAPTERM
000476 000476 *XT(I,J,2))
000477 000477 1
000478 000478 4299 CONTINUE
000479 000479 4299 CONTINUE
000480 000480 C
000481 000481 C
000482 000482 C INCREMENT TIME AND CYCLE
000483 000483 C
000484 000484 NCYCLE = NCYCLE + 1
000485 000485 TI = TI + DT
000486 000486 GO TO 900
000487 000487 C *****
000488 000488 C ***** END CYCLE *****
000489 000489 C *****
000490 000490 C
000491 000491 C
000492 000492 5100 CONTINUE
000493 000493 C
000494 000494 C CALL PRINT AND PLOT ROUTINES
000495 000495 C

```

98

000496
000497
000498
000499
000500
000501
000502
000503
000504
000505
000506
000507
000508
000509
000510
000511
000512

000496
000497
000498
000499
000500
000501
000502
000503
000504
000505
000506
000507
000508
000509
000510
000511
000512

CALL PRINT
CALL PLOTPK0

C

C CHECK FOR DUMP

C

IF (T1.0E.TF) GO TO 7000
TP = TP + TPI
GO TO 1100

C

C

7000 CONTINUE

CALL DROP
CALL OOTIM(ISEC)
CPTIME = FLOAT(ISEC) / 1.E6
WRITE (6,0100) CPTIME
CALL EXIT(1)

C

```

*****
000001 000517
000002 000518
000003 000519
000004 000520
000005 000521
000006 000522
000007 000523
000008 000524
000009 000525
000010 000526
000011 000527
000012 000528
000013 000529
000014 000530
000015 000531
000016 000532
000017 000533
000018 000534
000019 000535
000020 000536
000021 000537
000022 000538
000023 000539
000024 000540
000025 000541
000026 000542
000027 000543
000028 000544
000029 000545
000030 000546
000031 000547
000032 000548
000033 000549
000034 000550
000035 000551
000036 000552
000037 000553
000038 000554
000039 000555
000040 000556
000041 000557
000042 000558
000043 000559
000044 000560
000045 000561
000046 000562
000047 000563
000048 000564
000049 000565
000050 000566

```

```

CHAT 167D 0913414A M05/17/79
C
C
C
SURROUTINE PRINT
C
C
PARAMETER (NI=23)
PARAMETER (NJ=23)
PARAMETER (NK=10)
COMMON/ARQAYS/GAM(NI,NJ),HC(NI,NJ),PHI(NI,NJ),T(NI,NJ),
1 UWHAR(NI,NJ),VWHAR(NI,NJ),I(NI,NJ),
2 V(NI,NJ),W(NI,NJ),WNP1(NI,NJ),XT(NI,NJ,NK),
3 XINP1(NI,NJ,NK),J(NI,NJ),TNP1(NI,NJ),TRAR(NJ)
C
C
COMMON/VAR/AKAP0,AKAP0,RETA,RETA,NT,DTR,DV,DV,DZ,AKR,DYR,DZR,
1 DXSQ,DYSQ,DZSQ,DXSQ,DYSQ,DZSQ,EIGFIF,EPSTLON,GRAV,
2 IRAR,IRP1,TRP2,JBAR,IRP1,IRP2,
3 KRAR,KRP1,KRM1,NCYCLE,NIT,RNUA,TI,TF,TP,
4 TPC,TP1,THICK,TIN,TNOTHI,VIN,VOUT,W0,NC,XII,
5 RR,RW,RHOR,RHOW,AKR,OSIM,WMAX
C
C
C CALCULATE ENERGY DIFFERENCE
C
EDOTDIF = 0.0
DO 529 J = 2,JBP1
  UC = J(2,J)
  IF (UC.GT.0.0) TC = T(2,J)
  IF (UC.LE.0.0) TC = T(J,J)
  EDOTDIF = EDOTDIF - UC*TC*RWOW*RWOW*WNP1(J,J)
529 CONTINUE
C
C PRINT STATEMENTS
C
WRITE(6,9A00) TI,NCYCLE,NIT
WRITE(6,9A42)
WRITE(6,9A05) EDOTDIF,DQUH
WRITE(6,9A42)
IF (NCYCLE.GT.0.0) GO TO 5310
WRITE(6,9A10) DW,RR,RHOW,RHOR,AKR
WRITE(6,9A12) AKAP0,AKAP0,RETA,RETA,NT,DTR
WRITE(6,9A15) DX,DY,DZ,AKR,DYR,DZR
WRITE(6,9A17) DXSQ,DYSQ,DXSQ,DYSQ,DZSQ
WRITE(6,9A20) EIGFIF,EPSTLON,GRAV,TRAR,JBAR,KRAR
WRITE(6,9A22) RNUA,TI,TF,TP,TPC,TP1
WRITE(6,9A25) THICK,TIN,TNOTHI,VIN
WRITE(6,9A27) W0,NC,XII,NI,NJ,NK
5310 CONTINUE
C

```

```

100051 000567 WRITE(6,9240)
100052 000568 WRITE(6,9235)
100053 000569 WRITE(6,9240)
100054 000570 DO 5390 I = 1,IBP2
100055 000571 DU 5390 J = 1,JBP2
100056 000572 WRITE(6,9045) T(J,U(I,J),V(I,J),PHI(I,J),T(I,J),WNP1(I,J),
      1 GAM(I,J),U(I,J),XI(I,J,2),Y(I,J,3),YI(I,J,4)
5390 CONTINUE
      1 CUNTINUE
      1 WRITE(6,9240)
5390 CONTINUE
C
C PRINT OUT CURRENT TIME ON IFLETYPE WHEN RUNNING
C
      1 PRINT 9110,TI
      1 RETURN
C
C
C
C
C FORMATS
C
9000 FORMAT(14I,25X,6HTIME=,1PE12.5,10X,8HNCYCLE=,16,10X,8HNTT=,16)
9001 FORMAT(74#NOITNIF,E9.2,
      1 5X,8HDSUM=E9.2)
9010 FORMAT(34RW=E9.2,5X,3HAK=E9.2,5X,5HROWW=E9.2,3X,5HROWR=,
      1 E9.2,3X,4HAKR=E9.2)
9012 FORMAT(64AKAPR=E9.2,2X,6HAKAPR=E9.2,2X,8HBETA=E9.2,3X,
      1 6H3PTAR=E9.2,2X,3HNT=E9.2,5X,4HNTD=E9.2)
9014 FORMAT(34RX=E9.2,5X,3HNT=E9.2,5X,3HNTD=E9.2,5X,4HAKR=E9.2,
      1 4X,4HBYR=E9.2,4X,4HYZR=E9.2)
9017 FORMAT(54XSO=E9.2,3X,5HNYSO=E9.2,3X,4HNYSO=E9.2,2X,
      1 6HJVSQR=E9.2,2X,4HJZSQ=E9.2)
9020 FORMAT(74PIGFI=E9.2,1X,8HEPSYLON=E9.2,1X,5HGRAV=E9.2,
      1 2X,8HIBAR=E9.2,4X,8HIBAR=E9.2,4X,5HAKR=E9.2)
9022 FORMAT(54HNUA=E9.2,3X,3HTI=E9.2,2X,3HTF=E9.2,5X,3HTD=,
      1 E9.2,5X,4HTPC=E9.2,4X,4HTPI=E9.2)
9024 FORMAT(64THICK=E9.2,2X,3HTIN=E9.2,4X,7HTWOTH=E9.2,1X,
      1 4HVTN=E9.2)
9027 FORMAT(34W0=E9.2,5X,3HUC=E9.2,5X,4HXTI=E9.2,4X,3HNTI=16,
      1 8X,3HNJ=16,8X,3HNR=16)
9034 FORMAT(1X,8HCELL I J,8X,10H,10X,1HV,9X,3HNTI=9X,1HT,8X,4HWNPI,
      1 7X,3HOGAM,9X,1H,6X,9HXI(I,J,2),2X,9HYI(I,J,3),2X,
      2 9HXI(I,J,4))
9040 FORMAT(1X)
9042 FORMAT(1X)
9044 FORMAT(21A,2X,10(1PE10.3,1X))
9110 FORMAT(3CURRENT TIME IS 3,1PE12.5,/)
C
      1 END

```

```

*****
000001 000616
000002 000617
000003 000618
000004 000619
000005 000620
000006 000621
000007 000622
000008 000623
000009 000624
000010 000625
000011 000626
000012 000627
000013 000628
000014 000629
000015 000630
000016 000631
000017 000632
000018 000633
000019 000634
000020 000635
000021 000636

```

C
C

CHAT 1570 09136146 M05/17/79

SUBROUTINE UNDROP

```

LCM (BYNJIN), (FLMRJFC)
COMMON /BYNJLV/ IRSIZE, TDIFF (5000)
COMMON /FLMRJFC/ FLMRHF (5000)
DIMENSION IHEAD (6)
ISIZE = 4000
CALL FILM40
CALL IDENT80 (12, FLMRHF, 5000, 1100000, 105.0)
IHEAD (1) = 10M T3 RUTH
IHEAD (2) = 10MGE0 120
IHEAD (3) = 10MDC=30.
IHEAD (4) = 10MIRAR=0
IHEAD (5) = 10MRNUA=.01
IHEAD (6) = 10MVIN = 10.
CALL HEAD80 (12, 60, IHEAD, 10MROK T3RUTH)
CALL KEEPFLM (12)
RETURN
END

```

```

*****
000001 000637
000002 000638
000003 000639
000004 000640
000005 000641
000006 000642
000007 000643
000008 000644
000009 000645
000010 000646
000011 000647

```

C
C
C

CHAT 1570 09136146 M05/17/79

SUBROUTINE DROP

```

LCM (FLMRJFC)
COMMON /FLMRJFC/ FLMRHF (5000)
CALL FILM40
CALL IDENT80 (12, FLMRHF, 5000, 1100000, 105.0)
CALL KEEPFLM (12)
RETURN
END

```

```

000001 000648 C
000002 000649 C
000003 000650 SUBROUTINE PLOTPKG
000004 000651 C
000005 000652 C
000006 000653 PARAMETER (NT=23)
000007 000654 PARAMETER (N1=23)
000008 000655 PARAMETER (NK=10)
000009 000656 COMMON/AR0AYS/GAM(N1,NJ),MC(N1,NJ),PHI(N1,NJ),T(N1,NJ),
000010 000657 1 UWBAR(NT,NJ),VWBAR(NT,NJ),I(N1,NJ),
000011 000658 2 V(NT,NJ),W(N1,NJ),WNB1(NT,NJ),XT(NT,NJ,NK),
000012 000659 3 XIND1(NT,NJ,NK),J(NT,NJ),TNPD1(NT,NJ),YBAR(N1)
000013 000660 C
000014 000661 C
000015 000662 COMMON/VAR/AXAP0,AKAP0,RETA,RETA0,NT,NT0,DY,DY0,DZ,DXR,DYR,DZR,
000016 000663 1 DXSQ,DYSQ,DZSQ,DXSQ0,DYSQ0,DZSQ0,ETGFI,FPS,LON,RAV,
000017 000664 2 IRAR,IRP1,TR02,JHAR,JRP1,JRP2,
000018 000665 3 KRAR,KRP1,KRM1,NCYCLE,NT,RNUA,TI,YF,TP,
000019 000666 4 TPC,TPI,THIC,TIN,TNOHT,VIN,VOUT,W0,DC,XIT,
000020 000667 5 RQ,BW,RHOR,RHOW,AKR,NSIM,WMAX
000021 000668 C
000022 000669 C
000023 000670 COMMON/CJF07/IXL,IYR,IYT,IYR,XMN,XXM,YMX,YMN
000024 000671 DIMENSION CON(25),RCN(2),TIME(A),VAVE(5),V(1),Y(1)
000025 000672 EQUIVALENCE (UWBAR(1,1),X(1)),(VWBAR(1,1),V(1))
000026 000673 C
000027 000674 C
000028 000675 C CONTOUR PLOTS
000029 000676 C
000030 000677 IPLT=1
000031 000678 IFINISH=1
000032 000679 C
000033 000680 C SET COORDINATES FOR CELL CENTERED QUANTITIES
000034 000681 C (THE BOUNDARIES ARE UNCHANGED.)
000035 000682 C
000036 000683 95 CONTINUE
000037 000684 X(1)=0.
000038 000685 Y(1)=0.
000039 000686 X(2) = .5*DX
000040 000687 Y(2) = .5*DY
000041 000688 DO 100 I = 2,IRAR
000042 000689 X(I+1) = X(I) + DX
000043 000690 100 CONTINUE
000044 000691 C
000045 000692 DO 120 J = 2,JBAR
000046 000693 Y(J+1) = Y(J) + DY
000047 000694 120 CONTINUE
000048 000695 X(IRP2)=X(IRP1)+.5*DX
000049 000696 Y(IRP2)=Y(IRP1)+.5*DY
000050 000697 DO TV (125,155,195,215),IPLT
000051 000698 C
000052 000699 125 CONTINUE

```


00053	000700	C
00054	000701	C LABELS AND NUMBER OF CHARACTERS FOR AXES
00055	000702	RCD(1) = AXH-AXIS
00056	000703	RCD(2) = AHY-AXIS
		NXN = 4
		NYN = 6
000057	000704	C
000058	000705	C NO. OF CHARACTERS FOR FRAME TITLE
000059	000706	MZL = 40
000060	000707	C
000061	000708	C PLAT W
000062	000709	C
000063	000710	C WRITE LARGE CHARACTERS FOR TIME (REPLACES ENCODE)
000064	000711	C
000065	000712	C
000066	000713	WRITE(A3,000) TI
000067	000714	REWIND 63
000068	000715	READ(63,941) (TIME(I),I=1,4)
000069	000716	901 FORMAT(4A10)
000070	000717	900 FORMAT(1X,3CRACK WIDTH AT T= E.F12.5,10X)
000071	000718	C
000072	000719	C STORE CRACK WIDTH IN DIMMY ARRAY MC SUCH THAT THE PRIMARY CRACK
000073	000720	C IS EXCLUDED. (IT IS THE LAST COLUMN IN THE PRIMARY CRACK.)
000074	000721	J4 = .5*JPAR
000075	000722	DO 150 I = 2,JRP1
000076	000723	DO 150 J = 2,JRP1
000077	000724	MC(I,J) = JNP1(I,J)
000078	000725	IF(I,LP,2) MC(I,J)=WMAX
000079	000726	150 CONTINUE
000080	000727	C
000081	000728	C SET UPPER AND LOWER BOUNDARIES IN THE PRIMARY CRACK TO
000082	000729	C ADJACENT INNER VALUES
000083	000730	C SET UPPER AND LOWER BOUNDARIES TO ZERO OUTSIDE PRIMARY CRACK
000084	000731	DO 153 I=1,IAP2
000085	000732	IF(I,3T,2) GO TO 152
000086	000733	MC(I,1)=WMAX
000087	000734	MC(I,JRP2)=WMAX
000088	000735	GO TO 153
000089	000736	152 CONTINUE
000090	000737	MC(I,1)=0.
000091	000738	MC(I,JRP2)=0.
000092	000739	153 CONTINUE
000093	000740	C
000094	000741	C SET LEFT BOUNDARY WIDTH TO ADJACENT INNER
000095	000742	C SET RIGHT BOUNDARY WIDTH TO 0.
000096	000743	DO 154 J=1,JRP2
000097	000744	MC(1,J)=WMAX
000098	000745	MC(IHP2,J)=0.
000099	000746	154 CONTINUE
000100	000747	CALL CONTDJR(X,-IAP2,Y,-JRP2,MC(1,1),NT,NJ,10,0,0,0,-1,0
000101	000748	CON,DX,DY,1,TIME,MZL,6HX-AXIS,6HY-AXIS,6)
000102	000749	1
000103	000750	IPLT=IPLT.1
000104	000751	GO TO 325

```

000105 000752 C
000106 000753 C PLAT T
000107 000754 C
000108 000755 15= CONTINUE
000109 000756 REWIND 63
000110 000757 WRITE(63,210) T1
000111 000758 REWIND 63
000112 000759 READ(63,201) (TIME(I),I=1,4)
000113 000760 91= FORMAT(1X,EWATER TEMP AT T= F12.5,12Y)
000114 000761 C
000115 000762 C STORE WATER TEMPERATURE IN DUMMY ARRAY
000116 000763 DO 180 I=2,IAP1
000117 000764 DO 180 J=2,JAP1
000118 000765 HC(I,J)=T(I,J)
000119 000766 18= CONTINUE
000120 000767 C
000121 000768 C SET UPPER AND LOWER BOUNDARY TEMPERATURES OF PRIMARY CRACKS TO
000122 000769 ADJACENT INNER
000123 000770 C SET UPPER AND LOWER BOUNDARY TEMPERATURES OUTSIDE PRIMARY CRACK TO
000124 000771 TEMP AT INFINITY
000125 000772 HC(1,2) = HC(2,2)
000126 000773 HC(IMP2,2) = XII
000127 000774 HC(1,JAP1) = HC(2,IBP1)
000128 000775 HC(IMP2,JAP1) = XII
000129 000776 DO 183 I=1,IAP2
000130 000777 IF (I,37,2) GO TO 182
000131 000778 HC(I,1)=HC(I,2)
000132 000779 HC(I,JAP2)=HC(I,JAP1)
000133 000780 GO TO 183
000134 000781 182 CONTINUE
000135 000782 HC(I,1)=XII
000136 000783 HC(I,JAP2)=XII
000137 000784 184 CONTINUE
000138 000785 C
000139 000786 C SET LEFT BOUNDARY TEMPERATURES TO ADJACENT INNER
000140 000787 C SET RIGHT BOUNDARY TEMPERATURES TO TEMP AT INFINITE
000141 000788 DO 184 J=1,JAP2
000142 000789 HC(1,J)=HC(2,J)
000143 000790 HC(IMP2,J)=XII
000144 000791 184 CONTINUE
000145 000792 C
000146 000793 C SET SPECIAL CONTOUR MATRIX
000147 000794 NC = -10
000148 000795 NOC = -NC
000149 000796 CALL MIN4(HC,NI,IRP2,IRP2,I,J,7MIN)
000150 000797 CALL MAX4(HC,NI,IRP2,IRP2,I,J,7MAX)
000151 000798 DELZ = (ZMAX - ZMIN) / (NOC-1.)
000152 000799 C
000153 000800 C REMOVE BOUNDARY VALUES
000154 000801 ZMIN = ZMIN + DELZ
000155 000802 ZMAX = ZMAX - DELZ
000156 000803 DELZ = (ZMAX-ZMIN) / (NOC-1.)

```

```

000157 000804      CON(1) = 7MIN
000158 000805      DO 189 I = 2,NOC
000159 000806          CON(I) = CON(I-1) * NELZ
000160 000807      189 CONTINUE
000161 000808      C
000162 000809      199 CONTINUE
000163 000810          CALL CONTRJR(X,IRP2,Y,JRP2,HC(1,1),NY,NJ,NP,0.0,0.0,NFL7,
000164 000811              1      CON,DX,DY,1,TIME,MZL,6HY-AXIS,6,6HY-AXIS,6)
000165 000812              IPLT=IPLT+1
000166 000813              GO TO 325
000167 000814      C
000168 000815      C PLOT PHI
000169 000816      C
000170 000817      199 CONTINUE
000171 000818          REWIND 63
000172 000819          WRITE(63,020) TI
000173 000820          REWIND 63
000174 000821          READ(63,001) (TIME(I),I=1,4)
000175 000822      920 FORMAT(1X,EMHOIFEDN PRESSURE AT T= E,EP,5,15X)
000176 000823      C
000177 000824      C
000178 000825      C USE HC ARRAY TO PLOT PHIS. THIS ELIMINATES PLOTTING
000179 000826      C ZERO PRESSURES
000180 000827      C
000181 000828          DO 200 I = 2,IRP1
000182 000829          DO 200 J = 2,JRP1
000183 000830              HC(I,J) = PHI(I,J)
000184 000831              IF (JNP1(I,J).LT.W0) HC(I,J)=HC(I-1,J)
000185 000832      200 CONTINUE
000186 000833      C
000187 000834      C SET UPPER AND LOWER BOUNDARY PRESSURES TO ADJACENT INNER
000188 000835          HC(1,2) = HC(2,2)
000189 000836          HC(IRP2,2) = HC(IRP1,2)
000190 000837          HC(1,JRP1) = HC(2,JRP1)
000191 000838          HC(IRP2,JRP1) = HC(IRP1,JRP1)
000192 000839          DO 202 I=1,IRP2
000193 000840              HC(I,1)=HC(I,2)
000194 000841              HC(I,JRP2)=HC(I,JRP1)
000195 000842      202 CONTINUE
000196 000843      C
000197 000844      C SET LEFT AND RIGHT BOUNDARY PRESSURES TO ADJACENT INNER
000198 000845          DO 203 J=1,JRP1
000199 000846              HC(1,J)=HC(2,J)
000200 000847              HC(IRP2,J)=HC(IRP1,J)
000201 000848      203 CONTINUE
000202 000849          CALL CONTRJR(X,-IRP2,Y,-JRP2,HC(1,1),NY,NJ,10.0,0.0,-1.0,
000203 000850              1      CON,DX,DY,1,TIME,MZL,6HY-AXIS,6,6HY-AXIS,6)
000204 000851              IPLT=IPLT+1
000205 000852              GO TO 325
000206 000853      C
000207 000854      C
000208 000855      C

```

```

100209 000856 C
100210 000857 C VECTOR PLOTS
100211 000858 C
100212 000859 21x CONTINUE
100213 000860 IFINISH=2
100214 000861 C
100215 000862 C SET VELOCITY IN PRIMARY CRACK TO ZERO FOR PLOT PURPOSES
100216 000863 C
100217 000864 DO 258 J = 1,JRP1
100218 000865 V(2,J) = 1.E-10
100219 000866 25x CONTINUE
100220 000867 C
100221 000868 C NORMALIZATION
100222 000869 C
100223 000870 VELF = 0.0
100224 000871 DO 300 I = 2,IHP1
100225 000872 DU 301 J = 2,JRP1
100226 000873 STEET = SQRT((.5*(U(I+1,1)+U(I,J)))**2
100227 000874 +(.5*(V(I,J+1)+V(I,J)))**2)
100228 000875 VELF = AMAX1(STEET*VELF)
100229 000876 301 CONTINUE
100230 000877 30x CONTINUE
100231 000878 C
100232 000879 C VELF IS MAXIMUM VELOCITY COMPONENT IN SYSTEM
100233 000880 C PICK OUT THE MAXIMUM VELOCITY COMPONENT IN SYSTEM
100234 000881 C
100235 000882 DF = DX
100236 000883 IF (DX.GT.DY) DF = DY
100237 000884 VNORM = .40
100238 000885 VELF = VNORM*DF/VELF
100239 000886 C
100240 000887 C SET UP FRAME AND LABEL
100241 000888 C
100242 000889 C
100243 000890 C WRITE LARGE CHARACTERS FOR TIME
100244 000891 C
100245 000892 REWIND 63
100246 000893 WRITE(63,930) TI
100247 000894 REWIND 63
100248 000895 READ(63,941) (TIME(I),I=1,4)
100249 000896 930 FORMAT(1X,EVELOCITY AT T= E12.5,10X)
100250 000897 C
100251 000898 CALL ADV(1)
100252 000899 CALL FRAME(IXL,IXR,IYT,IYH)
100253 000900 CALL DGA(IXL,IXR,IYT,IYR,XMN,XMX,YMN)
100254 000901 C
100255 000902 C CENTER THE TITLE AND THE LABELS FOR THE VECTOR PLOTS
100256 000903 IX=MAX0(40,IXL+(IXR-IXL-12*M7L)/4)
100257 000904 LZL=MAX0(MZL-(1024-IX)/12,0)+1
100258 000905 IF (LZL.GT.1) LZL=(LZL+M3)/M5+1
100259 000906 CALL DLC4(IX,MAX0(10,IYT-74-16*LZL),MZL,TIME(1),2)
100260 000907 CALL DLC4(MAX0(54,IXI)+(IXH-IXL-12*MXX)/4,IYR+20,NYN,RCO(1),2)
100261 000908 CALL DLCV(78,M7NO(IYR+55,IYB-(IYR-IYT-12*MYN)/2),NYN,RCO(2),2)

```

```

100262 000909 C
100263 000910 C CALCULATE VELOCITY VECTORS
100264 000911 C
100265 000912 DO 320 J = 2, JBP1
100266 000913 Y1 = Y(J)
100267 000914 DO 320 I = 2, IRP1
100268 000915 X1 = X(I)
100269 000916 X2 = X1 + VELF*(III(I-1, J) + III(I, J))
100270 000917 Y2 = Y1 + VELF*(IV(I, J-1) + IV(I, J))
100271 000918 CALL CONVRT(X1, Y1, XMM, XMX, IXI, IYR)
100272 000919 CALL CONVRT(Y1, Y1, YMM, YMX, IYR, IYT)
100273 000920 CALL CONVRT(X2, Y2, XMM, XMX, IXL, IYR)
100274 000921 CALL CONVRT(Y2, Y2, YMM, YMX, IYR, IYT)
100275 000922 CALL DOV(IXI, IYI, IX2, IY2)
100276 000923 320 CONTINUE
100277 000924 C
100278 000925 C DETERMINE MAX SPEED AND ZONE
100279 000926 VMX = 0.
100280 000927 IMX = 1
100281 000928 JMX = 1
100282 000929 DO 342 I = 2, IRP1
100283 000930 DO 342 J = 2, IRP1
100284 000931 VAVE(1) = .5*(III(I-1, J) + III(I, J))
100285 000932 VAVE(2) = .5*(IV(I, J-1) + IV(I, J))
100286 000933 VAVE(1) = VAVE(1) + VAVE(1)
100287 000934 VAVE(2) = VAVE(2) + VAVE(2)
100288 000935 VIEST = SQRT(VAVE(1) + VAVE(2))
100289 000936 IF (VMX, GT, VIEST) GO TO 342
100290 000937 VMX = VIEST
100291 000938 IMX = I
100292 000939 JMX = J
100293 000940 342 CONTINUE
100294 000941 C
100295 000942 C RESET VELOCITY IN PRIMARY CRACK TO INLET VELOCITY
100296 000943 C
100297 000944 DO 398 J = 1, JBP1
100298 000945 V(2, J) = VIN
100299 000946 398 CONTINUE
100300 000947 C
100301 000948 C IDENTIFY MAX SPEED AND ZONE
100302 000949 REWIND 63
100303 000950 WRITE(63, 940) VMX, IMX, JMX
100304 000951 REWIND 63
100305 000952 READ(63, 941) (TIME(I), I=1, 6)
100306 000953 CALL DLCH(116, 1000, 60, TIME(1), 2)
100307 000954 940 FORMAT(1X, 19HMAXIMUM SPEED, I, J, 1, 1DE12.5, 210)
100308 000955 941 FORMAT(1X, 6A10)
100309 000956 C
100310 000957 C DRAW DOTIED LINE BETWEEN PRIMARY FRACTURE AND
100311 000958 C THERMAL STRESS FRACTURE
100312 000959 C

```

```

000313 000960 C SET UP X COORDINATES
000314 000961 329 CONTINUE
000315 000962 XF = X(2)*.5*DX
000316 000963 CALL CONVRT(XF,IX1,XMN,XXMX,IXL,IXR,
000317 000964 IX2 = IX1
000318 000965 C
000319 000966 C SET UP Y COORDINATES
000320 000967 JDOT = 5
000321 000968 DYDO! = DY/JDOT
000322 000969 YL = DYDO*.5
000323 000970 JDTO! = JMAR*JDOT+.1
000324 000971 Y(1) = 0.
000325 000972 DO 330 J = 1,JDTOY
000326 000973 Y(J+1) = Y(J)+DYDOT
000327 000974 330 CONTINUE
000328 000975 C
000329 000976 C PLT DOT5
000330 000977 DO 340 J = 1,JDTOY
000331 000978 Y1 = Y(J)
000332 000979 Y2 = Y(J)+YL
000333 000980 CALL CONVRT(Y1,Y1,YMN,YMX,YVH,YVT)
000334 000981 CALL CONVRT(Y2,Y2,YMN,YMX,YVH,YVT)
000335 000982 CALL DOV(IX1,IY1,IX2,IY2)
000336 000983 340 CONTINUE
000337 000984 GO TO (45,345),IFINISH
000338 000985 C
000339 000986 345 RETURN
000340 000987 END
*****
000001 000988 C
000002 000989 C
000003 000990 SURROUTINE PLTJHIX,Y,NPTS,INC,LNN,NSYM,C,YAA,YAA,LABEL7,NZ1,LABFLX
000004 000991 1,NXL,LAHELY,NYL)
000005 000992 C
000006 000993 C LABEL AXES, WRITE TITLES, DRAW BORDERS AND CONNECT CONTINUED POINTS
000007 000994 C ESTABLISHED IN TRTC.18
000008 000995 COMMON /C IE07/ IXL,IXR,IYI,IYB,XMN,XXMX,YMX,YMN
000009 000996 DIMENSION X(1), Y(1)
000010 000997 INTEGER 30IDF
000011 000998 B=AMAX1(AMAX1(C,0.)*(1,NXL+1),0.)
000012 000999 LIN=LNN
000013 001000 KSYM=IARS(NSYM)
000014 001001 KINC=MAX0(IARS(INC),1)
000015 001002 C
000016 001003 C ESTABLISH NO. CHARACTERS AND NO. POINTS
000017 001004 NPTS=IARS(NPTS)
000018 001005 NZL=IARS(NZL)
000019 001006 XKA=ABS(XAA)
000020 001007 YYA=ABS(YAA)
000021 001008 NXN=IARS(NXL)
000022 001009 NYN=IARS(NYL)
000023 001010 GRIDP=AMAX1(1.,ARS(C))
000024 001011 IF (NSYM*LT.0) GO TO 130

```

```

000025 001012 C
000026 001013 C ADVANCE FRAME AND DETERMINE MAX AND MIN FOR AXES
000027 001014 CALL ADV (1)
000028 001015 CALL MAXV (X,KINC,MPTS,YSIR,XXM)
000029 001016 CALL MAXV (Y,KINC,MPTS,YSIR,YMY)
000030 001017 CALL MINV (X,KINC,MPTS,YSIR,XXM)
000031 001018 CALL MINV (Y,KINC,MPTS,YSIR,YMY)
000032 001019 C
000033 001020 C IF XAX AND/OR YAX ARE NON-ZERO THE LENGTHS
000034 001021 C WILL BE CONSIDERED AS RATIOS WHERE THE LONGEST
000035 001022 C SIDE IS FITTED ON A RAD POINT LINE.
000036 001023 C DETERMINE ASPECT RATIO OF ROUNDER
000037 001024 IF (XAX.EQ.0) XAX=1.
000038 001025 IF (YAX.EQ.0) YAX=1.
000039 001026 DXM=.001*ABS(XXM)
000040 001027 IF (DXM.EQ.0) DXM=.0001
000041 001028 XMY=XMY-DXM
000042 001029 XMX=XMX+DXM
000043 001030 60 IF (SIGN(1,NYL).LT.0.AND. INC.OT.0) YAX=(YMY-XMY)/YVA
000044 001031 IF (SIGN(1,NXL).LT.0.AND. NPTS.OT.0) XAX=(XMX-XMY)/XAX
000045 001032 C
000046 001033 C DETERMINE 4020 COORDINATES OF ROUNDER
000047 001034 IXL=120
000048 001035 IYR=Y10
000049 001036 FACT=800./AMAX1(XAX,YAX)
000050 001037 IXR=120+IFIX(FACT*XAX)
000051 001038 IYT=Y10+IFIX(FACT*YAX)
000052 001039 CALL FRAME (IXL,IXR,IYT,IYH)
000053 001040 IF (SIGN(1,XAX).GT.0) GO TO 70
000054 001041 SWAP=XMY
000055 001042 XMY=XMX
000056 001043 XMX=SWAP
000057 001044 70 IF (SIGN(1,YAX).GT.0) GO TO 80
000058 001045 SWAP=XMY
000059 001046 YMY=XMX
000060 001047 XMX=SWAP
000061 001048 80 CALL DGA (IXL,IXR,IYT,IYH,XMY,XMX,YMY,XMX)
000062 001049 C
000063 001050 C CENTER TITLE AND LABEL AXES
000064 001051 IFI (NXN.NP.0) CALL DICH (MAX0(54,IYL+(IXR-IXL-12*NYN)/2),IYR+24,
000065 001052 1,NYN,LABEL(X,2)
000066 001053 INCX=70
000067 001054 IF (NYN.GT.0) INCX=10
000068 001055 IFI (NYN.NP.0) CALL DICH (INCX,MIN0(IYR+52,IYR-(IYR-IYT-12*NYN)/2)
000069 001056 1,NYN,LABEL(Y,2)
000070 001057 IX=MAX0(40,IXL+(IXR-IXL-12*NYN)/2)
000071 001058 LZL=MAX0(4ZL-(1024-IX)/12,0)+1
000072 001059 IF (LZL.GT.1) LZL=(LZL+0.5)/0.5+1
000073 001060 IFI (NZL.NP.0) CALL DICH (IX,MAX0(10,IYT-24-16*LZL),4Z1,LABEL(Z,2)
000074 001061 C
000075 001062 C INCREASE EXPOSURE
000076 001063 CALL EXH
000077 001064 IF (NZL.LP.0) GO TO 220

```

```

000076 001065 C
000079 001066 C PLOT POINTS AND/OR LINE
000080 001067 13a MPTS=MPTS+KINC
000081 001068 DO 210 NXP=1,MPTS,KINC
000082 001069 XTWO=X(NXP)
000083 001070 YTHO=Y(NXP)
000084 001071 IF (MPTS=1) XTWO=ALOG10(XTHO)
000085 001072 IF (INC=0) YTHO=ALOG10(YTHO)
000086 001073 CALL CONVST (XTWO,NXTWO,XMN,XMX,IXI,IXR)
000087 001074 CALL CONVST (YTHO,NYTHO,YMN,YMX,IYR,IYT)
000088 001075 IF (NXP=EN.1) GO TO 190
000089 001076 IF (LIN=3F.0) GO TO 140
000090 001077 14a IF (MOD((NXP-1)/KINC),TARS(LIN),NF.0) GO TO 150
000091 001078 C
000092 001079 C LIGHTEN EXPOSURE
000093 001080 CALL EXL
000094 001081 C
000095 001082 C REMOVES CONTOUR LABELS
000096 001083 C CALL DLCH (NXTWO,NYTHO,NKSYM,1)
000097 001084 CALL EXH
000098 001085 GO TO 200
000099 001086 15a IF (M=EQ.0) GO TO 200
000100 001087 16a DO 170 I3=1,4
000101 001088 17a CALL PLT (NXTWO,NYTHO,42)
000102 001089 GO TO 200
000103 001090 18a IF (M=EQ.0) CALL ORV (NAGNE,NYONE,NXTWO,NYTHO)
000104 001091 19a IF (LIN=VF.0) GO TO 140
000105 001092 IF (M=NF.0) GO TO 140
000106 001093 20a NYONE=NYTHO
000107 001094 NXONE=NXTWO
000108 001095 21a CONTINUE
000109 001096 22a RETURN
000110 001097 END
000111 001098 CHAT 1670 05136146 M05/17/79
000112 001099 C
000113 001100 C
000114 001101 SURROUTINE CONTHJR(X,NX,Y,NY,Z,NZX,NZY,NF,ZMN,ZMX,NL,ZPLAN,DMPX
000115 001102 1,DMPY,IGRN,ITITLE,NTITLE,XLABEL,NX1,RL,YLABEL,NY1,RL)
000116 001103 C
000117 001104 C ESTABLISH THE CONTOUR SCHEME TO BE USED BY TRICJR
000118 001105 COMMON /CONTRCON/ ISYM(50)
000119 001106 DIMENSION XSCALE(2), YSCALE(2)
000120 001107 EQUIVALENCE (XMIN,XSCALE(1)), (XMAX,XSCALE(2))
000121 001108 EQUIVALENCE (YMIN,YSCALE(1)), (YMAX,YSCALE(2))
000122 001109 DIMENSION X(1), Y(1), Z(NZX, ), ZPLAN(1)
000123 001110 DIMENSION ZT(4)
000124 001111 DIMENSION CON(6)
000125 001112 NDC=MIND(TARS(INC),50)
000126 001113 ZMIN=ZMN
000127 001114 ZMAX=ZMX
000128 001115 DELZ=ULZ
000129 001116 DMAPA=DMPX

```



```

000020 001117      DMAPY=DMAPY
000021 001118      NOX=IARS(UNX)
000022 001119      NOY=IARS(VNY)
000023 001120      DO 10 I=1,50
000024 001121      10 ISYM(I)=0
000025 001122      C
000026 001123      C ESTABLISH SCALES
000027 001124      CALL MINV (X,1,NOX,I,XMIN)
000028 001125      CALL MAXV (Y,1,NOY,I,YMAX)
000029 001126      CALL MAXV (X,1,NOX,I,XMAX)
000030 001127      CALL MINV (Y,1,NOY,I,YMIN)
000031 001128      FORD=0.
000032 001129      IF (IORD,AT,0) FORD=-FORD
000033 001130      CALL PLOJA (XSCALE,YSCALE,2,1,1,1,FORD,DMAPX,DMAPY,TITLE,-NTITLE
000034 001131      1,XLABEL,YLABEL,YLABEL,NY(0))
000035 001132      IF (NC,LT,0) GO TO 50
000036 001133      C
000037 001134      C DETERMINE MAX-MIN CONTOURS AND SPACING
000038 001135      C STORE IN ARRAY ZPLAN
000039 001136      IF (NNX,LP,0) CALL MINM (Z,NZX,NOX,NY,I,1,ZMIN)
000040 001137      IF (NNY,LP,0) CALL MAXM (Z,NZX,NOX,NY,I,1,ZMAX)
000041 001138      IF (DELZ,AT,0) GO TO 20
000042 001139      DELZ=(ZMAX-ZMIN)/(NOC-1.)
000043 001140      20 IF (NZY,3T,0) GO TO 30
000044 001141      ZMAX=ZMAX-AMOD(ZMAX,DELZ)
000045 001142      ZMIN=ZMIN-AMOD(ZMIN,DELZ)
000046 001143      NOC=MIN0(NOC,IFIX((ZMAX-ZMIN)/DELZ+1,0))
000047 001144      30 ZPLAN(1)=ZMIN
000048 001145      DO 40 I=2,NOC
000049 001146      40 ZPLAN(I)=ZPLAN(I-1)+DELZ
000050 001147      C
000051 001148      C STORE THE ARRAY BEING CONTOURED IN ZT AND CONTOUR ONE CELL AT A TIME
000052 001149      50 DO 90 NY=1,NY
000053 001150      IX=MUD(NY,1)
000054 001151      OY=Y(NY)-Y(NY-1)
000055 001152      DO 80 IX=1,NX
000056 001153      NX=IX
000057 001154      IF (IX,NE,0) NX=NOX-IX+2
000058 001155      ZT(1)=Z(VY-1,NY-1)
000059 001156      ZT(2)=Z(VY,NY-1)
000060 001157      ZT(3)=Z(VY,NY)
000061 001158      ZT(4)=Z(VY-1,NY)
000062 001159      DX=X(NX)-X(NX-1)
000063 001160      IF (ABS(ZT(3)-ZT(1))-ABS(ZT(4)-ZT(2))) 70,40,40
000064 001161      60 CALL TRICIR (X(NX),Y(NY),-DX,-OY,NOC,ZPLAN,ZT(4),ZT(3),ZT(2))
000065 001162      CALL TRICIR (X(NX-1),Y(NY-1),DX,OY,NOC,ZPLAN,ZT(2),ZT(1),ZT(4))
000066 001163      GO TO 80
000067 001164      70 CALL TRICIR (X(NX-1),Y(NY),DX,-OY,NOC,ZPLAN,ZT(3),ZT(4),ZT(1))
000068 001165      CALL TRICIR (X(NX),Y(NY-1),-DX,OY,NOC,ZPLAN,ZT(1),ZT(2),ZT(3))
000069 001166      80 CONTINUE
000070 001167      90 CONTINUE

```

```

000071 001168
000072 001169
000073 001170
000074 001171
000075 001172
000076 001173
000077 001174
000078 001175
000079 001176
000080 001177
000081 001178
000082 001179
*****
000001 001180
000002 001181
000003 001182
000004 001183
000005 001184
000006 001185
000007 001186
000008 001187
000009 001188
000010 001189
000011 001190
000012 001191
000013 001192
000014 001193
000015 001194
000016 001195
000017 001196
000018 001197
000019 001198
000020 001199
000021 001200
000022 001201
000023 001202
000024 001203
000025 001204
000026 001205
000027 001206
000028 001207
000029 001208
000030 001209
000031 001210
000032 001211
000033 001212
000034 001213
000035 001214
000036 001215
000037 001216
000038 001217
000039 001218
000040 001219

C
C IDENTIFY THE MAX, MIN, AND SPACING FOR CONTINUOUS
  REMIND 63
  WRITE(63,000) ZPLAN(1),ZPLAN(NOC),DELZ
  REMIND 63
  READ(63,951) (CON(I),I=1,6)
  CALL DLC4(48,1000,60,CON(1),2)
900 FORMAT(1X,15HMIN,MAX,DELTA: ,103E12.4)
901 FORMAT(1X,6A10)
  RETURN

C
  END
  CHAT 1570 09136146 H05/17/79

C
C
C SURROUTINE TRICUB(X,Y,DY,DY,NOC,ZPLAN,7X,7Y,ZV)
C
C
C PERFORM ACTUAL INTERPOLATIONS NEEDED TO PRODUCE CONTINUOUS
C DEFINED IN CONTRJR. PLANR USED TO CONNECT POINTS.
C
  COMMON /CONTRCOM/ ISYM(50)
  DIMENSION XP(2,50), YP(2,50), ZT(4), ZPLAN(1)
  ZT(1)=ZX
  ZT(2)=ZV
  ZT(3)=ZY
  ZT(4)=ZX
  ZTMIN=AMIN1(ZT(1),ZT(2),ZT(3))
  ZTMAX=AMAX1(ZT(1),ZT(2),ZT(3))
  IMIN=NOC+1
  IMAX=0
  DO 10 K=1,NOC
    J=NOC-K+1
    IF (ZPLAN(J).GE.ZTMIN) IMIN=J
    IF (ZPLAN(K).LE.ZTMAX) IMAX=K
10 CONTINUE
  INT=IMAX-IMIN
  IF (INT.LE.0.OR.ZTMIN.EQ.ZTMAX) GO TO 130
  I2=1
  DO 110 K=1,3
    ZTMAX=AMAX1(ZT(K),ZT(K+1))
    ZPMIN=AMIN1(ZT(K),ZT(K+1))
    MIN=NOC+1
    MAX=0
    DO 20 J=1,NOC
      INZ=NOC-J+1
      IF (ZPLAN(J).GT.ZPMIN.OR.(ZPLAN(INZ).EQ.ZPMIN.AND.ZTMIN.FO.ZPMIN
1) MIN=INZ
      IF (ZPLAN(J).LE.ZTMAX) MAX=J
20 CONTINUE
  I42=MAX-41N
  IF (INZ.LE.0.OR.ZTMAX.FO.ZPMIN) GO TO 110
  IF (INZ-INT) 40,30,40

```

```

000041 001220
000042 001221
000043 001222
000044 001223
000045 001224
000046 001225
000047 001226
000048 001227
000049 001228
000050 001229
000051 001230
000052 001231
000053 001232
000054 001233
000055 001234
000056 001235
000057 001236
000058 001237
000059 001238
000060 001239
000061 001240
000062 001241
000063 001242
000064 001243
000065 001244

```

```

30 GO TO (50,40), IZ
40 IZ=1
   GO TO 60
50 IZ=2
60 DO 100 J=MIN,MAX
   GO TO (70,80,90), K
70 XP(IZ,J)=V+DX*(ZPLAN(J)-ZT(2))/(ZT(1)-ZT(3))
   YP(IZ,J)=V
   GO TO 100
80 XP(IZ,J)=V
   YP(IZ,J)=V+DY*(ZPLAN(J)-ZT(2))/(ZT(3)-ZT(1))
   GO TO 100
90 XZ(IZ,J)=V+DX*(ZPLAN(J)-ZT(3))/(ZT(1)-ZT(2))
   YP(IZ,J)=V+DY*(ZPLAN(J)-ZT(1))/(ZT(3)-ZT(2))
100 CONTINUE
110 CONTINUE
   DO 120 J=MIN,IMAX
   ISYM(J)=ISYM(J)+1
   L=3
   IF (MOD(ISYM(J),10).NE.1) L=0
   CALL PL0JA (XP(1,J),YP(1,J),2,1,L,.1,0,.0,.0,.0,.0,.0,.0)
120 CONTINUE
C
130 RETURN
   END

```

APPENDIX C
CODE LISTING FOR GEØ-III

Example Number 6

Figures 9-19

LASL Identification Number LP-1094

```

*****
000001 000001 C
000002 000002 C VARIARIF DEFINITIONS
000003 000003 C
000004 000004 C
000005 000005 C AKAPR - KAPPA ROCK THERMOELECTRIC CONDUCTIVITY OF ROCK
000006 000006 C AKAPN - RATIO OF THERMAL CONDUCTIVITY IN ROCK TO THE
000007 000007 C PRODUCT OF THE DENSITY OF WATER AND VOLUMETRIC
000008 000008 C EXPANSION COEFFICIENT FOR WATER
000009 000009 C BETA - BETA WATER- VOLUMETRIC EXPANSION COEFFICIENT FOR WATER
000010 000010 C BETAR - BETA ROCK- LINEAR EXPANSION COEFFICIENT FOR ROCK
000011 000011 C DC - DISTANCE BETWEEN SECONDARY CHACKS
000012 000012 C DT - DELTA T, CHANGE IN THE ELAPSED TIME
000013 000013 C DX - CHANGE IN THE DISTANCE IN X DIRECTION
000014 000014 C DY - CHANGE IN THE DISTANCE IN Y DIRECTION
000015 000015 C DZ - CHANGE IN THE DISTANCE IN Z DIRECTION
000016 000016 C GAM - NEWTON RAPHSOON RELAXATION COEFFICIENT
000017 000017 C GRAV - ACCELERATION OF GRAVITY
000018 000018 C HEIGHT - A MEASURE OF INITIAL HALF CRACK HEIGHT
000019 000019 C HC - HEAT CONVECTION TERMS
000020 000020 C IBAR - RIGHT BOUNDARY OF TOTAL MESH IN X DIRECTION
000021 000021 C JBAR - TOP BOUNDARY OF MESH IN Y DIRECTION
000022 000022 C KBAR - TOP BOUNDARY OF MESH IN Z DIRECTION
000023 000023 C PHI - PRESSURE/ DENSITY
000024 000024 C RNU - COEFFICIENT OF KINEMATIC VISCOSITY
000025 000025 C RNUA - COEFFICIENT OF ARTIFICIAL KINEMATIC VISCOSITY
000026 000026 C T - WATER TEMPERATURE
000027 000027 C NCYF - TOTAL CALCULATION TIME, BASED ON CYCLE
000028 000028 C THICK - THICKNESS OF INITIAL COOLING ZONE IN THE ROCK
000029 000029 C TI - RUNNING PROBLEM TIME WITHIN EACH HYDRO CYCLE
000030 000030 C TTI - TOTAL ELAPSED TIME
000031 000031 C TIN - WATER TEMPERATURE AT INLET
000032 000032 C NCYP - TIME TO PRINT, BASED ON CYCLE
000033 000033 C NCYI - TIME PRINT INTERVAL, BASED ON CYCLE
000034 000034 C U - VELOCITY IN X DIRECTION
000035 000035 C V - VELOCITY IN Y DIRECTION

```

```

000036 000036 C VIN - VFLACITY AT INLET
000037 000037 C VOUT - VELOCITY AT OUTLET
000038 000038 C W - CRACK WIDTH
000039 000039 C W0 - YIELD WIDTH FOR CRACK PRODUCTION
000040 000040 C XI - ROCK TEMPERATURE
000041 000041 C XII - ROCK TEMPERATURE AT INFINITY
000042 000042 C
000043 000043 C
000044 000044 C
000045 000045 C PROGRAM GFO(OUTPUT,GE00U!,TAPE6=GE00OUT,TAPE63=100B)
000046 000046 C
000047 000047 C
000048 000048 C PARAMETER (NI=23)
000049 000049 C PARAMETER (NJ=23)
000050 000050 C PARAMETER (NK=10)
000051 000051 C COMMON/ARAYS/AM(NI,NJ),HC(NI,NJ),PHI(NI,NJ),T(NI,NJ),
000052 000052 1 IWBAR(NI,NJ),VWRAP(NI,NJ),II(NI,NJ),
000053 000053 2 V(NI,NJ),W(NI,NJ),WNP,(NI,NJ),XI(NI,NJ,NK),
000054 000054 3 XINP,(NI,NJ,NK),D(NI,NJ),TNP,(NI,NJ),TRAR(NJ)
000055 000055 C
000056 000056 C
000057 000057 C COMMON/VAR/AKAP0,AKAP0,HELA,BETAR,DT,DTR,DX,DY,DZ,DXR,DYR,DZR,
000058 000058 1 NX50,DY50,N50,NYSOR,DYSOR,DZ50R,EIGFIF,EPSILON,GRAB,
000059 000059 2 IRAR,IRP,JRP2,IRAR,JRP,JRP2,HEIGHT,
000060 000060 3 KRAR,KBP,WRM1,NCYCLF,NIT,RNUA,II,TTI,NCYF,NCYP,
000061 000061 4 NCYT,THICK,TIN,TWOTHY,VIN,VOUT,W,DC,XII,
000062 000062 5 BR,RW,RHOR,RHOW,AKR,DSUM,WMAX,NCYTER,NCYTERI
000063 000063 C
000064 000064 C
000065 000065 C CALL IINGCWR
000066 000066 C CALL IINDRNP
000067 000067 C
000068 000068 C
000069 000069 C TIMING ST,DIE
000070 000070 C
000071 000071 C CALL NNTIM
000072 000072 C ISEC = 0
000073 000073 C
000074 000074 C
000075 000075 C
000076 000076 C VARIABLES
000077 000077 C
000078 000078 B = 1.0
000079 000079 BR = 0.25
000080 000080 RHOW = 1.0
000081 000081 RHOR = 2.65
000082 000082 AKR = 6.2E-3
000083 000083 AKAP0 = AKR / (RHOW*BW)
000084 000084 AKAPR = AKR / (RHOR*BR)
000085 000085 BETA = 2.7E-4
000086 000086 BETAR = 2.5E-5
000087 000087 DC = 50.0
000088 000088 DT = 0.0

```

000089	000089	KBAR = 4
000090	000090	KBP1 = KBAR+1
000091	000091	KBM1 = KBAR-1
000092	000092	DX = 25.0
000093	000093	DY = 15.0
000094	000094	DZ = 0.5*DC/LOAT(KBM1)
000095	000095	DXR = 1./DX
000096	000096	DYR = 1./DY
000097	000097	DZR = 1./DZ
000098	000098	DXSQ = DX**2
000099	000099	DYSQ = DY**2
000100	000100	DZSQ = DZ**2
000101	000101	DXSQR = 1./DXSQ
000102	000102	DYSQR = 1./DYSQ
000103	000103	DZSQR = 1./DZSQ
000104	000104	DTXI = 0.4/(AKAPR*(DXSQR+DYSQR+DZSQR))
000105	000105	EIGFIF = A./TS.
000106	000106	EPSILON = 2.E-3
000107	000107	GRAV = 980.0
000108	000108	HEIGHT = 1.F*10
000109	000109	IBAR = 20
000110	000110	IBP1 = IBAR+1
000111	000111	IBP2 = IBAR+2
000112	000112	JBAR = 20
000113	000113	JBP1 = JBAR+1
000114	000114	JBP2 = JBAR+2
000115	000115	NEW = 0
000116	000116	NEWF = 15
000117	000117	NAIF = 100
000118	000118	NCVF = 1000
000119	000119	NCVI = 1000
000120	000120	NCYTERI = 50
000121	000121	RNUA = 0.01
000122	000122	THICK = 50.0
000123	000123	TI1 = 0.0
000124	000124	TIN = 150.0
000125	000125	TWOTHY = 2./A.
000126	000126	VIN = 0.0
000127	000127	WD = 0.0
000128	000128	XI1 = 300.0
000129	000129	C
000130	000130	GO TO 170
000131	000131	C
000132	000132	C REZONE VARIABLES PACKAGE
000133	000133	C
000134	000134	100 CONTINUE
000135	000135	NEW = NEW + 1
000136	000136	C
000137	000137	C PRINT OUT CURRENT VALUE OF NEW ON TERMINAL WHILE RUNNING
000138	000138	C
000139	000139	PRINT 9115,NEW
000140	000140	9115 FORMAT(//.4HNEW=,I4,//)

```

000141 000141
000142 000142
000143 000143
000144 000144
000145 000145
000146 000146
000147 000147
000148 000148
000149 000149
000150 000150
000151 000151
000152 000152
000153 000153
000154 000154
000155 000155
000156 000156
000157 000157
000158 000158
000159 000159
000160 000160
000161 000161
000162 000162
000163 000163
000164 000164
000165 000165
000166 000166
000167 000167
000168 000168
000169 000169
000170 000170
000171 000171
000172 000172
000173 000173
000174 000174
000175 000175
000176 000176
000177 000177
000178 000178
000179 000179
000180 000180
000181 000181
000182 000182
000183 000183
000184 000184
000185 000185
000186 000186
000187 000187
000188 000188
000189 000189
000190 000190
000191 000191

```

```

C NEW COUNTS NUMBER OF HYDRO-CYCLE GROUPS
  IF (NEW,OT,NEW) GO TO 7000
170 CONTINUE
  IF (NEW,NE,0) GO TO 380
C
C INITIAL CONDITIONS
C
  DO 199 I = 2,IBAR
    DO 198 J = 2,JBP1
      U(I,J) = 0.0
198 CONTINUE
199 CONTINUE
C
  DO 299 I = 3,IBP1
    DO 298 J = 2,JBAR
      V(I,J) = 0.0
      V(2,J) = VIN
298 CONTINUE
299 CONTINUE
C
C
  V(2,1) = VIN
  V(2,JBP1) = VIN
C
C CALCULATE INDEX FOR THE DOWN SIDE OF MIDDLE
C
  HJBAR = FLOAT(,IBAR)*.5 + 1.01
  JM = HJBAR
  DO 379 I = 3,IBP1
    DO 378 J = 2,JBP1
      DO 377 K = 1,KBAR
        YBAR = (FLOAT(I-2) - 0.5)*DX/THICK
        YBAR = (FLOAT(J-JM)*DY)/HEIGHT
        RAD15 = SQRT(XBAR**2 + YBAR**2)
        Y(I,J,K) = T1N + (XII-T1N)*ERF(RADIUS)
377 CONTINUE
378 CONTINUE
379 CONTINUE
C
  NEW = 1
  GO TO 430
C
C XI UPDATE PACKAGE - TO ALLOW ROCK TEMP UPDATE
C
380 CONTINUE
  NXI = 0
381 CONTINUE
  NXI = NXI + 1
C
C BOUNDARY CONDITIONS ON XI
C

```

```

000192 000192 DO 389 I = 3, IAP1
000193 000193 DO 387 K = 1, KBP1
000194 000194 XI(I, JAP2, K) = XI(I, JAP1, K)
000195 000195 XI(I, J1, K) = XI(I, 2, K)
000196 000196 387 CONTINUE
000197 000197 389 CONTINUE
000198 000198 C
000199 000199 C
000200 000200 C BOUNDARY CONDITIONS ON XI
000201 000201 C
000202 000202 DO 394 J = 2, JBP1
000203 000203 DO 393 K = 1, KBP1
000204 000204 XI(IAP2, J, K) = XI(I, J, K)
000205 000205 393 CONTINUE
000206 000206 394 CONTINUE
000207 000207 C
000208 000208 C
000209 000209 DO 399 I = 3, IAP1
000210 000210 DO 398 J = 2, JBP1
000211 000211 XI(I, J, 1) = Y(I, J)
000212 000212 XI(I, J, KBP1) = Y(I, J, KBAP)
000213 000213 398 CONTINUE
000214 000214 399 CONTINUE
000215 000215 C
000216 000216 DO 407 J = 2, JBP1
000217 000217 DO 407 K = 1, KBP1
000218 000218 XI(2, J, K) = 2.0*YIN - XI(3, J, K)
000219 000219 407 CONTINUE
000220 000220 408 CONTINUE
000221 000221 C
000222 000222 C CALCULATE NEW XI
000223 000223 C
000224 000224 DO 419 I = 3, IAP1
000225 000225 DO 418 J = 2, JBP1
000226 000226 DO 417 K = 2, KBAP
000227 000227 XINP1(I, J, K) = XI(I, J, K) + AKAP*DTXI*
000228 000228 1. (CXSUR*(XI(I+1, J, K) + XI(I-1, J, K) - 2.0*XI(I, J, K))
000229 000229 2. + DYSQR*(XI(I, J+1, K) + XI(I, J-1, K) - 2.0*XI(I, J, K))
000230 000230 3. + DZSQR*(XI(I, J, K+1) + XI(I, J, K-1) - 2.0*XI(I, J, K)))
000231 000231 417 CONTINUE
000232 000232 418 CONTINUE
000233 000233 419 CONTINUE
000234 000234 C
000235 000235 C STORE NEW XI INTO OLD
000236 000236 C
000237 000237 DO 429 I = 3, IAP1
000238 000238 DO 428 J = 2, JBP1
000239 000239 DO 427 K = 2, KBAP
000240 000240 XI(I, J, K) = XINP1(I, J, K)
000241 000241 427 CONTINUE
000242 000242 428 CONTINUE
000243 000243 429 CONTINUE

```



```

000244 000244
000245 000245
000246 000246
000247 000247
000248 000248
000249 000249
000250 000250
000251 000251
000252 000252
000253 000253
000254 000254
000255 000255
000256 000256
000257 000257
000258 000258
000259 000259
000260 000260
000261 000261
000262 000262
000263 000263
000264 000264
000265 000265
000266 000266
000267 000267
000268 000268
000269 000269
000270 000270
000271 000271
000272 000272
000273 000273
000274 000274
000275 000275
000276 000276
000277 000277
000278 000278
000279 000279
000280 000280
000281 000281
000282 000282
000283 000283
000284 000284
000285 000285
000286 000286
000287 000287
000288 000288
000289 000289
000290 000290
000291 000291
000292 000292
000293 000293
000294 000294
000295 000295

```

```

C      TTI = TTI + DT * I
C
      IF (N * I .LT. N * IF) GO TO 781
430 CONTINUE
C
C      DO 439 I = 3, IAP1
      DO 438 J = 2, JBP1
        T(I,J) = XI(I,J,T)
        PHI(I,J) = 0.0
        T(2,J) = TIN
        PHI(2,J) = 0.0
438 CONTINUE
439 CONTINUE
C
C CALCULATE W FOR ZERO PRINT AND TO ESTABLISH OLD W
C
C      DO 599 I = 3, IAP1
      DO 598 J = 2, JBP1
        SUM1 = 0.0
        DO 597 K = 2, KBP1
          SUM1 = SUM1 + (AT(I,J,K) - XII)
597 CONTINUE
        WNP1(I,J) = -2.0 * BETAR * DT * (.5 * (XI(I,J,1) - XII) + SUM1)
        IF (WNP1(I,J) .LE. W0) WNP1(I,J) = 0.0
        W(I,J) = WNP1(I,J)
598 CONTINUE
599 CONTINUE
C
      WMAX = 0.0
      DO 658 J = 2, JBP1
        W(2,J) = W(2,J)
        WNP1(2,J) = WNP1(3,J)
        WTEST = WNP1(3,J)
        WMAX = AMAX1(WTEST, WMAX)
658 CONTINUE
C
C CALCULATE DT AND SET UP FINISH TIMES
C
      QUFST = GRAV * BETAR * (TIN - 3.0) * (XII - TIN) / 17.856
      DEPSH = EPS10N * QUEST * NYH
      VTEST = (WMAX * 2) * QUFST
      IF (DT * NE * 0.0) GO TO 700
      DT = (.2 * SQRT(DX * 2 + DY * 2)) / (VTEST * ARS(VIN))
700 CONTINUE
      DTR = 1. / DT
      TI = 0.0
      NCYP = 0
      NCYTR = 1

```

```

000296 000296 C
000297 000297 C
000298 000298 C SET CYCLE COUNTERS
000299 000299 C
000300 000300 NCYCLP = 2
000301 000301 C
000302 000302 C*****
000303 000303 C*****REFIN CYCLE *****
000304 000304 C*****
000305 000305 C
000306 000306 900 CONTINUE
000307 000307 C
000308 000308 C
000309 000309 C CHECK FOR PRINTING INTERVAL
000310 000310 C
000311 000311 IF (NCYCLE.EQ.NCYTER) GOTO 1000
000312 000312 1000 CONTINUE
000313 000313 C
000314 000314 IF (NCYCLE.EQ.NCYP) GO TO 5100
000315 000315 C
000316 000316 C
000317 000317 C
000318 000318 1100 CONTINUE
000319 000319 IF (NCYCLE.EQ.NCYF) GO TO 1200
000320 000320 C
000321 000321 C BOUNDARY CONDITIONS ON XI
000322 000322 C
000323 000323 C
000324 000324 DO 1199 I = 1,IBP1
000325 000325 DO 1199 J = 2,JBP1
000326 000326 XI(I,J,1) = Y(I,1)
000327 000327 1198 CONTINUE
000328 000328 1199 CONTINUE
000329 000329 C
000330 000330 C
000331 000331 C STORE NEW W INTO OLD W AND CALCULATE NEW W
000332 000332 C
000333 000333 DO 1599 I = 1,IBP1
000334 000334 DO 1599 J = 2,JBP1
000335 000335 W(I,1) = WNP1(I,1)
000336 000336 SUM1 = 0.0
000337 000337 DO 1599 K = 2,KBP1
000338 000338 SUM1 = SUM1 + (A1(I,J,K) - XI)
000339 000339 1597 CONTINUE
000340 000340 WNP1(I,1) = -2.0*RETA*DNZ*(.5*(XI(I,J,1)-XII) + SUM1)
000341 000341 IF (WNP1(I,J).LE.W0) WNP1(I,1) = 0.0

```

```

000342 000342
000343 000343
000344 000344
000345 000345
000346 000346
000347 000347
000348 000348
000349 000349
000350 000350
000351 000351
000352 000352
000353 000353
000354 000354
000355 000355
000356 000356
000357 000357
000358 000358
000359 000359
000360 000360
000361 000361
000362 000362
000363 000363
000364 000364
000365 000365
000366 000366
000367 000367
000368 000368
000369 000369
000370 000370
000371 000371
000372 000372
000373 000373
000374 000374
000375 000375
000376 000376
000377 000377
000378 000378
000379 000379
000380 000380
000381 000381
000382 000382
000383 000383
000384 000384
000385 000385
000386 000386
000387 000387
000388 000388
000389 000389
000390 000390

```

```

IF (WNP1(I,J).NE.0.0.AND.W(I,J).EQ.0.0) T(I,J)=XINP1(I,J,1)
1598 CONTINUE
1599 CONTINUE
C
WMAX = 0.0
DO 1658 J = 2,IRP1
  WTEST = WNP1(3,J)
  WMAX = AMAX1(WTEST,WMAX)
1658 CONTINUE
C
C
DO 1698 J = 2,IRP1
  W(2,J) = WNP1(2,J)
  WNP1(2,J) = WNP1(3,J)
1698 CONTINUE
C
C INITIALIZE FOR ITERATION
C
C
DO 2199 I = 2,IBAR
  DO 2198 J = 2,IRP1
    UC = .5*(U(I-1,J) + U(I,J))
    IF (UC.RE.0.0) A = W(I,J)*UC*U(I-1,J)
    IF (UC.IT.0.0) A = W(I,J)*UC*U(I,J)
    UC = .5*(U(I,J) + U(I+1,J))
    IF (UC.RE.0.0) B = W(I+1,J)*UC*U(I,J)
    IF (UC.IT.0.0) B = W(I+1,J)*UC*U(I+1,J)
    WFAC = .5*AMIN1(W(I,J-1)+W(I+1,J-1),W(I,J)+W(I+1,J))
    VC = .5*(V(I,J-1) + V(I+1,J-1))
    IF (VC.RE.0.0) C = WFAC*VC*U(I,J-1)
    IF (VC.IT.0.0) C = WFAC*VC*U(I+1,J)
    WFAC = .5*AMIN1(W(I,J)+W(I+1,J),W(I,J+1)+W(I+1,J+1))
    VC = .5*(V(I,J) + V(I+1,J))
    IF (VC.RE.0.0) D1 = WFAC*VC*U(I,J)
    IF (VC.IT.0.0) D1 = WFAC*VC*U(I+1,J)
    WFACA = .5*AMIN1(W(I,J+1)+W(I+1,J+1),W(I,J)+W(I+1,J))
    WFACR = .5*AMIN1(W(I,J)+W(I+1,J),W(I,J-1)+W(I+1,J-1))
    UNBAR(I,J) = (U(I,J)+.5*(W(I,J)+W(I+1,J)))
    + DT*(ETCFIF*(DXR*(A-B) + NYR*(C-D))
    + RNUA*NXSOR*(W(I+1,J)*(U(I+1,J)-U(I,J))
    - W(I,J)*(U(I,J)-W(I-1,J)))
    + RNUA*NYSOR*(WFACA*(U(I,J+1)-U(I,J))
    - WFACR*(U(I,J)-U(I,J-1)))
  2198 CONTINUE
2199 CONTINUE
C
C
C

```

110

```

000391 000391
000392 000392
000393 000393
000394 000394
000395 000395
000396 000396
000397 000397
000398 000398
000399 000399
000400 000400
000401 000401
000402 000402
000403 000403
000404 000404
000405 000405
000406 000406
000407 000407
000408 000408
000409 000409
000410 000410
000411 000411
000412 000412
000413 000413
000414 000414
000415 000415
000416 000416
000417 000417
000418 000418
000419 000419
000420 000420
000421 000421
000422 000422
000423 000423
000424 000424
000425 000425
000426 000426
000427 000427
000428 000428
000429 000429
000430 000430
000431 000431
000432 000432
000433 000433
000434 000434
000435 000435
000436 000436
000437 000437
000438 000438
000439 000439
000440 000440
000441 000441
000442 000442
000443 000443

```

```

DO 2299 I = 1, IBP1
  DO 2298 J = 2, JBAR
    VC = .5*(V(I,J-1) + V(I,J))
    IF (VC.GF.0.) F = W(I,J)*VC*V(I,J-1)
    IF (VC.IT.0.) F = W(I,J)*VC*V(I,J)
    VC = .5*(V(I,J) + V(I,J+1))
    IF (VC.GF.0.) F = W(I,J+1)*VC*V(I,J)
    IF (VC.IT.0.) F = W(I,J+1)*VC*V(I,J+1)
    WFAC = .5*AMIN1(W(I-1,J)+W(I+1,J+1), W(I,J)+W(I,J+1))
    UC = .5*(U(I-1,J) + U(I-1,J+1))
    IF (UC.GF.0.) G = WFAC*UC*V(I-1,J)
    IF (UC.IT.0.) G = WFAC*UC*V(I,J)
    WFAC = .5*AMIN1(W(I,J)+W(I,J+1), W(I+1,J)+W(I+1,J+1))
    UC = .5*(U(I,J) + U(I,J+1))
    IF (UC.GF.0.) H = WFAC*UC*V(I,J)
    IF (UC.IT.0.) H = WFAC*UC*V(I+1,J)
    WFACC = .5*AMIN1(W(I,J)+W(I,J+1), W(I+1,J)+W(I+1,J+1))
    WFACD = .5*AMIN1(W(I,J)+W(I,J+1), W(I-1,J)+W(I-1,J+1))

    VWBAR(I,J) = (V(I,J)+.5*(W(I,J)+W(I,J+1)))
    + DT*(ET*FIF*(NVR*(E-F) + NXR*(G-H))
    + .5*(W(I,J)+W(I,J+1))*BETA*GHAV
    + (.5*(T(I,J) + T(I,J+1)) - TIN)
    + RNUA*NXSOR*(WFACC*(V(I+1,J)-V(I,J))
    - WFACD*(V(I,J) - V(I-1,J)))
    + RNUA*NYSOR*(W(I,J+1)*(V(I,J+1) - V(I,J))
    - W(I,J)*(V(I,J) - V(I,J-1)))

  2298 CONTINUE
  2299 CONTINUE

C
C ***** BEGIN ITERATION *****
C
C CALCULATE D (MASS EQUATION)
C
  NIT = 0
  2500 CONTINUE
  DTEST = 0.0
  DSUM = 0.0
  DO 2509 I = 1, IBP1
    DO 2598 J = 2, JBP1
      WFACA = AMIN1(WNP1(I,J), WNP1(I+1,J))
      WFACR = AMIN1(WNP1(I-1,J), WNP1(I,J))
      WFACC = AMIN1(WNP1(I,J), WNP1(I,J+1))
      WFACD = AMIN1(WNP1(I,J), WNP1(I,J-1))

      D(I,J) = TWOHI*(WFACA*U(I,J)-WFACR*U(I-1,J))
      + DXR*(WFACC*V(I,J) - WFACD*V(I,J-1))*DVR
      + (WNP1(I,J) - W(I,1))*DTR

      DARS = ARS(D(I,J))
      DTEST = AMAX1(DARS, DTEST)
      DSUM = DSUM + D(I,J)
    2598
  2509

```

```

000444
000445
000446
000447
000448
000449
000450
000451
000452
000453
000454
000455
000456
000457
000458
000459
000460
000461
000462
000463
000464
000465
000466
000467
000468
000469
000470
000471
000472
000473
000474
000475
000476
000477
000478
000479
000480
000481
000482
000483
000484
000485
000486
000487
000488
000489
000490
000491
000492
000493
000494
000495
000496
000497

```

```

C
C CALCULATE GAMMA
C
      GTERM = (2.232*DT)/(T(I,J)-3.8)
      IF (WNP1(I,J).EQ.0.0) GO TO 2550
      GAM(I,J) = 0.75*(WNP1(I,J)**2) /
                  (WNP1(I,J)**3*DT*(DXSQ+DYSQ))
      1
      GO TO 2570
2550  CONTINUE
      GAM(I,J) = 0.0
2570  CONTINUE
C
C CALCULATE PHI
C
      PHI(I,J) = PHI(I,J) - (GAM(I,J)*D(T,J))
2598  CONTINUE
2599  CONTINUE
C
C BOUNDARY CONDITIONS ON PHI
C
      DO 2658 J = 2,JR1
      PHI(2,J) = -PHI(3,J)
2658  CONTINUE
C
C TEST FOR CONVERGENCE
C
      IF (NIT.EQ.0) GO TO 2658
      DEPS = DEPS0*WMAX**3
      IF (DT*ST.LT.DEPS) GO TO 3000
      IF (NIT.GE.1000) GO TO 3000
2650  CONTINUE
C
C NO CONVERGENCE
C
C
C CALCULATE U AND V
C
      DO 2699 I = 2,IBAR
      DO 2699 J = 2,JR1
      GTERMU = 2.232*DT / (.5*(T(I,J)+T(I+1,J))-3.8)
      WFAU = .5*(WNP1(I,J)+WNP1(I+1,J))
      WMINU = AMIN1(WNP1(I,J),WNP1(I+1,J))
      U(I,J) = WMINU*((WBAR(I,J)+WMINU*DT*DXR*(PHI(I,J)
                    -PHI(I+1,J))) / (WMINU*WFAU+GTERMU))
2698  CONTINUE
2699  CONTINUE
C
      DO 2799 I = 3,IBP1
      DO 2799 J = 2,JR1
      GTERMV = 2.232*DT / (.5*(T(I,J)+T(I,J+1))-3.8)
      WFAV = .5*(WNP1(I,J)+WNP1(I,J+1))
      WMINV = AMIN1(WNP1(I,J),WNP1(I,J+1))
      V(I,J) = WMINV*((WBAR(I,J)+WMINV*DT*DYR*(PHI(I,J)
                    -PHI(I,J+1))) / (WMINV*WFAV+GTERMV))

```

```

000498      000498      27981 CONTINUE
000499      000499      2799 CONTINUE
000500      000500      NIT = NIT + 1
000501      000501      GO TO 250A
000502      000502      C
000503      000503      C ***** END ITERATION *****
000504      000504      C
000505      000505      C
000506      000506      C CONVERGENCE ATTAINED. CONTINUE CYCLE
000507      000507      C
000508      000508      3000 CONTINUE
000509      000509      C
000510      000510      C CALCULATE NEW TEMPERATURES (WATER TEMPERATURE CONVECTION)
000511      000511      C
000512      000512      DO 4199 I = 2,IRP1
000513      000513      DO 4198 J = 2,IRP1
000514      000514      WFAC = AMIN1(WNP1(I,1),WNP1(I-1,J))
000515      000515      UC = U(I-1,J)
000516      000516      IF (UC.GE.0.0) AA = WFAC*UC*T(I-1,J)
000517      000517      IF (UC.LE.0.0) AA = WFAC*UC*T(I,J)
000518      000518      WFAC = AMIN1(WNP1(I,1),WNP1(I+1,J))
000519      000519      UC = U(I,J)
000520      000520      IF (UC.GE.0.0) BB = WFAC*UC*T(I,J)
000521      000521      IF (UC.LE.0.0) BB = WFAC*UC*T(I+1,J)
000522      000522      WFAC = AMIN1(WNP1(I,1),WNP1(I,J-1))
000523      000523      VC = V(I,J-1)
000524      000524      IF (VC.GE.0.0) CC = WFAC*VC*T(I,J-1)
000525      000525      IF (VC.LE.0.0) CC = WFAC*VC*T(I,J)
000526      000526      WFAC = AMIN1(WNP1(I,1),WNP1(I,J+1))
000527      000527      VC = V(I,J)
000528      000528      IF (VC.GE.0.0) DD = WFAC*VC*T(I,J)
000529      000529      IF (VC.LE.0.0) DD = WFAC*VC*T(I,J+1)
000530      000530      C
000531      000531      HC(I,J) = DT*(T*NTHI*DXR*(AA-BB) + T*OTHI*DYR*(CC-DD))
000532      000532      + W(I,1)*T(I,J) + T(I,J)*N(I,J)*NT
000533      000533      4198 CONTINUE
000534      000534      4199 CONTINUE
000535      000535      C
000536      000536      C
000537      000537      AKAPTERM = 2.0*AKAPO*NT*U7D
000538      000538      DO 4299 I = 1,IRP1
000539      000539      DO 4298 J = 2,IRP1
000540      000540      T(I,J) = 1.0 / (WNP1(I,J) + AKAPTERM)*(HC(I,J) + AKAPTERM
000541      000541      *XI(I,J,2))
000542      000542      4298 CONTINUE
000543      000543      4299 CONTINUE
000544      000544      C
000545      000545      C
000546      000546      C
000547      000547      C INCREMENT TIME AND CYCLE
000548      000548      C

```

000549	000549	NCYCLF = NCYCIF * 1
000550	000550	TI = TI + DT
000551	000551	TTI = TTI + DT
000552	000552	GO TO 900
000553	000553	
000554	000554	C *****
000555	000555	C ***** END CYCLE *****
000556	000556	C *****
000557	000557	C
000558	000558	C
000559	000559	5100 CONTINUE
000560	000560	C
000561	000561	C CALL PRINT AND PLOT ROUTINES
000562	000562	C
000563	000563	CALL PRINT
000564	000564	CALL PLOTPKG
000565	000565	C
000566	000566	C CHECK FOR DONE
000567	000567	C
000568	000568	NCY = NCY + NCYI
000569	000569	GO TO 1100
000570	000570	
000571	000571	C
000572	000572	7000 CONTINUE
000573	000573	CALL DROP
000574	000574	CALL NOTIM(ISEC)
000575	000575	CPTIME = FLOAT(ISEC) / 1.E6
000576	000576	WRITE (6,9100) CPTIME
000577	000577	CALL EXIT(1)
000578	000578	C
000579	000579	C
000580	000580	9100 FORMAT(1,14HSECONDS USED =,1PE14.5)
000581	000581	C
000582	000582	END

114

```

1*****
000001 0005A3 C
000002 0005A4 C
000003 0005A5 C
000004 0005A6
000005 0005A7 C
000006 0005A8 C
000007 0005A9
000008 0005A0
000009 0005A1
000010 0005A2
000011 0005A3
000012 0005A4
000013 0005A5
000014 0005A6
000015 0005A7 C
000016 0005A8 C
000017 0005A9
000018 0006A0
000019 0006A1
000020 0006A2
000021 0006A3
000022 0006A4
000023 0006A5
000024 0006A6 C
000025 0006A7 C
000026 0006A8 C
000027 0006A9
000028 0006B0
000029 0006B1
000030 0006B2
000031 0006B3
000032 0006B4
000033 0006B5
000034 0006B6
000035 0006B7 C
000036 0006B8 C
000037 0006B9
000038 0006C0
000039 0006C1
000040 0006C2
000041 0006C3
000042 0006C4
000043 0006C5
000044 0006C6 C
000045 0006C7 C
000046 0006C8
000047 0006C9
000048 0006D0
000049 0006D1
000050 0006D2
000051 0006D3
000052 0006D4
000053 0006D5

```

CMAI 167D 11:10:39 007/06/79
 SUBROUTINE TFRM
 PARAMETER (NT=23)
 PARAMETER (N=23)
 PARAMETER (NK=10)
 COMMON/ARRAYS/RAM(NI,NJ),PHI(NI,NJ),T(NI,NJ),
 1 UWBAR(NI,NJ),VWRAR(NI,NJ),II(NI,NJ),
 2 V(NI,NJ),W(NI,NJ),WNP1(NI,NJ),XI(NI,NJ,NK),
 3 XINP1(NI,NJ,NK),D(NI,NJ),TNP1(NI,NJ),TRAM(NJ)
 C
 C
 COMMON/VAR/AKAP0,AKAP0,HETA,HETAR,DT,DTR,DX,DY,DZ,DXR,DYR,DZR,
 1 DXSQ,DYSQ,DZSQ,DYSOR,DYSOR,DZSOR,EIGFIF,EPSILON,GRAV,
 2 IRAR,IBP1,IRP2,IRAR,JRP1,JRP2,HEIGHT,
 3 KRAR,KBP1,KRM1,NCYCLE,NIT,RNUA,TI,TTI,NCYF,NCYP,
 4 NCYT,TWICK,TIN,TWOTHY,VIN,VOUT,W0,DC,XII,
 5 RR,RW,RHOR,RHUW,AKR,DSUM,WMAX,NCYTER,NCYTERI
 C
 NCYTER = NCYTER + NCYTERI
 C
 C CALCULATE ENERGY DIFFERENCE
 C
 EDOTDIF = 0.
 DO 4598 J = 2,IBP1
 UC = U(2,J)
 IF (UC.NT.0) TC = T(2,1)
 IF (UC.EF.0) TC = T(3,1)
 EDOTDIF = EDOTDIF + UC*TC*RHOR*BW*DY*WNP1(2,J)
 4598 CONTINUE
 C
 C CALCULATE VOLUME OF SECONDARY CRACK
 C
 VSUM = 0.
 DO 4699 I = 1,IBP1
 DO 4699 J = 2,JRP1
 VSUM = VSUM + WNP1(I,J)
 4698 CONTINUE
 4699 CONTINUE
 VOL = VSUM*DX*DY
 C
 C
 C PRINT OUT CURRENT TIME, ENERGY DIFFERENCE, ITERATION NUMBER
 C AND CYCLE NUMBER ON TELETYPE WHILE RUNNING
 C
 PRINT 9110,TI,TTI,EDOTDIF,VOL,NIT,NCYCLE
 RETURN
 9110 FORMAT(3HTI=F10.3,IX,4HTTI=F10.3,IX,
 1 2HW=F9.2,IX,2HW=F9.2,IX,2PHI=F14.1,IX,2HC=F14)
 END

[illegible]

000049	0006A4
000050	0006A5
000051	0006A6
000052	0006A7
000053	0006A8
000054	0006A9
000055	0006AQ
000056	0006Q1

C

```

WRITE(6,9240)
WRITE(6,9235)
WRITE(6,9240)
DO 5399 I = 1,IBP2
  DO 5398 J = 1,JRP2
    WRITE(6,9045) I,I,U(I,J),V(I,J),PHI(I,J),T(I,J),WNP1(I,J),

```

000057	000602
000058	000603
000059	000604
000060	000605
000061	000606
000062	000607
000063	000608
000064	000609
000065	000700
000066	000701
000067	000702
000068	000703
000069	000704
000070	000705
000071	000706
000072	000707
000073	000708
000074	000709
000075	000710
000076	000711
000077	000712
000078	000713
000079	000714
000080	000715
000081	000716
000082	000717
000083	000718
000084	000719
000085	000720
000086	000721
000087	000722
000088	000723
000089	000724
000090	000725
000091	000726
000092	000727
000093	000728
000094	000729
000095	000730
000096	000731

```

1          GAM(1,J),D(1,J),X(1,J),Z(1,J),Y(1,J),S(1,J),A(1,J),4)
5398      CONTINUE
5399      WRITE(6,9040)
CONTINUE
C
RETURN
C
C
C
C
C
FORMATS
C
0000 FORMAT(1H'I',5X,'HTIME= ',1PF12.5,5X,'HTOTAL TIME= ',1PE12.5,
1        5X,'HNCYCLES= ',1F10X,'HUNIT= ',16)
9005 FORMAT(7H'ENDOTIF',E9.2,
1        5X,'HNDQUM=',E9.2)
9010 FORMAT(3HRW=E9.2,5X,'HRW'=F9.2,5X,'SHRMOW=E9.2,3X,'SHRHOP=
1        E9.2,3X,'HAKR=F9.2)
9012 FORMAT(6H'HAKAPR=E9.2,2X,'HAKAPR=F9.2,2X,'SHBFTA=E9.2,3X,
1        6HBFTAR=E9.2,4X,'3HNT=F9.2,5X,'4HUTR=F9.2)
9015 FORMAT(3HDY=F9.2,5X,'3HDY=F9.2,5X,'3MDZ=E9.2,5X,'4HDXR=F9.2,
1        4X,'4HNYR=E9.2,4X,'4HNR=F9.2)
9017 FORMAT(5H'DXSQR=E9.2,3X,'5H'DYSQR=F9.2,3X,'6H'DXSQR=E9.2,2X,
1        6H'DYSQR=E9.2,2X,'6H'DZSQR=F9.2)
9020 FORMAT(7H'TIGTFI=E9.2,4X,'8HFPSION=E9.2,1X,'5HGRAV=E9.2,
1        5X,'5HIRAR=I4,8X,'5MJRAR=I4,2X,'5HKRAR=I4)
9022 FORMAT(5H'DNUA=E9.2,3X,'3HTI=F9.2,5X,'5HNCYF=I4,5X,'5HNCYP=
1        I4,8X,'5HNCYI=I4)
9025 FORMAT(6H'THIK=E9.2,2X,'4HTIN=F9.2,4X,'7HTWOTH=E9.2,1X,
1        4HVIN=F9.2,4X,'7HWEIGHT=F9.2)
9027 FORMAT(3HW0=F9.2,5X,'3HDC=F9.2,5X,'4HXII=E9.2,4X,'3HNI=I6,
1        8X,'3HNJ=I6,8X,'3WNK=I6)
9035 FORMAT(/,8HCPLI I J,R,Y,1MU,10X,1MV,9X,'3HPHI,9X,1MT,8X,'4HWP,1,
1        /X,'3HGAM,9X,1MD,6X,'9MXI(I,J,2),2X,'9MXI(I,J,3),2X,
2        9MXT(I,J,4))
9040 FORMAT(/)
9042 FORMAT(/)
9045 FORMAT(2I4,2X,I0(1PE10.3,1X))
C
END

```

```

1*****
000001 000732
000002 000733
000003 000734
000004 000735
000005 000736
000006 000737
000007 000738
000008 000739
000009 000740
000010 000741
000011 000742
000012 000743
000013 000744
000014 000745
000015 000746
000016 000747
000017 000748
000018 000749
000019 000750
000020 000751
000021 000752

```

C

C

C

CHAT 167D 11:10:39 R07/06/79

SUBROUTINE UNDRP

```

LCM (RINJIN), (FLMBHFC)
COMMON /RINJIN/ IBSIZE, IHUFF (5000)
COMMON /FLMBHFC/ FLMBHIF (5000)
DIMENSION IHEAD (6)
IBSIZE = 5000
CALL FILMBO
CALL IDENTBO (12, FLMBHIF, 5000, 1100000, 105.0,
IHEAD, 1) * 100 T3 PUTW
IHEAD (2) = 100 HGE0 235
IHEAD (3) = 100 HMX = 25.
IHEAD (4) = 100 HNC = 50.
IHEAD (5) = 100 HTIN = 152.
IHEAD (6) = 100 HVIN = 0.0
CALL HEADBO (12, 60, IHEAD, 100 HMX T3 PUTW)
CALL KEEPFLM (12)
RETURN
END

```

```

1*****
000001 000753
000002 000754
000003 000755
000004 000756
000005 000757
000006 000758
000007 000759
000008 000760
000009 000761
000010 000762
000011 000763

```

C

C

C

CHAT 167D 11:10:39 R07/06/79

SUBROUTINE DRAP

```

LCM (FLMBHFC)
COMMON /FLMBHFC/ FLMBHIF (5000)
CALL FILMBO
CALL IDENTBO (12, FLMBHIF, 5000, 1100000, 105.0,
CALL KEEPFLM (12)
RETURN
END

```

```

1*****
000001 0007A4 C CHAT 167D 11110:30 R07/06/79
000002 0007A5 C
000003 0007A6 C SURROUING PLOTPKG
000004 0007A7 C
000005 0007A8 C
000006 0007A9 PARAMETER (N1=23)
000007 000770 PARAMETER (N1=23)
000008 000771 PARAMETER (NK=10)
000009 000772 COMMON/ARRAYS/RAM(N1,NJ),PHI(N1,NJ),T(N1,NJ),
000010 000773 1 UWBAR(N1,NJ),VWRAP(N1,NJ),U(N1,NJ),
000011 000774 2 V(N1,NJ),W(N1,NJ),WNP1(N1,NJ),XI(N1,NJ,NK),
000012 000775 3 XINP1(N1,NJ,NK),D(N1,NJ),TNP1(N1,NJ),TRAR(NJ)
000013 000776 C
000014 000777 C
000015 000778 COMMON/VAR/AKAP0,AKAP0,BETA,BETAR,DT,DTR,Dx,Dy,Dz,DxR,DyR,DzR,
000016 000779 1 DXSQ,DYSQ,DZSQ,DYSQR,DYSQR,DZSQ,EIGFIF,EPSILON,GRAV,
000017 0007A0 2 IRAR,IRP1,IRP2,IRAR,JRP1,JRP2,HEIGHT,
000018 0007A1 3 KAR,KBP,KRM,NCYCLF,NIT,RNU,TI,TTI,NCYP,NCYP,
000019 0007A2 4 NCYI,THICK,TIN,TWOCHI,VIN,VOUT,W0,DC,XII,
000020 0007A3 5 BR,RW,RHOR,RHOW,AKR,DSUM,WMAX,NCYTER,NCYTERI
000021 0007A4 C
000022 0007A5 C
000023 0007A6 COMMON/CJF57/IXL,IXR,IYT,IYB,XMN,XMX,YMY,YMN
000024 0007A7 DIMENSION CON(25),BCD(2),TIME(6),VAVE(2),X(1),Y(1)
000025 0007A8 EQUIVALENCE (UWBAR(1,1),X(1)),(VWRAP(1,1),Y(1))
000026 0007A9 C
000027 0007A0 C
000028 0007A1 C CONTOUR PLOTS
000029 0007A2 C
000030 0007A3 IPLT=1
000031 0007A4 IFINISH=1
000032 0007A5 C
000033 0007A6 C SET COORDINATES FOR CELL CENTERED QUANTITIES
000034 0007A7 C (THE BOUNDARIES ARE UNCHANGED.)
000035 0007A8 C
000036 0007A9 95 CONTINUE
000037 000800 X(1)=0.
000038 000801 Y(1)=0.
000039 000802 X(2)=.5*DX
000040 000803 Y(2)=.5*DY
000041 000804 DO 100 I=2,IRAR
000042 000805 X(I+1)=X(I)+DX
000043 000806 100 CONTINUE
000044 000807 C
000045 000808 DO 120 J=2,JRAR
000046 000809 Y(J+1)=Y(J)+DY
000047 000810 120 CONTINUE
000048 000811 X(IRP2)=X(IRP1)+.5*DX
000049 000812 Y(IRP2)=Y(IRP1)+.5*DY
000050 000813 GO TO (125,155,195,245),IPLT

```

```

000051 000814
000052 000815
000053 000816
000054 000817
000055 000818
000056 000819
000057 000820
000058 000821
000059 000822
000060 000823
000061 000824
000062 000825
000063 000826
000064 000827
000065 000828
000066 000829
000067 000830
000068 000831
000069 000832
000070 000833
000071 000834
000072 000835
000073 000836
000074 000837
000075 000838
000076 000839
000077 000840
000078 000841
000079 000842
000080 000843
000081 000844
000082 000845
000083 000846
000084 000847
000085 000848
000086 000849
000087 000850
000088 000851
000089 000852
000090 000853
000091 000854
000092 000855
000093 000856
000094 000857
000095 000858
000096 000859
000097 000860
000098 000861
000099 000862

C
125 CONTINUE
C
C LABELS AND NUMBER OF CHARACTERS FOR AXES
BCD(1) = AHX_AXIS
BCD(2) = AHY_AXIS
NAN = 6
NYN = 6
C
C NO. OF CHARACTERS FOR FRAME TITLE
MZL = 40
C
C PLOT W
C
C WRITE LARGE CHARACTERS FOR TIME (REPLACES ENCODE)
C
WRITE(63.000) TI,TTI
REWIND 63
READ(63.000) (TIME(I),I=1,4)
901 FORMAT(4A10)
900 FORMAT(1X,=CRACK WIDTH AT T=,=F10.3,=,=E10.3)
C
C STORE CRACK WIDTH IN DUMMY ARRAY WC SUCH THAT THE PRIMARY CRACK
C IS EXCLUDED. (IT IS THE LAST COLUMN IN THE PRIMARY CRACK.)
JM = 2*JPAR
DO 150 I = 2,JRP1
DO 150 J = 2,JRP1
WC(I,J) = WNP1(I,J)
IF(I.LF.2) WC(I,J)=WMAX
150 CONTINUE
C
C SET UPPER AND LOWER BOUNDARIES IN THE PRIMARY CRACK TO
C ADJACENT INNER VALUES
C
C SET UPPER AND LOWER BOUNDARIES TO ZERO OUTSIDE PRIMARY CRACK
DO 153 I=1,JRP2
IF(I.GT.2) GO TO 152
WC(I,1)=WMAX
WC(I,JRP2)=WMAX
GO TO 153
152 CONTINUE
WC(I,1)=0
WC(I,JRP2)=0
153 CONTINUE
C
C SET LEFT BOUNDARY WIDTH TO ADJACENT INNER
C SET RIGHT BOUNDARY WIDTH TO 0.
DO 154 J=1,JRP2
WC(1,J)=WMAX
WC(JRP3,J)=0

```

```

000100 0008A3 154 CONTINUE
000101 0008A4 CALL CONTR(JR,X,-IBP2,Y,-JRP2,HC(1,1),NI,NJ,17,0,0,0,-1,0,
000102 0008A5 1 CON,DX,DY,T,TIME,MZL,6HX=AXIS,6,6HY=AXIS,6)
000103 0008A6 IPLT=IPLT.T
000104 0008A7 GO TO 325
000105 0008A8 C
000106 0008A9 C PLOT T
000107 0008A9 C
000108 0008A7 155 CONTINUE
000109 0008A7 REWIND 63
000110 0008A7 WRITE(63,910) T1,TT1
000111 0008A7 REWIND 63
000112 0008A7 READ(63,911) (TIME(I),I=1,4)
000113 0008A7 910 FORMAT(1X,=WATER TEMP AT T= E.E10.3,E.E10.3)

000115 0008A7 L
000116 0008A7 C STORE WATER TEMPERATURE IN DUMMY ARRAY
000117 0008A7 DO 180 I=2,IAP1
000118 0008A7 DO 180 J=2,JRP1
000119 0008A7 HC(I,J)=T(I,J)
000120 0008A7 180 CONTINUE
000121 0008A7 C
000122 0008A7 C SET UPPER AND LOWER BOUNDARY TEMPERATURES OF PRIMARY CRACKS TO
000123 0008A7 C ADJACENT INNER
000124 0008A7 C SET UPPER AND LOWER BOUNDARY TEMPERATURES OUTSIDE PRIMARY CRACK TO
000125 0008A7 C TEMP AT INFINITY
000126 0008A7 HC(1,2)=HC(2,2)
000127 0008A7 HC(IAP2,2)=XII
000128 0008A7 HC(1,IAP1)=HC(2,JRP1)
000129 0008A7 HC(IAP2,JRP1)=XII
000130 0008A7 DO 183 I=1,IAP2
000131 0008A7 IF(I.GT.2) GO TO 182
000132 0008A7 HC(I,1)=HC(I,2)
000133 0008A7 HC(I,JRP2)=HC(I,JRP1)
000134 0008A7 GO TO 183
000135 0008A7 182 CONTINUE
000136 0008A7 HC(I,1)=XII
000137 0008A7 HC(I,JRP2)=XII
000138 0008A7 183 CONTINUE
000139 0008A7 C
000140 0008A7 C SET LEFT BOUNDARY TEMPERATURES TO ADJACENT INNER
000141 0008A7 C SET RIGHT BOUNDARY TEMPERATURES TO TEMP AT INFINITY
000142 0008A7 DO 184 J=2,JRP2
000143 0008A7 HC(I,J)=HC(2,J)
000144 0008A7 HC(IAP2,J)=XII
000145 0008A7 184 CONTINUE
000146 0008A7 C
000147 0008A7 C SET SPECIAL CONTRA MATRIX
000148 0008A7 NC = -10
000149 0008A7 NOC = -NC
000150 0008A7 CALL MINM(HC,NI,IBP2,JRP2,I,J,ZMIN)
000151 0008A7 CALL MAXM(HC,NI,IBP2,JRP2,I,J,ZMAX)
000151 0008A7 DELZ = (ZMAX - ZMIN) / (NOC-1.)

```

000152	000915	C
000153	000916	C REMOVE BOUNDARY VALUES
000154	000917	7MIN = ZMIN + DELZ
000155	000918	ZMAX = ZMAX + DELZ
000156	000919	DELZ = (ZMAX-7MIN) / (NOC-1.)
000157	000920	CON(1) = 7MIN
000158	000921	DO 189 I = 2,NOC
000159	000922	CON(I) = CON(I-1) + DELZ
000160	000923	189 CONTINUE
000161	000924	C
000162	000925	192 CONTINUE
000163	000926	CALL CONTRJR(X,IRP2,Y,IRP2,HC(1,1),NI,NJ,NC,0,0,DELZ,
000164	000927	CON,DX,DY,T,TIME,MZL,AXIS=AXIS,6,AY=AXIS,6)
000165	000928	1 IPLT=IRPLT+1
000166	000929	GO TO 325
000167	000930	C
000168	000931	C PLOT PHI
000169	000932	C
000170	000933	195 CONTINUE
000171	000934	HLWIND 63
000172	000935	WRITE(63,020) TI,TTI
000173	000936	REWIND 63
000174	000937	READ(63,021) (TIME(I),I=1,4)
000175	000938	920 FORMAT(EMOND PRESSURE AT T= ,F10.3,E.3,E10.3)
000176	000939	C
000177	000940	C
000178	000941	C USE HC ARRAY TO PLOT PHIS. THIS ELIMINATES PLOTTING
000179	000942	C ZERO PRESSURES
000180	000943	C
000181	000944	DO 200 I = 2,IRP1
000182	000945	DO 200 J = 2,IRP1
000183	000946	HC(I,J) = PHI(I,J)
000184	000947	IF (WNP1(I,J).LT.Wy) HC(I,J)=HC(I-1,J)
000185	000948	200 CONTINUE
000186	000949	C
000187	000950	C SET UPPER AND LOWER BOUNDARY PRESSURES TO ADJACENT INNER
000188	000951	HC(1,2) = HC(2,2)
000189	000952	HC(IRP2,2) = HC(IRP1,2)
000190	000953	HC(1,IRP1) = HC(2,IRP1)
000191	000954	HC(IRP2,IRP1) = HC(IRP1,IRP1)
000192	000955	DO 202 I = 1,IRP2
000193	000956	HC(I,1) = HC(I,2)
000194	000957	HC(I,IRP2) = HC(I,IRP1)
000195	000958	202 CONTINUE
000196	000959	C
000197	000960	C SET LEFT AND RIGHT BOUNDARY PRESSURES TO ADJACENT INNER
000198	000961	DO 203 J = 1,IRP2
000199	000962	HC(1,J) = HC(2,J)
000200	000963	HC(IRP2,J) = HC(IRP1,J)
000201	000964	203 CONTINUE

```

000202 0009A5 CALL CONTRJ(X,-IBP2*Y,-JRP2*HC(1.1),NI,NJ,10.0,0.0,-1.0,
000203 0009A6 1 CON,DX,DY,1,TIME,MZL,6HX-AXIS,6,6HY-AXIS,6)
000204 0009A7 IPLT=IPLT+1
000205 0009A8 GO TO 325
000206 0009A9 C
000207 000970 C
000208 000971 C
000209 000972 C
000210 000973 C VECTOR PLOTS
000211 000974 C
000212 000975 215 CONTINUE
000213 000976 IFINISH=2
000214 000977 C
000215 000978 C SET VELOCITY IN PRIMARY CRACK TO ZERO FOR PLOT PURPOSES
000216 000979 C
000217 0009A0 DO 25A J = 1,JRP1
000218 0009A1 V(P,J) = 1.F-10
000219 0009A2 25B CONTINUE
000220 0009A3 C
000221 0009A4 C NORMALIZATION
000222 0009A5 C
000223 0009A6 VELF = 0.0
000224 0009A7 DO 300 I = 2,IJRP1
000225 0009A8 DO 301 J = 2,JRP1
000226 0009A9 STEET = SQRT((U(I+1,J)+U(I,J))**2
000227 0009A0 1 + (.5*(V(I,J+1)+V(I,J))**2)
000228 0009A1 VELF = AMAX1(STEET,VELF)
000229 0009A2 301 CONTINUE
000230 0009A3 300 CONTINUE
000231 0009A4 C
000232 0009A5 C VELF IS MAXIMUM VELOCITY COMPONENT IN SYSTEM
000233 0009A6 C PICK OUT THE MAXIMUM VELOCITY COMPONENT IN SYSTEM
000234 0009A7 C
000235 0009A8 DF = DX
000236 0009A9 IF (DX GT DY) DF = DY
000237 0010A0 VNORM = .50
000238 0010A1 VELF = VNORM*DF/VELF
000239 0010A2 C
000240 0010A3 C SET UP FRAME AND LABEL
000241 0010A4 C
000242 0010A5 C
000243 0010A6 C WRITE LARGE CHARACTERS FOR TIME
000244 0010A7 C
000245 0010A8 REWIND 63
000246 0010A9 WRITE(63,930) TI,TTI
000247 001010 REWIND 63
000248 001011 READ(63,921) (TIME(I),I=1,4)
000249 001012 930 FORMAT(1X,=VELOCITY AT T=,E10.3,E3.3,E12.3)
000250 001013 C
000251 001014 CALL ADV(TI)
000252 001015 CALL FRAME(IXL,IXR,IYT,IYR,XMN,XXM,XXN,YYM,YYN)
000253 001016 CALL DGA(IXL,IXR,IYT,IYR,XMN,XXM,XXN,YYM,YYN)

```



```

000254 001017 C
000255 001018 C CENTER THE TITLE AND THE LABELS FOR THE VECTOR PLOTS
000256 001019 C IX=MAX0(4R,IXL,(IXR-IXI-12*MZL)/2)
000257 001020 IX=IXI
000258 001021 LZL=MAX0(M7I,(1024-IXI)/12,M)+1
000259 001022 IF (L7I,GT,1) L7I=(LZL+83)/85+1
000260 001023 CALL NLCH/IX,MAX0(10,IYT-24,14*(L7I)*MZL,TIME(1),2)
000261 001024 CALL NLCH/MAX0(54,IXL,(IXR-IXL-12*NXN)/2),IYR+26,NXN,BCD(1),2)
000262 001025 CALL NLCH/78,MIN0(IYB+52,IYR-(IYR-IYT-12*NYN)/2),NYN,BCD(2),2)
000263 001026 C
000264 001027 C CALCULATE VELOCITY VECTORS
000265 001028 C
000266 001029 DO 320 J = 2,JRP1
000267 001030 Y1 = Y(I,J)
000268 001031 DO 320 I = 2,IRP1
000269 001032 X1 = X(I,J)
000270 001033 X2 = X1+VFLF*(U(I-1,J)+U(I,J))
000271 001034 Y2 = Y1+VFLF*(V(I,J-1)+V(I,J))
000272 001035 CALL CONVRT(X1,IX1,XMN,XMX,IXL,IXR)
000273 001036 CALL CONVRT(Y1,IY1,VMN,VMX,IYR,IYT)
000274 001037 CALL CONVRT(X2,IX2,XMN,XMX,IXL,IXR)
000275 001038 CALL CONVRT(Y2,IY2,VMN,VMX,IYR,IYT)
000276 001039 CALL DRV(IX1,IY1,IX2,IY2)
000277 001040 320 CONTINUE
000278 001041 C
000279 001042 C DETERMINE MAX SPEED AND ZONE
000280 001043 VMX = 0.
000281 001044 IMX = 1
000282 001045 JMX = 1
000283 001046 DO 342 I = 2,IRP1
000284 001047 DO 342 J = 2,JRP1
000285 001048 VAVE(1) = .5*(U(I-1,J) + U(I,J))
000286 001049 VAVE(2) = .5*(V(I,J-1) + V(I,J))
000287 001050 VAVE(1) = VAVE(1)+VAVE(1)
000288 001051 VAVE(2) = VAVE(2)+VAVE(2)
000289 001052 VTEST = SQRT(VAVE(1)+VAVE(2))
000290 001053 IF (VMX,GT,VTEST) GO TO 342
000291 001054 VMX = VTEST
000292 001055 IMX = I
000293 001056 JMX = J
000294 001057 342 CONTINUE
000295 001058 C
000296 001059 C RESET VELOCITY IN PRIMARY CRACK TO INLET VELOCITY
000297 001060 C
000298 001061 DO 398 J = 1,JRP1
000299 001062 V(2,J) = VIN
000300 001063 398 CONTINUE
000301 001064 C
000302 001065 C IDENTIFY MAX SPEED AND ZONE
000303 001066 REWIND 63
000304 001067 WRITE(63,940) VMX,IMX,JMX

```

```

000305 001068 REWIND 63
000306 001069 READ(43,9A1) (TIME(I),I=1,4)
000307 001070 CALL NLCH(116,1000,60,TIME(1),2)
000308 001071 940 FORMAT(1X,10HMAXIMUM SPEED,7,J1,1PE12.5,214,
000309 001072 941 FORMAT(1X,6A1A)
000310 001073 C
000311 001074 C DRAW DOTTED LINE BETWEEN PRIMARY FRACTURE AND
000312 001075 C THERMAL STRESS FRACTURE
000313 001076 C
000314 001077 C SET UP X COORDINATES
000315 001078 325 CONTINUE
000316 001079 XF = X(2) + .56DX
000317 001080 CALL CONVRT(XF,IX1,XMN,XX,XL,IXR)
000318 001081 IX2 = IX1
000319 001082 C
000320 001083 C SET UP Y COORDINATES
000321 001084 JDOT = 5
000322 001085 DYDOT = DY/JDOT
000323 001086 YL = NYDOT + .5
000324 001087 JDOT = JARS/JDOT + 1
000325 001088 Y(1) = 0.
000326 001089 DO 330 J = 1, JDOT
000327 001090 Y(J+1) = Y(J) + DYDOT
000328 001091 330 CONTINUE
000329 001092 C
000330 001093 C PLOT DOTS
000331 001094 DO 340 J = 1, JDOT
000332 001095 Y1 = Y(J)
000333 001096 Y2 = Y(J) + VI
000334 001097 CALL CONVRT(Y1,IY1,YMN,YMX,IYR,IYT)
000335 001098 CALL CONVRT(Y2,IY2,YMN,YMX,IYR,IYT)
000336 001099 CALL DRV(IX1,IY1,IX2,IY2)
000337 001100 340 CONTINUE
000338 001101 GO TO (95,745),IFINISH
000339 001102 C
000340 001103 345 RETURN
000341 001104 END

1*****
000001 001105 C
000002 001106 C
000003 001107 SURROUTINE PLNJB(X,Y,NPTS,INC,LNN,NSYM,C,XAA,YAA,LABELZ,NZL,LABELX
000004 001108 1,NXL,LAEL(Y,NYL))
000005 001109 C
000006 001110 C LABEL AXES, WRITE TITLES, DRAW BORDERS AND CONNECT CONTOUR POINTS
000007 001111 C ESTABLISHED IN TRICJR
000008 001112 COMMON /C,IF07,IXL,IXR,IYT,IYR,XMN,XX,YMX,YMN
000009 001113 DIMENSION X(7),Y(1)
000010 001114 INTEGER GRID
000011 001115 R=AMAX1(AMAX1(C,0.)*(LNN+1),0.)
000012 001116 LIN=LNN
000013 001117 KSYM=IARS(NSYM)
000014 001118 KINC=MAX0(IARS(INC),1)

```

```

000015 001119
000016 001120
000017 001121
000018 001122
000019 001123
000020 001124
000021 001125
000022 001126
000023 001127
000024 001128
000025 001129
000026 001130
000027 001131
000028 001132
000029 001133
000030 001134
000031 001135
000032 001136
000033 001137
000034 001138
000035 001139
000036 001140
000037 001141
000038 001142
000039 001143
000040 001144
000041 001145
000042 001146
000043 001147
000044 001148
000045 001149
000046 001150
000047 001151
000048 001152
000049 001153
000050 001154
000051 001155
000052 001156
000053 001157
000054 001158
000055 001159
000056 001160
000058 001162
000059 001163
000060 001164
000061 001165
000062 001166
000063 001167
000064 001168
000065 001169
000066 001170
000067 001171
000068 001172

```

C ESTABLISH NO. CHARACTERS AND NO. POINTS

```

MPTS=IARS(NPTS)
MZL=IARS(NZL)
XXA=ARS(XAA)
YYA=ARS(YAA)
NAN=IARS(NXL)
NYN=IARS(NYL)
GRIDF=AMAX1(1.,ABS(C))
IF (NBYM.I.T.N) GO TO 130

```

C ADVANCE FRAME AND DETERMINE MAX AND MIN FOR AXES

```

CALL ADV (1)
CALL MAXV (X,KINC,MPTS,ISUR,XXM)
CALL MAXV (Y,KINC,MPTS,ISUR,XXM)
CALL MINV (X,KINC,MPTS,ISUR,XXM)
CALL MINV (Y,KINC,MPTS,ISUR,XXM)

```

C IF XXA AND/OR YYA ARE NON-ZERO THE LENGTHS
C WILL BE CONSIDERED AS RATIOS WHERE THE LONGEST
C SIDE IS FITTED ON A 90° POINT LINE.

C DETERMINE ASPECT RATIO OF BORDERS

```

IF (XXA.EQ.0) XXA=6.
IF (YYA.EQ.0) YYA=10.
DXM=.001*ARS(XXM)
IF (DXM.EQ.0) DXM=.001
XXM=XXM-DXM

```

```

60 IF ((SIGN(1,NYL).LT.0.AND.T.C.GT.0) .OR. YYA=XXM-XXM) YYA
IF ((SIGN(1,NXL).LT.0.AND.NPTS.GT.0) .OR. XXA=XXM-XXM) XXA

```

C DETERMINE 4020 COORDINATES OF BORDER

```

IXL=120
IYB=970
FACT=ABS(1/AMAX1(XXA,YYA))
IXR=120+FIX(FACT*XXA)
IYT=970-FIX(FACT*YYA)
CALL FRAME (IXL,IXR,IYT,IYB)
IF (SIGN(1,XAA).GT.0) GO TO 70
SWAP=XXM
XXM=XXM
XXM=SWAP
SWAP=YYM
YYM=YYM
YYM=SWAP
80 CALL NGA (IXI,IXR,IYT,IYB,XXM,XXM,YYM,YYM)

```

C CENTER TITLE AND LABEL AXES

```

IF (NXX.NF.0) CALL DLCH (MAX0(54,IXL+(IXR-IXL-12*NXX)/2),IYB+26
1,NXX,AREIX,2)
INCX=78
IF (NXX.NF.0) INCX=18
IF (NXX.NF.0) CALL DLCH (INCX,MIN0(IYB+50,IYB-(IYB-IYT-12*NYN)/2)

```

```

000069 001173 1,NYN,(ARELY,2)
000070 001174 C IX=MAX0(4A,IXL,(IXR-IXI-12*LZL)/2)
000071 001175 IX=IXI
000072 001176 LZL=MAX0(M7L-(1024-IXI)/12,A)+1
000073 001177 IF (L7L.GT.1) LZL=(LZL+R3)/R5+1
000074 001178 IF (N7L.NE.0) CALL DLCH (IX,MAX0(10*IYT-24-16*LZL),M7L,LABEL7,2)

000075 001179 C
000076 001180 C INCREASE EXPOSURE
000077 001181 CALL FXH
000078 001182 IF (N7L.LT.0) GO TO 220
000079 001183 C
000080 001184 C PLOT POINTS AND/OR LINE
000081 001185 130 MPTS=MPTS+KINC
000082 001186 DO 210,NXP=1,MPTS,KINC
000083 001187 XTWO=X(NXP)
000084 001188 YTWO=Y(NXP)
000085 001189 IF (MPTS.I.T.A) XTWO=A/OG10(XTWO)
000086 001190 IF (INC.LT.0) YTWO=A/LOG10(YTWO)
000087 001191 CALL CONVBT (XTWO,NXTWO,XMN,XXM,IXL,IXR)
000088 001192 CALL CONVBT (YTWO,NYTWO,YMN,YYM,IYB,IYT)
000089 001193 IF (NXP.EQ.1) GO TO 190
000090 001194 IF (LIN.GE.0) GO TO 180
000091 001195 140 IF (MOD((NXP-1)/KINC,ABS(LIN)),NE.0) GO TO 150
000092 001196 C
000093 001197 C LIGHTEN EXPOSURE
000094 001198 CALL FXL
000095 001199 C
000096 001200 C REMOVES CONTOUR LABELS
000097 001201 C CALL DLCH (NXTWO,NYTWO,0,KSYM,1)
000098 001202 CALL FXH
000099 001203 GO TO 200
000100 001204 150 IF (R.FO.A) GO TO 200
000101 001205 160 DO 170,IB=1,A
000102 001206 170 CALL PLT (NXTWO,NYTWO,42)
000103 001207 GO TO 200
000104 001208 180 IF (R.EQ.A) CALL DRV (NXONE,NYONE,NXTWO,NYTWO)
000105 001209 190 IF (LIN.NE.0) GO TO 140
000106 001210 IF (R.NE.A) GO TO 160
000107 001211 200 NYONE=NYTWO
000108 001212 NXONE=NXTWO
000109 001213 210 CONTINUE
000110 001214 220 RETURN
000111 001215 END

1*****
000001 001216 C
000002 001217 C
000003 001218 SURROUTINE CONTRJB(X,NMX,Y,NMY,7,NZX,NZY,NC,ZMN,ZMX,DLZ,7PLAN,DMPX
000004 001219 1,DMPY,IGRO,ITITLE,NTITLE,XLABEL,NXLRL,YLABEL,NYLRL)
000005 001220 C
000006 001221 C

```

CHAT 1670 11110139 M07/06/79

```

000007 001222 C ESTABLISH THE CONTOUR SCHEME TO BE USED BY TRICOR
000008 001223 COMMON /CNTRCCM/ ISYM(50)
000009 001224 DIMENSION XSCALE(2), YSCALE(2)
000010 001225 EQUIVALENCE (XMIN,XSCALE(1)), (XMAX,XSCALE(2))
000011 001226 EQUIVALENCE (YMIN,YSCALE(1)), (YMAX,YSCALE(2))
000012 001227 DIMENSION X(1), Y(1), Z(NZX,1), ZPLAN(1)
000013 001228 DIMENSION ZT(4)
000014 001229 DIMENSION CON(6)
000015 001230 NOCMIN=(IARS(1NC)+50)
000016 001231 ZMIN=ZMN
000017 001232 ZMAX=ZMX
000018 001233 DELZ=DL7
000019 001234 DMAPX=DMPX
000020 001235 DMAPY=DMPY
000021 001236 NOX=IARS(1NX)
000022 001237 NOY=IARS(1NY)
000023 001238 DO 10 I=1,50
000024 001239 10 ISYM(I)=0
000025 001240 C
000026 001241 C ESTABLISH SCALES
000027 001242 CALL MINV (X,I,NOX,I,XMIN)
000028 001243 CALL MAXV (Y,I,NOY,I,YMAX)
000029 001244 CALL MAXV (X,I,NOX,I,XMAX)
000030 001245 CALL MINV (Y,I,NOY,I,YMIN)
000031 001246 FGRD=0.
000032 001247 IF (IARR GT 0) FGRD=IARR
000033 001248 CALL PLOJR (XSCALE,YSCALE,2,1,1,1,FGRD,DMAPX,DMAPY,ITITLE,-NTITLE
000034 001249 1,XLAFR,NXLBI,YLABEL,NYLBL)
000035 001250 IF (NP.LT.0) GO TO 50
000036 001251 C
000037 001252 C DETERMINE MAX-MIN CONTOURS AND SPACING
000038 001253 C STORE IN ARRAY ZPLAN
000039 001254 IF (1NX.LT.0) CALL MINM (Z,NZX,NOX,NOY,I,J,ZMIN)
000040 001255 IF (1NY.LT.0) CALL MAXM (Z,NZX,NOX,NOY,I,J,ZMAX)
000041 001256 IF (DFLZ.GT.0) GO TO 20
000042 001257 DELZ=(ZMAX-ZMIN)/(NOC-1.)
000043 001258 20 IF (1NY.GT.0) GO TO 30
000044 001259 ZMAX=ZMAX-AMOD(ZMAX,DELZ)
000045 001260 ZMIN=ZMIN-AMOD(ZMIN,DELZ)
000046 001261 NOCMIN=(NOC-1)*FIX((ZMAX-ZMIN)/DELZ+1.01)
000047 001262 30 ZPLAN(1)=ZMIN
000048 001263 DO 40 I=2,NOC
000049 001264 40 ZPLAN(I)=ZPLAN(I-1)+DELZ
000050 001265 C
000051 001266 C STORE THE ARRAY BEING CONTOURED IN ZT AND CONTOUR ONE CELL AT A TIME
000052 001267 50 DO 90 NY=2,NOY
000053 001268 IX=MOD(NY,2)
000054 001269 DY=Y(NY)-Y(NY-1)
000055 001270 DO 80 INX=2,NOX
000056 001271 NX=INX

```

```

000057 001272 IF (Y.NE.0) NX=NCX-INY*2
000058 001273 ZT(1)=Z(NY-1,NY-1)
000059 001274 ZT(2)=Z(NX,NY-1)
000060 001275 ZT(3)=Z(NY,NY)
000061 001276 ZT(4)=Z(NY-1,NY)
000062 001277 DX=X(NX)-Y/NX-1
000063 001278 IF (ABS(ZT(3)-ZT(1))-ABS(ZT(4)-ZT(2))) 70,60,40
000064 001279 60 CALL TRICIR (Y(NX),Y(NY),-DX,-DY,NOC,ZPLAN,ZT(4),ZT(3),ZT(2))
000065 001280 CALL TRICIR (Y(NX-1),Y(NY-1),DX,DY,NOC,ZPLAN,ZT(2),ZT(1),ZT(4))
000066 001281 GO TO 60
000067 001282 70 CALL TRICIR (Y(NX-1),Y(NY),DX,-DY,NOC,ZPLAN,ZT(3),ZT(4),ZT(1))
000068 001283 CALL TRICIR (Y(NX),Y(NY-1),-DX,DY,NOC,ZPLAN,ZT(1),ZT(2),ZT(3))
000069 001284 80 CONTINUE
000070 001285 90 CONTINUE
000071 001286 C
000072 001287 C IDENTIFY THE MAX. MIN. AND SPACING FOR CONTOURS
000073 001288 REWIND 63
000074 001289 WRITE(63,900) ZPLAN(1),ZPLAN(NOC),DELZ
000075 001290 REWIND 63
000076 001291 READ(63,901) (CON(I),I=1,6)
000077 001292 CALL NLCH(48,1000,60,CON(1),2)
000078 001293 FORMAT(1X,15MIN,MAX,DELTA,1P3E12.4)
000079 001294 900 FORMAT(1X,6A10)
000080 001295 901 RETURN
000081 001296 C
000082 001297 END

```

1*****2*****

```

000001 001298 CHAT 1670 11:10:39 MON/06/79
000002 001299 C
000003 001300 C SURROUTINE TRICJB(X,Y,DX,DY,NOC,ZPLAN,ZX,ZY,ZT)
000004 001301 C
000005 001302 C
000006 001303 C PERFORM ACTUAL INTERPOLATIONS NEEDED TO PRODUCE CONTOURS
000007 001304 C DEFINED IN CONTRJB. PLOTH USED TO CONNECT POINTS.
000008 001305 C
000009 001306 COMMON /CONTRC/ ISYM(50)
000010 001307 DIMENSION YP(2,50), ZP(2,50), ZT(4), ZPLAN(1)
000011 001308 ZT(1)=ZX
000012 001309 ZT(2)=ZY
000013 001310 ZT(3)=ZX
000014 001311 ZT(4)=ZY
000015 001312 ZTMIN=AMIN1(ZT(1),ZT(2),ZT(3),ZT(4))
000016 001313 ZTMAX=AMAX1(ZT(1),ZT(2),ZT(3),ZT(4))
000017 001314 IMIN=NOC+1
000018 001315 IMAX=NOC
000019 001316 DO 10 K=1,NOC
000020 001317 J=NOC-K+1
000021 001318 IF (ZPLAN(J).GE.ZTMIN) IMIN=J
000022 001319 IF (ZPLAN(K).LE.ZTMAX) IMAX=K
000023 001320 10 CONTINUE
000024 001321 INT=IMAX-IMIN
000025 001322 IF (INT.LE.0.OR.ZTMIN.EQ.ZTMAX) GO TO 13

```

000026 001323
 000027 001324
 000028 001325
 000029 001326
 000030 001327
 000031 001328
 000032 001329
 000033 001330
 000034 001331
 000035 001332
 000036 001333
 000037 001334
 000038 001335
 000039 001336
 000040 001337
 000041 001338
 000042 001339
 000043 001340
 000044 001341
 000045 001342
 000046 001343
 000047 001344
 000048 001345
 000049 001346
 000050 001347
 000051 001348
 000052 001349
 000053 001350
 000054 001351
 000055 001352
 000056 001353

 000057 001354
 000058 001355
 000059 001356
 000060 001357
 000061 001358
 000062 001359
 000063 001360
 000064 001361
 000065 001362

```

I2=1
DO 110 K=1,3
  ZTMAX=AMAX1(ZT(K),ZT(K+1))
  ZPMIN=AMIN1(ZT(K),ZT(K+1))
  MIN=NOC+1
  MAX=0
  DO 20 J=1,NOC
    IN7=NOC-J+1
    IF (ZPLAN(IN7).GT.ZPMIN.OR.(ZPLAN(IN7).EQ.ZPMIN.AND.ZTMIN.EQ.ZPMIN
      1) MIN=IN7
    IF (ZPLAN(J).LT.ZTMAX) MAX=J
  20 CONTINUE
  IN7=MAX-MIN
  IF (IN7.LT.0.OR.ZTMAX.EQ.ZPMIN) GO TO 110
  IF (IN7=INT(.50.40) .I2
  30 GO TO (50.40) .I2
  40 I2=1
  GO TO 60
  50 I2=2
  60 DO 100 J=MIN,MAX
    GO TO (70.80.90) .K
  70 XP(I2,J)=Y*DX*(ZPLAN(J)-ZT(I2))/(ZT(1)-ZT(I2))
    YP(I2,J)=Y
    GO TO 100
  80 XP(I2,J)=Y*DY*(ZPLAN(J)-ZT(I2))/(ZT(3)-ZT(I2))
    YP(I2,J)=Y-DY*(ZPLAN(J)-ZT(I2))/(ZT(1)-ZT(I2))
  90 XP(I2,J)=Y*DX*(ZPLAN(J)-ZT(I3))/(ZT(1)-ZT(I3))
    YP(I2,J)=Y-DY*(ZPLAN(J)-ZT(I3))/(ZT(3)-ZT(I3))
  100 CONTINUE
  110 CONTINUE

  DO 120 J=MIN,MAX
    ISYM(J)=ISYM(J)+1
    L=3
    IF (MOD(ISYM(J),10).NE.1) L=0
    CALL PLNJA (XP,1,J),YP,1,J,2,1,L,J,0.,0.,0.,0.,0.,0.,0.,0.)
  120 CONTINUE
C
  130 RETURN
  ENN
  
```

APPENDIX D

OUTPUT EXAMPLES

Three sets of computer output are exhibited in this appendix, one for each version of the code.

GEØ-I

The first part of a printed output listing is shown for time $t = 0$, generated by the code and set-up described in Appendix A. Following that is part of the printed output for cycle number 100. This calculation produces the results described as example 3 in Table II and Fig. 6.

GEØ-II

The first part of a printed output listing is shown for time $t = 0$, generated by the code and set-up described in Appendix B. Following that is part of the printed output for cycle number 50. This calculation produces the results described as example 4 in Table II and Fig. 7.

GEØ-III

For the code GEØ-III, we exhibit the first part of each printed cycle showing the alternation between hydro cycles and rock temperature cycles. This calculation produces the results described as example 6 in Table III and Fig. 9.

GEØ-I

Example 3

Figure 6

TIME= 0.

NCYCLES= 0

NTT= 0

ENDOTIN= 3.25E+01

EODTOTUT 0.

EODTOTUT=3.25E+01

DSUM= 0.

RW= 1.00E+00
AKAPR= 9.36E-03
DX= 5.00E-01
DYSQ= 2.50E-01
EIGEIF= 5.33E-01
RNUA= 0.
THICK= 5.00E+00
WN= 0.

RR= 2.50E-01
AKAPO= 4.20E-03
DY= 5.00E-01
DYSQ= 2.50E-01
EPSILON= 1.00E-03
TIN= 6.50E+01
WR= 1.00E+01

RHOW= 1.00E+00
BETA= 2.07E-04
CT= 1.67E+00
DYSQ= 4.00E+00
GRAV= 9.80E+02
IF= 1.00E+02
TNOTHI= 6.47E-01
XTI= 3.00E+02

WHOR= 2.45E+00
HETAR= 2.50E-05
DXR= 2.00E+00
DYSQR= 4.00E+00
II= 2
IP= 0.
VIN= 1.00E-01
NIR= 23

AKR= 4.20E-03
NT= 1.00E-01
NYR= 2.00E+00
NZSQ= 3.40E-01
THAR= 20
TPC= 1.00E-04
HEIGHT= 5.00E+00
NJ= 23

OTR= 1.00E+01
OZR= 6.00E-01
JBAR= 20
TPI= 1.00E+01
NK= 10

CFLL	I	J	U	V	PHI	T	WN01	GAM	D	XI(I,J,2)	XI(I,J,3)	XI(I,J,4)
1	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	3	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	4	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	5	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	16	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	17	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	19	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	20	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	21	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

ENDOTIN= 3.25E+01 EDOTOUT 5.21E+01 EDDTOTIF 1.96E+01 DSDUM= 7.68E-15

[illegible]

2	1	0.	1.000E-01	0.	4.500E+01	1.000F+01	0.	0.	0.	0.	0.
2	2	3.258F-01	9.873E-02	-1.111E-01	8.799E+01	1.000F+01	9.375E-02	-1.815E-03	-2.192E+01	-2.345E+01	-2.347E-01
2	3	3.296F-01	9.727E-02	-1.109F-01	8.152E+01	1.000F+01	9.375E-02	-1.813F-03	1.456E+01	1.491E+01	1.288E-01
2	4	2.905F-01	9.588E-02	-1.106E-01	9.270E+01	1.000F+01	9.375E-02	-1.812E-03	2.848E+01	2.509E+01	2.506E-01
2	5	2.700F-01	9.451E-02	-1.103F-01	9.100E+01	1.000F+01	9.375E-02	-1.814E-03	3.311E+01	3.164E+01	3.162E-01
2	6	2.439F-01	9.298E-02	-1.099F-01	8.819E+01	1.000F+01	9.375E-02	-1.821F-03	3.845E+01	3.715E+01	3.713E-01
2	7	3.334F-01	9.111E-02	-1.095E-01	8.490E+01	1.000F+01	9.375E-02	-1.837E-03	4.388E+01	4.276E+01	4.276E-01
2	8	4.063F-01	8.971E-02	-1.094E-01	8.138E+01	1.000F+01	9.375E-02	-1.842E-03	4.949E+01	4.865E+01	4.864E-01
2	9	4.680F-01	8.581E-02	-1.094E-01	7.778E+01	1.000F+01	9.375E-02	-1.894E-03	5.497E+01	5.441E+01	5.440E-01
2	10	4.604F-01	8.281E-02	-1.097E-01	7.446E+01	1.000F+01	9.375E-02	-1.924E-03	5.929E+01	5.898E+01	5.898E-01
2	11	3.292F-01	8.058E-02	-1.101E-01	7.229E+01	1.000F+01	9.375E-02	-1.961F-03	6.011E+01	5.999E+01	5.999E-01
2	12	7.158F-02	7.994F-02	-1.107F-01	7.276E+01	1.000F+01	9.375E-02	-1.985E-03	5.593E+01	5.592E+01	5.592E-01
2	13	-1.506F-01	8.072E-02	-1.111E-01	7.568E+01	1.000F+01	9.375E-02	-1.997E-03	5.051E+01	5.039E+01	5.039E-01
2	14	-2.477F-01	8.219E-02	-1.112E-01	7.984E+01	1.000F+01	9.375E-02	-1.999E-03	4.566E+01	4.577E+01	4.577E-01
2	15	-4.276F-01	8.419E-02	-1.109F-01	8.432E+01	1.000F+01	9.375E-02	-1.994E-03	4.144E+01	4.156E+01	4.156E-01
2	16	-5.222F-01	8.461E-02	-1.100E-01	8.849E+01	1.000F+01	9.375E-02	-1.983F-03	3.763E+01	3.781E+01	3.781E-01
2	17	-6.618F-01	8.032E-02	-1.088F-01	9.275E+01	1.000F+01	9.375E-02	-1.972E-03	3.434E+01	3.447E+01	3.447E-01
2	18	-7.404F-01	9.213F-02	-1.073E-01	9.638E+01	1.000F+01	9.375E-02	-1.962E-03	3.133E+01	3.149E+01	3.149E-01
2	19	-7.741F-01	9.487E-02	-1.059F-01	9.952E+01	1.000F+01	9.375E-02	-1.958E-03	2.848E+01	2.870E+01	2.870E-01
2	20	-7.991F-01	9.733E-02	-1.050E-01	1.022E+02	1.000F+01	9.375E-02	-1.959E-03	2.443E+01	2.471E+01	2.472E-01
2	21	-6.919F-01	9.015E-02	-1.047E-01	1.050E+02	1.000F+01	9.375E-02	-1.962E-03	8.982E+00	9.198E+00	9.201E+00
2	22	0.	0.	0.	0.	1.000F+01	0.	0.	0.	0.	0.
3	1	0.	0.	0.	0.	0.	0.	0.	3.000E+02	3.000E+02	3.000E+02
3	2	4.330F-01	9.263F-02	-6.488E+00	1.245E+02	3.466F-02	2.705F+01	-6.447F-05	1.895E+02	1.911E+02	1.911E+02
3	3	5.168F-01	1.008E-01	-3.797E+00	1.169E+02	4.038F-02	2.222F+01	1.631E-04	1.679E+02	1.695E+02	1.696E+02
3	4	4.896F-01	6.762E-02	-2.791E+00	1.121E+02	4.306F-02	2.177E+01	4.075F-04	1.580E+02	1.596E+02	1.596E+02
3	5	4.634F-01	5.944E-02	-7.020E-01	1.045E+02	4.581F-02	2.047E+01	4.264F-04	1.482E+02	1.497E+02	1.497E+02
3	6	4.401F-01	-3.485E-02	5.220E-01	9.800E+01	4.888F-02	1.918F+01	7.020F-04	1.373E+02	1.386E+02	1.386E+02
3	7	4.300F-01	-3.771E-02	1.591E+00	9.136E+01	5.216F-02	1.797E+01	8.763F-04	1.254E+02	1.265F+02	1.265E+02
3	8	4.332F-01	1.403E-02	2.770E+00	8.550E+01	5.559F-02	1.686F+01	8.537E-04	1.129E+02	1.137F+02	1.137E+02
3	9	4.304F-01	1.141E-01	2.719E+00	8.060E+01	5.898F-02	1.589F+01	7.194F-04	1.003E+02	1.008F+02	1.008E+02
3	10	3.427F-01	2.344F-01	2.548E+00	7.697E+01	6.189F-02	1.515F+01	5.049F-04	8.942E+01	8.973E+01	8.974E+01
3	11	2.526F-01	3.447E-01	1.906E+00	7.541E+01	6.325F-02	1.482F+01	2.823F-04	8.434E+01	8.448E+01	8.449E+01
3	12	2.165F-02	4.212E-01	1.420E+00	7.889E+01	6.184F-02	1.516F+01	1.331E-04	8.948E+01	8.947F+01	8.947E+01
3	13	-2.660F-01	5.256E-01	6.616E-01	9.778F+01	5.844F-02	1.604F+01	1.077E-04	1.006E+02	1.006F+02	1.006E+02
3	14	-5.310F-01	7.298E-01	1.383E+00	1.132E+02	5.436F-02	1.724F+01	1.753F-04	1.137E+02	1.136E+02	1.136E+02
3	15	-7.373F-01	9.516E-01	2.013E+00	1.316E+02	5.033F-02	1.863F+01	2.690F-04	1.269E+02	1.267E+02	1.267E+02
3	16	-8.763F-01	1.127E+00	4.833E+00	1.480E+02	4.655F-02	2.014F+01	3.440F-04	1.392E+02	1.391E+02	1.391E+02
3	17	-9.386F-01	1.189E+00	7.020E+00	1.627E+02	4.310F-02	2.175F+01	3.871E-04	1.505E+02	1.504F+02	1.504E+02
3	18	-9.192F-01	1.095E+00	4.770E+00	1.733E+02	4.000F-02	2.344E+01	4.089F-04	1.607E+02	1.605E+02	1.605E+02
3	19	-8.303F-01	8.474E-01	1.372E+01	1.892E+02	3.720F-02	2.520F+01	4.320E-04	1.698E+02	1.696E+02	1.696E+02
3	20	-6.921F-01	5.131E-01	1.961E+01	2.018F+02	3.434F-02	2.730F+01	4.709F-04	1.792E+02	1.789E+02	1.789E+02
3	21	-4.358F-01	0.	2.798E+01	2.144E+02	2.854F-02	3.285E+01	4.705E-04	2.002E+02	2.000E+02	2.000E+02
3	22	0.	0.	0.	0.	0.	0.	0.	3.000E+02	3.000E+02	3.000E+02

4	1	0.	0.	0.	0.	0.	0.	0.	0.	3.000E+02	3.000E+02	3.000E+02
4	2	3.691E-01	1.577E-01	-1.763E+01	1.812E+02	1.718E-02	5.456E+01	4.896E-05	2.498E+02	2.516E+02	2.516E+02	2.516E+02
4	3	5.200E-01	2.050E-01	-1.144E+01	1.573E+02	2.358E-02	3.977E+01	1.813E-04	2.280E+02	2.302E+02	2.302E+02	2.302E+02
4	4	5.427E-01	2.177E-01	-7.791E+00	1.442E+02	2.782E-02	3.370E+01	3.190E-04	2.132E+02	2.154E+02	2.154E+02	2.154E+02
4	5	5.381E-01	1.033E-01	-5.091E+00	1.319E+02	3.213E-02	2.918E+01	4.487E-04	1.981E+02	2.002E+02	2.003E+02	2.003E+02
4	6	5.211E-01	1.613E-01	-2.042E+00	1.106E+02	3.680E-02	2.548E+01	5.506E-04	1.814E+02	1.836E+02	1.836E+02	1.836E+02
4	7	4.949E-01	1.405E-01	-1.274E+00	1.079E+02	4.174E-02	2.246E+01	6.040E-04	1.640E+02	1.656E+02	1.657E+02	1.657E+02
4	8	4.660E-01	1.510E-01	-4.929E-02	9.798E+01	4.415E-02	2.005E+01	5.907E-04	1.454E+02	1.471E+02	1.472E+02	1.472E+02
4	9	4.181E-01	2.013E-01	5.861E-01	9.043E+01	5.145E-02	1.822E+01	5.043E-04	1.285E+02	1.294E+02	1.295E+02	1.295E+02
4	10	3.249E-01	2.831E-01	8.471E-01	8.578E+01	5.515E-02	1.700E+01	3.416E-04	1.147E+02	1.153E+02	1.153E+02	1.153E+02
4	11	1.737E-01	3.018E-01	8.506E-01	8.533E+01	5.661E-02	1.656E+01	2.157E-04	1.091E+02	1.094E+02	1.094E+02	1.094E+02
4	12	-6.821E-02	5.154E-01	9.644E-01	9.450E+01	5.474E-02	1.713E+01	1.261E-04	1.152E+02	1.153E+02	1.153E+02	1.153E+02
4	13	-3.625E-01	6.384E-01	1.798E+00	1.121E+02	5.047E-02	1.858E+01	1.279E-04	1.294E+02	1.294E+02	1.294E+02	1.294E+02
4	14	-6.184E-01	7.460E-01	3.656E+00	1.325E+02	4.519E-02	2.075E+01	1.941E-04	1.471E+02	1.472E+02	1.472E+02	1.472E+02
4	15	-7.898E-01	8.132E-01	6.391E+00	1.524E+02	3.973E-02	2.360E+01	2.730E-04	1.655E+02	1.657E+02	1.657E+02	1.657E+02
4	16	-8.134E-01	8.145E-01	4.730E+00	1.710E+02	3.447E-02	2.720E+01	3.315E-04	1.834E+02	1.836E+02	1.837E+02	1.837E+02
4	17	-8.797E-01	7.406E-01	1.357E+01	1.884E+02	2.957E-02	3.170E+01	3.620E-04	2.001E+02	2.003E+02	2.003E+02	2.003E+02
4	18	-8.208E-01	6.010E-01	1.009E+01	2.058E+02	2.509E-02	3.736E+01	3.724E-04	2.152E+02	2.154E+02	2.154E+02	2.154E+02
4	19	-7.114E-01	4.156E-01	2.383E+01	2.228E+02	2.104E-02	4.457E+01	3.745E-04	2.284E+02	2.287E+02	2.287E+02	2.287E+02
4	20	-5.581E-01	2.121E-01	3.162E+01	2.396E+02	1.716E-02	5.463E+01	3.704E-04	2.414E+02	2.414E+02	2.414E+02	2.414E+02
4	21	-3.104E-01	0.	4.281E+01	2.581E+02	1.188E-02	7.891E+01	3.222E-04	2.594E+02	2.594E+02	2.594E+02	2.594E+02
4	22	0.	0.	0.	0.	0.	0.	0.	3.000E+02	3.000E+02	3.000E+02	3.000E+02
5	1	0.	0.	0.	0.	0.	0.	0.	3.000E+02	3.000E+02	3.000E+02	3.000E+02
5	2	2.812E-01	1.510E-01	-3.131E+01	2.261E+02	1.225E-02	7.653E+01	1.577E-04	2.625E+02	2.637E+02	2.637E+02	2.637E+02
5	3	4.611E-01	2.310E-01	-2.128E+01	1.926E+02	1.852E-02	5.061E+01	2.399E-04	2.427E+02	2.443E+02	2.443E+02	2.443E+02
5	4	5.203E-01	2.022E-01	-1.515E+01	1.734E+02	2.297E-02	4.081E+01	3.214E-04	2.280E+02	2.298E+02	2.298E+02	2.298E+02
5	5	5.445E-01	3.035E-01	-1.074E+01	1.569E+02	2.742E-02	3.419E+01	3.932E-04	2.130E+02	2.148E+02	2.148E+02	2.148E+02
5	6	5.470E-01	3.053E-01	-7.357E+00	1.411E+02	3.213E-02	2.918E+01	4.349E-04	1.969E+02	1.986E+02	1.986E+02	1.986E+02
5	7	5.275E-01	3.035E-01	-4.797E+00	1.267E+02	3.649E-02	2.534E+01	4.395E-04	1.800E+02	1.815E+02	1.815E+02	1.815E+02
5	8	4.836E-01	3.153E-01	-2.946E+00	1.144E+02	4.114E-02	2.246E+01	3.978E-04	1.632E+02	1.644E+02	1.644E+02	1.644E+02
5	9	4.083E-01	3.521E-01	-1.667E+00	1.054E+02	4.596E-02	2.040E+01	3.184E-04	1.480E+02	1.490E+02	1.490E+02	1.490E+02
5	10	2.868E-01	4.148E-01	-7.589E-01	1.005E+02	4.900E-02	1.913E+01	2.203E-04	1.368E+02	1.375E+02	1.375E+02	1.375E+02
5	11	9.412E-02	5.138E-01	8.348E-02	1.016E+02	5.003E-02	1.874E+01	1.405E-04	1.326E+02	1.331E+02	1.331E+02	1.331E+02
5	12	-1.529E-01	6.272E-01	1.331E+00	1.135E+02	4.841E-02	1.936E+01	1.109E-04	1.373E+02	1.375E+02	1.375E+02	1.375E+02
5	13	-4.209E-01	7.088E-01	3.456E+00	1.309E+02	4.481E-02	2.092E+01	1.444E-04	1.488E+02	1.490E+02	1.490E+02	1.490E+02
5	14	-6.298E-01	7.423E-01	6.419E+00	1.501E+02	4.014E-02	2.336E+01	2.172E-04	1.643E+02	1.644E+02	1.644E+02	1.644E+02
5	15	-7.451E-01	7.224E-01	1.013E+01	1.688E+02	3.510E-02	2.671E+01	2.918E-04	1.813E+02	1.815E+02	1.815E+02	1.815E+02
5	16	-7.744E-01	6.500E-01	1.565E+01	1.865E+02	3.010E-02	3.115E+01	3.441E-04	1.984E+02	1.986E+02	1.986E+02	1.986E+02
5	17	-7.395E-01	5.336E-01	2.132E+01	2.074E+02	2.534E-02	3.700E+01	3.490E-04	2.144E+02	2.148E+02	2.148E+02	2.148E+02
5	18	-6.600E-01	3.879E-01	2.796E+01	2.201E+02	2.093E-02	4.480E+01	3.727E-04	2.295E+02	2.297E+02	2.297E+02	2.297E+02
5	19	-5.505E-01	2.317E-01	3.605E+01	2.366E+02	1.641E-02	5.544E+01	3.624E-04	2.428E+02	2.430E+02	2.430E+02	2.430E+02
5	20	-4.147E-01	9.081E-02	4.452E+01	2.524E+02	1.313E-02	7.138E+01	3.376E-04	2.553E+02	2.554E+02	2.554E+02	2.554E+02
5	21	-2.172E-01	0.	6.144E+01	2.698E+02	8.452E-03	1.109E+02	2.631E-04	2.712E+02	2.712E+02	2.712E+02	2.712E+02
5	22	0.	0.	0.	0.	0.	0.	0.	3.000E+02	3.000E+02	3.000E+02	3.000E+02
6	1	0.	0.	0.	0.	0.	0.	0.	3.000E+02	3.000E+02	3.000E+02	3.000E+02

GEØ-II

Example 4

Figure 7

1 TIME= 0. NCYCLE= 0 NIT= 0

FNOTOIF 0. DSUM= 0.

AW= 1.00F+00 AR= 2.50E-01 R40W= 1.00F+00 RHOR= 2.65F+00 AKR= 6.20F-03
 AKAPR= 9.36E-03 AKAPR= 4.20E-03 3ETA= 2.07F-04 METAR= 2.50F-05 NT= 5.20F-01 NTR= 1.95E+00
 DX= 1.00F+01 DY= 1.00F+01 DT= 5.00E+00 NXR= 1.00F-01 NYR= 1.00F-01 NZR= 2.00E-01
 NYSQ= 1.00F+02 NYSQ= 1.00E+02 DVSQR= 1.00E-02 NYSQR= 1.00E-02 NZSOR= 4.00F-02
 FIGFIF= 5.33F-01 F25JION= 2.00E-03 GRAV= 9.80E+02 YBAR= 20 KBAR= 4
 RNUA= 1.00F-02 TI= 0. TF= 2.60E+02 TP= 0. TPC= 5.20F-04 TPI= 2.60E+01
 THICK= 5.00E+01 TIN= 6.50E+01 TWOTHI= 6.67F-01 VIN= 1.00F+01 NI= 23
 WN= 0. NC= 3.00E+01 KTI= 3.00F+02 NJ= 23 NK= 15

CFLL	I	J	U	V	PHI	T	WNO1	GAM	D	XI(I,J,2)	XI(I,J,3)	XI(I,J,4)
1	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	3	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	4	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	5	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	16	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	17	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	19	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	20	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	21	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

TIME= 2.59871E+01

ICYCLF 50

NTT=

FD0TDIF 2.04F+03

DSUM-2.54E-03

[illegible]

2	1	0.	1.000E+01	0.	0.	0.	0.	0.	0.	0.	0.
2	2	2.601E+00	1.000E+01	1.532E+01	6.500E+01	1.856E-01	0.	0.	0.	0.	0.
2	3	1.931E+00	1.000E+01	7.774E+00	6.500E+01	1.856E-01	0.	0.	0.	0.	0.
2	4	1.385E+00	1.000E+01	4.012E+00	6.500E+01	1.855E-01	0.	0.	0.	0.	0.
2	5	1.105E+00	1.000E+01	4.017E+00	6.500E+01	1.854E-01	0.	0.	0.	0.	0.
2	6	9.445E-01	1.000E+01	3.468E+00	6.500E+01	1.854E-01	0.	0.	0.	0.	0.
2	7	8.358E-01	1.000E+01	3.055E+00	6.500E+01	1.853E-01	0.	0.	0.	0.	0.
2	8	7.564E-01	1.000E+01	2.742E+00	6.500E+01	1.853E-01	0.	0.	0.	0.	0.
2	9	6.969E-01	1.000E+01	2.503E+00	6.500E+01	1.853E-01	0.	0.	0.	0.	0.
2	10	6.514E-01	1.000E+01	2.319E+00	6.500E+01	1.852E-01	0.	0.	0.	0.	0.
2	11	6.161E-01	1.000E+01	2.172E+00	6.500E+01	1.852E-01	0.	0.	0.	0.	0.
2	12	5.881E-01	1.000E+01	2.053E+00	6.500E+01	1.851E-01	0.	0.	0.	0.	0.
2	13	5.656E-01	1.000E+01	1.954E+00	6.500E+01	1.851E-01	0.	0.	0.	0.	0.
2	14	5.470E-01	1.000E+01	1.869E+00	6.500E+01	1.850E-01	0.	0.	0.	0.	0.
2	15	5.312E-01	1.000E+01	1.795E+00	6.500E+01	1.850E-01	0.	0.	0.	0.	0.
2	16	5.230E-01	1.000E+01	1.761E+00	6.500E+01	1.850E-01	0.	0.	0.	0.	0.
2	17	5.697E-01	1.000E+01	2.004E+00	6.500E+01	1.849E-01	0.	0.	0.	0.	0.
2	18	8.819E-01	1.000E+01	3.562E+00	6.500E+01	1.850E-01	0.	0.	0.	0.	0.
2	19	1.503E+00	1.000E+01	5.999E+00	6.500E+01	1.846E-01	0.	0.	0.	0.	0.
2	20	-1.204E+00	1.000E+01	-3.418E+00	6.500E+01	1.811E-01	0.	0.	0.	0.	0.
2	21	-1.620E+01	1.000E+01	-4.984E+01	6.500E+01	1.767E-01	0.	0.	0.	0.	0.
2	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	2	2.390E+00	7.119E-01	-1.532E+01	6.485E+01	1.856E-01	3.887E+02	4.537E-06	9.120E+01	9.142E+01	9.142E.01
3	3	2.051E+00	1.023E+00	-7.774E+00	6.726E+01	1.856E-01	3.888E+02	3.007E-06	9.120E+01	9.142E+01	9.142E.01
3	4	1.662E+00	1.097E+00	-4.012E+00	6.779E+01	1.855E-01	3.889E+02	9.542E-07	9.121E+01	9.142E+01	9.142E.01
3	5	1.363E+00	1.127E+00	-4.017E+00	6.828E+01	1.854E-01	3.891E+02	-1.081E-06	9.122E+01	9.142E+01	9.142E.01
3	6	1.152E+00	1.164E+00	-3.468E+00	6.871E+01	1.854E-01	3.892E+02	-3.004E-06	9.123E+01	9.142E+01	9.142E.01
3	7	1.003E+00	1.211E+00	-3.055E+00	6.909E+01	1.853E-01	3.893E+02	-4.776E-06	9.124E+01	9.142E+01	9.142E.01
3	8	8.941E-01	1.264E+00	-2.742E+00	6.946E+01	1.853E-01	3.894E+02	-4.353E-06	9.125E+01	9.142E+01	9.142E.01
3	9	8.137E-01	1.322E+00	-2.503E+00	6.980E+01	1.853E-01	3.895E+02	-7.711E-06	9.126E+01	9.142E+01	9.142E.01
3	10	7.528E-01	1.382E+00	-2.319E+00	7.014E+01	1.852E-01	3.896E+02	-8.856E-06	9.127E+01	9.142E+01	9.142E.01
3	11	7.056E-01	1.444E+00	-2.172E+00	7.046E+01	1.852E-01	3.897E+02	-9.814E-06	9.127E+01	9.142E+01	9.142E.01
3	12	6.676E-01	1.508E+00	-2.053E+00	7.078E+01	1.851E-01	3.897E+02	-1.062E-05	9.128E+01	9.142E+01	9.142E.01
3	13	6.358E-01	1.574E+00	-1.954E+00	7.109E+01	1.851E-01	3.898E+02	-1.127E-05	9.129E+01	9.142E+01	9.142E.01
3	14	6.074E-01	1.645E+00	-1.869E+00	7.141E+01	1.850E-01	3.899E+02	-1.178E-05	9.129E+01	9.142E+01	9.142E.01
3	15	5.810E-01	1.720E+00	-1.795E+00	7.174E+01	1.850E-01	3.900E+02	-1.230E-05	9.130E+01	9.142E+01	9.142E.01
3	16	5.590E-01	1.805E+00	-1.761E+00	7.205E+01	1.850E-01	3.901E+02	-1.278E-05	9.131E+01	9.142E+01	9.142E.01
3	17	5.631E-01	1.932E+00	-2.004E+00	7.213E+01	1.849E-01	3.901E+02	-1.705E-05	9.132E+01	9.142E+01	9.142E.01
3	18	5.021E-01	2.426E+00	-3.562E+00	7.176E+01	1.850E-01	3.909E+02	-1.844E-05	9.133E+01	9.142E+01	9.142E.01
3	19	-7.354E-01	4.582E+00	-5.999E+00	7.444E+01	1.846E-01	3.908E+02	-7.155E-06	9.143E+01	9.142E+01	9.142E.01
3	20	-5.292E+00	7.631E+00	3.418E+00	1.017E+02	1.811E-01	3.983E+02	2.181E-05	9.144E+01	9.143E+01	9.142E.01
3	21	-1.123E+01	0.	4.984E+01	1.361E+02	1.767E-01	4.082E+02	4.800E-05	9.195E+01	9.143E+01	9.142E.01
3	22	0.	0.	0.	0.	0.	0.	0.	9.194E+01	9.143E+01	9.142E.01

4	1	0.	0.	0.	0.	0.	0.	0.	1.416E+02	1.422E+02	1.422E+02
4	2	2.619E+00	4.018E-01	-5.140E+01	7.388E+01	1.468E-01	4.916E+02	1.210E-05	1.416E+02	1.422E+02	1.422E+02
4	3	2.072E+00	8.841E-01	-3.791E+01	7.498E+01	1.466E-01	4.921E+02	6.886E-06	1.416E+02	1.422E+02	1.422E+02
4	4	1.659E+00	1.293E+00	-2.913E+01	7.643E+01	1.464E-01	4.927E+02	1.880E-07	1.416E+02	1.422E+02	1.422E+02
4	5	1.355E+00	1.635E+00	-2.940E+01	7.790E+01	1.462E-01	4.934E+02	-6.261E-06	1.417E+02	1.422E+02	1.422E+02
4	6	1.136E+00	1.933E+00	-1.949E+01	7.931E+01	1.461E-01	4.940E+02	-1.226E-05	1.417E+02	1.422E+02	1.422E+02
4	7	9.790E-01	2.201E+00	-1.670E+01	8.063E+01	1.459E-01	4.945E+02	-1.774E-05	1.417E+02	1.422E+02	1.422E+02
4	8	8.656E-01	2.447E+00	-1.467E+01	8.186E+01	1.457E-01	4.951E+02	-2.255E-05	1.417E+02	1.422E+02	1.422E+02
4	9	7.822E-01	2.675E+00	-1.215E+01	8.302E+01	1.456E-01	4.956E+02	-2.665E-05	1.418E+02	1.422E+02	1.422E+02
4	10	7.193E-01	2.889E+00	-1.198E+01	8.411E+01	1.454E-01	4.961E+02	-3.004E-05	1.418E+02	1.422E+02	1.422E+02
4	11	6.702E-01	3.094E+00	-1.107E+01	8.515E+01	1.453E-01	4.966E+02	-3.280E-05	1.418E+02	1.422E+02	1.422E+02
4	12	6.297E-01	3.291E+00	-1.032E+01	8.614E+01	1.452E-01	4.970E+02	-3.502E-05	1.418E+02	1.422E+02	1.422E+02
4	13	5.934E-01	3.484E+00	-9.484E+00	8.710E+01	1.450E-01	4.974E+02	-3.683E-05	1.418E+02	1.422E+02	1.422E+02
4	14	5.567E-01	3.677E+00	-9.116E+00	8.805E+01	1.449E-01	4.979E+02	-3.832E-05	1.419E+02	1.422E+02	1.422E+02
4	15	5.134E-01	3.875E+00	-8.887E+00	8.899E+01	1.448E-01	4.983E+02	-3.999E-05	1.419E+02	1.422E+02	1.422E+02
4	16	4.546E-01	4.094E+00	-8.158E+00	8.992E+01	1.447E-01	4.987E+02	-4.383E-05	1.419E+02	1.422E+02	1.422E+02
4	17	2.792E-01	4.457E+00	-8.726E+00	9.070E+01	1.446E-01	4.991E+02	-5.085E-05	1.419E+02	1.422E+02	1.422E+02
4	18	-4.658E-01	5.763E+00	-8.431E+00	9.311E+01	1.442E-01	5.004E+02	-4.980E-05	1.420E+02	1.422E+02	1.422E+02
4	19	-2.281E+00	6.417E+00	-3.458E+00	1.082E+02	1.423E-01	5.070E+02	-1.192E-05	1.421E+02	1.422E+02	1.422E+02
4	20	-5.328E+00	5.155E+00	5.899E+01	1.315E+02	1.393E-01	5.178E+02	7.151E-05	1.423E+02	1.422E+02	1.422E+02
4	21	-8.407E+00	0.	1.450E+02	1.668E+02	1.349E-01	5.350E+02	1.454E-04	1.426E+02	1.422E+02	1.422E+02
4	22	0.	0.	0.	0.	0.	0.	0.	1.426E+02	1.422E+02	1.422E+02
5	1	0.	0.	0.	0.	0.	0.	0.	1.864E+02	1.873E+02	1.873E+02
5	2	2.533E+00	7.768E-01	-1.125E+02	8.625E+01	1.115E-01	6.473E+02	1.040E-05	1.864E+02	1.873E+02	1.873E+02
5	3	1.902E+00	1.475E+00	-8.411E+01	8.899E+01	1.111E-01	6.493E+02	1.236E-06	1.865E+02	1.873E+02	1.873E+02
5	4	1.460E+00	2.088E+00	-6.463E+01	9.189E+01	1.107E-01	6.515E+02	-9.557E-06	1.865E+02	1.873E+02	1.873E+02
5	5	1.151E+00	2.623E+00	-5.130E+01	9.476E+01	1.104E-01	6.537E+02	-1.927E-05	1.865E+02	1.873E+02	1.873E+02
5	6	9.368E-01	3.097E+00	-4.207E+01	9.752E+01	1.100E-01	6.558E+02	-2.784E-05	1.866E+02	1.873E+02	1.873E+02
5	7	7.875E-01	3.522E+00	-3.555E+01	1.001E+02	1.097E-01	6.578E+02	-3.534E-05	1.866E+02	1.873E+02	1.873E+02
5	8	6.823E-01	3.910E+00	-3.885E+01	1.026E+02	1.094E-01	6.597E+02	-4.168E-05	1.867E+02	1.873E+02	1.873E+02
5	9	6.065E-01	4.269E+00	-2.737E+01	1.049E+02	1.091E-01	6.615E+02	-4.687E-05	1.867E+02	1.873E+02	1.873E+02
5	10	5.499E-01	4.607E+00	-2.472E+01	1.070E+02	1.088E-01	6.632E+02	-5.095E-05	1.867E+02	1.873E+02	1.873E+02
5	11	5.049E-01	4.927E+00	-2.263E+01	1.091E+02	1.085E-01	6.648E+02	-5.401E-05	1.868E+02	1.873E+02	1.873E+02
5	12	4.656E-01	5.235E+00	-2.092E+01	1.111E+02	1.083E-01	6.664E+02	-5.623E-05	1.868E+02	1.873E+02	1.873E+02
5	13	4.257E-01	5.535E+00	-1.942E+01	1.130E+02	1.080E-01	6.679E+02	-5.788E-05	1.868E+02	1.873E+02	1.873E+02
5	14	3.783E-01	5.833E+00	-1.800E+01	1.149E+02	1.078E-01	6.695E+02	-5.937E-05	1.869E+02	1.873E+02	1.873E+02
5	15	3.080E-01	6.137E+00	-1.652E+01	1.168E+02	1.075E-01	6.710E+02	-6.131E-05	1.869E+02	1.873E+02	1.873E+02
5	16	1.433E-01	6.512E+00	-1.486E+01	1.188E+02	1.073E-01	6.727E+02	-6.441E-05	1.869E+02	1.873E+02	1.873E+02
5	17	-2.853E-01	7.850E+00	-1.156E+01	1.228E+02	1.067E-01	6.759E+02	-6.591E-05	1.870E+02	1.873E+02	1.873E+02
5	18	-1.125E+00	7.456E+00	9.286E-01	1.321E+02	1.055E-01	6.836E+02	-5.120E-05	1.871E+02	1.873E+02	1.873E+02
5	19	-2.444E+00	6.976E+00	3.000E+01	1.465E+02	1.037E-01	6.958E+02	5.707E-06	1.872E+02	1.873E+02	1.873E+02
5	20	-4.107E+00	4.471E+00	1.308E+02	1.668E+02	1.011E-01	7.136E+02	1.111E-04	1.873E+02	1.873E+02	1.873E+02
5	21	-5.665E+00	0.	3.187E+02	1.958E+02	9.744E-02	7.404E+02	2.035E-04	1.875E+02	1.873E+02	1.873E+02
5	22	0.	0.	0.	0.	0.	0.	0.	1.875E+02	1.873E+02	1.873E+02
6	1	0.	0.	0.	0.	0.	0.	0.	2.233E+02	2.243E+02	2.243E+02

GE0-III

Example 6

Figure 9

1 TIME= 0. TOTAL TIME= 0. CYCLE= 0 NIT= 0

ENDTIME 0. PSUM= 0.

BW= 1.00E+00 BR= 2.00E-01 BQW= 1.00E+00 BQW= 2.65E+00 AKR= 6.00E-03
 AKAP= 9.00E-03 AKAP= 6.00E-03 BETA= 2.00E-01 BETA= 2.50E-05 DT= 8.60E-01 DTR= 1.10E+00
 DX= 2.00E+01 DY= 1.00E+01 DZ= 8.35E+00 DAK= 4.00E-02 DYR= 1.00E-01 DZR= 1.20E-01
 DXSQ= 6.00E+02 DYSQ= 1.00E+02 DXSOR= 1.00E-03 DYSOR= 1.00E-02 DZSHK= 1.44E-04
 FTGFI= 5.33E-01 PSILCN= 2.00E-03 GRAV= 9.80E+02 ISAR= 20 JHAK= 50 KBAK= 4
 PNUA= 1.00E-02 T= 0. NCYF= 1000 NCYP= 0 NCYI= 1000
 THICK= 5.00E+01 LIN= 1.50E+02 TWOTHI= 6.67E-01 VIN= 0. HEIGHT= 1.00E+10
 W0= 0. MC= 5.00E+01 XII= 3.00E+02 NI= 23 NJ= 23 NK= 10

CELL	I	J	U	V	PHI	T	WNP1	GAM	D	XI(I,J,2)	XI(I,J,3)	XI(I,J,4)
1	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	3	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	4	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	5	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	16	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	17	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	19	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	20	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	21	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

FN0T0IF 9.91F+02 DSUM=-3.34E-06

[illegible]

2	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	2	2.862E+00	0.	2.159F+01	1.500E+02	1.660E-01	0.	0.	0.	0.	0.
2	3	2.264E+00	0.	1.410F+01	1.500E+02	1.659E-01	0.	0.	0.	0.	0.
2	4	1.841E+00	0.	1.243F+01	1.500E+02	1.658E-01	0.	0.	0.	0.	0.
2	5	1.517E+00	0.	9.921F+00	1.500E+02	1.656E-01	0.	0.	0.	0.	0.
2	6	1.284E+00	0.	8.158F+00	1.500E+02	1.655E-01	0.	0.	0.	0.	0.
2	7	1.112E+00	0.	6.884F+00	1.500E+02	1.654E-01	0.	0.	0.	0.	0.
2	8	9.800E-01	0.	5.937F+00	1.500E+02	1.653E-01	0.	0.	0.	0.	0.
2	9	8.782E-01	0.	5.210F+00	1.500E+02	1.652E-01	0.	0.	0.	0.	0.
2	10	7.926E-01	0.	4.632F+00	1.500E+02	1.651E-01	0.	0.	0.	0.	0.
2	11	7.220E-01	0.	4.158E+00	1.500E+02	1.650E-01	0.	0.	0.	0.	0.
2	12	6.646E-01	0.	3.764F+00	1.500E+02	1.649E-01	0.	0.	0.	0.	0.
2	13	6.203E-01	0.	3.466F+00	1.500E+02	1.648E-01	0.	0.	0.	0.	0.
2	14	6.084E-01	0.	3.382E+00	1.500E+02	1.647E-01	0.	0.	0.	0.	0.
2	15	6.488E-01	0.	3.610F+00	1.500E+02	1.646E-01	0.	0.	0.	0.	0.
2	16	7.103E-01	0.	3.882F+00	1.500E+02	1.644E-01	0.	0.	0.	0.	0.
2	17	6.663E-01	0.	3.422F+00	1.500E+02	1.640E-01	0.	0.	0.	0.	0.
2	18	1.270E-01	0.	5.088E-02	1.500E+02	1.633E-01	0.	0.	0.	0.	0.
2	19	-1.606E+00	0.	-9.942F+00	1.500E+02	1.624E-01	0.	0.	0.	0.	0.
2	20	-5.373E+01	0.	-3.572F+01	1.500E+02	1.612E-01	0.	0.	0.	0.	0.
2	21	-1.191F+01	0.	-8.159E+01	1.500E+02	1.593E-01	0.	0.	0.	0.	0.
2	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	2	2.563E+00	5.518E-01	-2.159F+01	1.544E+02	1.660E-01	4.515E+02	1.999E-10	1.914E+02	1.914E+02	1.914E+02
3	3	2.013E+00	1.173E+00	-1.410F+01	1.544E+02	1.659E-01	4.518E+02	2.129E-10	1.914E+02	1.914E+02	1.914E+02
3	4	1.597E+00	1.693E+00	-1.243F+01	1.554E+02	1.658E-01	4.522E+02	1.971E-10	1.914E+02	1.914E+02	1.914E+02
3	5	1.284E+00	1.968E+00	-9.921F+00	1.560E+02	1.656E-01	4.525E+02	1.491E-10	1.914E+02	1.914E+02	1.914E+02
3	6	1.053E+00	2.247E+00	-8.158F+00	1.564E+02	1.655E-01	4.529E+02	1.385E-10	1.914E+02	1.914E+02	1.914E+02
3	7	8.772E-01	2.520E+00	-6.884F+00	1.574E+02	1.654E-01	4.532E+02	1.106E-10	1.914E+02	1.914E+02	1.914E+02

TIME = 0.

TOTAL TIME = 1.6545E+05

NCYCLE = 0

NII = 1

OTDIF = 6.66E+02

DSUM = -3.34E-06

BW = 7.00E+00 BX = 2.50E-01 BROW = 1.0E+00 RMOR = 2.65E+00 AKW = 6.50E-03
 AAAPD = 9.46E-03 AAAPD = 6.2E-03 AETA = 2.0E-04 BCIAM = 2.50E-05 DT = 8.63E-01 UTH = 1.16E+00
 DX = 2.50E+01 DY = 1.0E+01 DZ = 8.33E+00 DAK = 4.00E-02 DYH = 1.00E-01 UZH = 1.20E-01
 DYSO = 6.2E+02 DYSO = 1.00E+02 DYSOR = 1.00E-03 DYSOR = 1.00E-02 DZSOH = 1.44E-02
 DIGFIF = 5.33E-01 DIGFIF = 2.0E-03 GRAV = 9.40E+02 IDAH = 20 JHAM = 20 KBAM = 4
 RMUA = 7.00E-02 TI = 0 NCYF = 1000 NCYP = 0 NCYI = 1000
 THICK = 5.00E+01 TH = 1.50E+02 TWOTHT = 6.67E-01 VIN = 0 HEIGHT = 1.00E+10
 W0 = 0 DC = 5.00E+01 XII = 3.00E+02 NI = 23 NJ = 23 NK = 10

CELL	I	J	U	V	PHI	T	WNP1	GAM	D	XI(I,J,2)	YI(I,J,3)	XI(I,J,4)
1	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	3	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	4	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	5	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	16	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	17	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	19	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	20	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	21	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	2	2.862E+00	0.	0.	0.	1.500E+02	2.043E-01	0.	0.	1.414E+02	1.387E+02	1.375E+02
2	3	2.264E+00	0.	0.	0.	1.500E+02	2.038E-01	0.	0.	1.410E+02	1.384E+02	1.372E+02
2	4	1.831E+00	0.	0.	0.	1.500E+02	2.031E-01	0.	0.	1.405E+02	1.380E+02	1.368E+02
2	5	1.517E+00	0.	0.	0.	1.500E+02	2.023E-01	0.	0.	1.399E+02	1.375E+02	1.363E+02
2	6	1.286E+00	0.	0.	0.	1.500E+02	2.015E-01	0.	0.	1.343E+02	1.369E+02	1.358E+02
2	7	1.112E+00	0.	0.	0.	1.500E+02	2.007E-01	0.	0.	1.387E+02	1.364E+02	1.353E+02
2	8	9.800E-01	0.	0.	0.	1.500E+02	1.999E-01	0.	0.	1.382E+02	1.358E+02	1.347E+02
2	9	8.762E-01	0.	0.	0.	1.500E+02	1.991E-01	0.	0.	1.376E+02	1.353E+02	1.342E+02

2	10	7.925E-11	0.	0.	1.500E+02	1.984E-01	0.	0.	1.371E+02	7.348E+02	1.337E+02
2	11	7.229E-11	0.	0.	1.500E+02	1.977E-01	0.	0.	1.366E+02	7.343E+02	1.332E+02
2	12	6.646E-11	0.	0.	1.500E+02	1.969E-01	0.	0.	1.361E+02	7.338E+02	1.327E+02
2	13	6.203E-11	0.	0.	1.500E+02	1.962E-01	0.	0.	1.356E+02	7.333E+02	1.322E+02
2	14	6.084E-11	0.	0.	1.500E+02	1.954E-01	0.	0.	1.350E+02	7.327E+02	1.317E+02
2	15	6.488E-11	0.	0.	1.500E+02	1.945E-01	0.	0.	1.344E+02	7.321E+02	1.311E+02
2	16	7.103E-11	0.	0.	1.500E+02	1.931E-01	0.	0.	1.334E+02	7.311E+02	1.302E+02
2	17	6.663E-11	0.	0.	1.500E+02	1.912E-01	0.	0.	1.319E+02	7.299E+02	1.291E+02
2	18	1.270E-11	0.	0.	1.500E+02	1.884E-01	0.	0.	1.297E+02	7.284E+02	1.279E+02
2	19	-1.606E+00	0.	0.	1.500E+02	1.850E-01	0.	0.	1.269E+02	7.265E+02	1.264E+02
2	20	-5.379E+00	0.	0.	1.500E+02	1.809E-01	0.	0.	1.236E+02	7.245E+02	1.250E+02
2	21	-1.191E-01	0.	0.	1.500E+02	1.765E-01	0.	0.	1.200E+02	7.229E+02	1.240E+02
2	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

TIME= 8.62515E+02 TOTAL TIME= 1.66117E+05 NCYCLE= 1000 NIT= 1

ENDTIME 1.06E+03 DSUM=-9.50E-05

CFLI	T	J	U	V	PHT	T	WNP1	GAM	U	XI(I,J,2)	XI(I,J,3)	XI(I,J,4)
1	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	3	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	4	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	5	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	16	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	17	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	19	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	20	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	21	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

1 TIME = 0. TOTAL TIME = 3.30579E+05 NUCLE = 1 NIT = 1

ENDTIME 1.07E+03 DZM = -9.5E-05

RW = 7.00E+00 RH = 2.00E-01 RNOW = 1.0E+00 RHOR = 2.65E+00 AKH = 0.00E-03
 AKAPD = 4.36E-03 AKAP = 6.20E-03 RFTA = 2.07E-04 BLIAR = 2.50E-05 DT = 8.62E-01 UTR = 1.16E+00
 DX = 2.0E+01 DY = 1.0E+01 DZ = 8.33E+00 DAR = 1.00E-02 DYR = 1.00E-01 DZR = 1.20E-01
 DXSOR = 0.2E+02 DYSOR = 1.00E+02 DXSQR = 1.0E+03 DYSQR = 1.00E+02 DZSQR = 1.44E+02
 FIGFIF = 5.33E-01 FIFILL = 2.10E-03 GRAV = 9.80E+02 INAR = 20 JRAK = 20 KRAK = 4
 RNDA = 1.00E-02 RIA = 0. NCYF = 1000 NCYP = 0 NCYT = 1000
 THICK = 5.00E+01 TIK = 1.50E+02 TJOITH = 5.07E-01 VIN = 0. HEIGHT = 1.00E+10
 WD = 0. WC = 5.0E+01 XT1 = 3.00E+02 -N1 = 23 NJ = 23 NK = 10

CELL	I	J	U	V	PHI	T	WNP1	GAM	D	XI(I,J,2)	XI(I,J,3)	XI(I,J,4)
1	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	3	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	4	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	5	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	6	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	7	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	8	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	9	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	10	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	11	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	14	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	16	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	17	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	19	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	20	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	21	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	2	2.260E+01	0.	0.	0.	1.500E+02	2.137E-01	0.	0.	1.472E+02	1.459E+02	1.452E+02
2	3	1.985E+00	0.	0.	0.	1.500E+02	2.135E-01	0.	0.	1.471E+02	1.457E+02	1.451E+02
2	4	1.732E+00	0.	0.	0.	1.500E+02	2.132E-01	0.	0.	1.468E+02	1.455E+02	1.448E+02
2	5	1.508E+00	0.	0.	0.	1.500E+02	2.127E-01	0.	0.	1.466E+02	1.452E+02	1.445E+02
2	6	1.315E+00	0.	0.	0.	1.500E+02	2.123E-01	0.	0.	1.463E+02	1.449E+02	1.442E+02
2	7	1.156E+00	0.	0.	0.	1.500E+02	2.118E-01	0.	0.	1.460E+02	1.445E+02	1.439E+02
2	8	1.024E+00	0.	0.	0.	1.500E+02	2.113E-01	0.	0.	1.456E+02	1.441E+02	1.433E+02
2	9	9.177E-01	0.	0.	0.	1.500E+02	2.107E-01	0.	0.	1.453E+02	1.437E+02	1.429E+02

[illegible]

ENDTIME 1.11E+03 NSUM=-1.73E-04

[illegible]

2	1	0.	0.	0.	0.	0.	0.	0.	0.	0.	
2	2	2.024F+00	0.	9.591F+00	1.500E+02	2.138E-01	0.	0.	1.472E+02	7.459E+02	1.452E+02
2	3	1.480F+00	0.	8.436E+00	1.500E+02	2.136E-01	0.	0.	1.471E+02	7.457E+02	1.451E+02
2	4	1.726F+00	0.	8.037F+00	1.500E+02	2.133E-01	0.	0.	1.468E+02	7.455E+02	1.448E+02
2	5	1.573F+00	0.	7.248F+00	1.500E+02	2.129E-01	0.	0.	1.466E+02	7.452E+02	1.445E+02
2	6	1.427F+00	0.	6.508F+00	1.500E+02	2.125E-01	0.	0.	1.463E+02	7.449E+02	1.442E+02
2	7	1.293F+00	0.	5.838E+00	1.500E+02	2.120E-01	0.	0.	1.460E+02	7.445E+02	1.438E+02
2	8	1.174F+00	0.	5.248F+00	1.500E+02	2.116E-01	0.	0.	1.456E+02	7.441E+02	1.433E+02
2	9	1.071F+00	0.	4.743F+00	1.500E+02	2.110E-01	0.	0.	1.453E+02	7.437E+02	1.429E+02
2	10	9.853F-01	0.	4.331F+00	1.500E+02	2.105E-01	0.	0.	1.450E+02	7.433E+02	1.425E+02
2	11	9.201F-01	0.	4.019F+00	1.500E+02	2.100E-01	0.	0.	1.446E+02	7.429E+02	1.420E+02
2	12	8.761F-01	0.	3.814F+00	1.500E+02	2.094E-01	0.	0.	1.442E+02	7.424E+02	1.415E+02
2	13	8.526F-01	0.	3.706F+00	1.500E+02	2.087E-01	0.	0.	1.438E+02	7.419E+02	1.409E+02
2	14	8.591F-01	0.	3.730F+00	1.500E+02	2.080E-01	0.	0.	1.433E+02	7.413E+02	1.403E+02
2	15	9.487F-01	0.	4.114F+00	1.500E+02	2.071E-01	0.	0.	1.427E+02	7.405E+02	1.395E+02
2	16	1.146F+00	0.	4.907F+00	1.500E+02	2.056E-01	0.	0.	1.416E+02	7.395E+02	1.385E+02
2	17	1.144F+00	0.	4.583F+00	1.500E+02	2.031E-01	0.	0.	1.398E+02	7.379E+02	1.371E+02
2	18	2.325F-01	0.	9.547F-01	1.500E+02	1.946E-01	0.	0.	1.371E+02	7.359E+02	1.354E+02
2	19	-2.263F+00	0.	-7.774F+00	1.500E+02	1.952E-01	0.	0.	1.337E+02	7.335E+02	1.335E+02
2	20	-6.739F+00	0.	-2.421F+01	1.500E+02	1.903E-01	0.	0.	1.296E+02	7.310E+02	1.316E+02
2	21	-1.358F+01	0.	-6.717F+01	1.500E+02	1.848E-01	0.	0.	1.252E+02	7.289E+02	1.302E+02
2	22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1	0.	0.	0.	0.	0.	0.	0.	1.528E+02	7.541E+02	1.548E+02
3	2	2.234F+00	-1.737F-02	-9.591F+00	1.503E+02	2.138E-01	3.506E+02	4.165E-10	1.528E+02	7.541E+02	1.548E+02
3	3	2.001F+00	-3.863F-03	-8.836F+00	1.504E+02	2.136E-01	3.509E+02	6.797E-10	1.529E+02	7.543E+02	1.549E+02
3	4	1.779F+00	3.136F-02	-8.037F+00	1.504E+02	2.133E-01	3.514E+02	1.173E-09	1.532E+02	7.545E+02	1.552E+02
3	5	1.574F+00	8.805F-02	-7.248F+00	1.505E+02	2.129E-01	3.520E+02	1.893E-09	1.534E+02	7.548E+02	1.555E+02
3	6	1.395F+00	1.549F-01	-6.508F+00	1.506E+02	2.125E-01	3.527E+02	2.800E-09	1.537E+02	7.551E+02	1.558E+02
3	7	1.231F+00	2.302F-01	-5.838F+00	1.507E+02	2.120E-01	3.535E+02	3.803E-09	1.540E+02	7.555E+02	1.562E+02

ACKNOWLEDGMENTS

We are grateful to T. L. Cook for his valuable contributions to the computer graphics.

BIBLIOGRAPHY

E. S. Robinson, R. M. Potter, B. B. McInteer, J. C. Rowley, D. E. Armstrong, R. L. Mills, and M. C. Smith, A. "A Preliminary Study of the Nuclear Subterranean," Los Alamos Scientific Laboratory report LA-4547 (April 1971).

F. H. Harlow and W. E. Pracht, "A Theoretical Study of Geothermal Energy Extraction," J. Geophys. Res. 77, No. 35 (December 10, 1972).

R. D. McFarland, "Geothermal Reservoir Models--Crack Plane Model," Los Alamos Scientific Laboratory report LA-5947-MS (April 1975).

R. D. McFarland and H. D. Murphy, "Extracting Energy from Hydraulically Fractured Geothermal Reservoirs," Proc. 11th Intersociety Energy Conversion Engineering Conf., State Line, Nevada., 1, 828-835, September 1976.