



University of Kentucky
UKnowledge

KWRRRI Research Reports

Kentucky Water Resources Research Institute

4-1972

Digital Computer Modeling of Limestone Groundwater Systems

Digital Object Identifier: <https://doi.org/10.13023/kwrrri.rr.50>

John Thrailkill
University of Kentucky

David P. Beiter
University of Kentucky

Right click to open a feedback form in a new tab to let us know how this document benefits you.

Follow this and additional works at: https://uknowledge.uky.edu/kwrrri_reports

 Part of the [Computer Sciences Commons](#), [Earth Sciences Commons](#), and the [Environmental Sciences Commons](#)

Repository Citation

Thrailkill, John and Beiter, David P, "Digital Computer Modeling of Limestone Groundwater Systems" (1972). *KWRRRI Research Reports*. 145.
https://uknowledge.uky.edu/kwrrri_reports/145

This Report is brought to you for free and open access by the Kentucky Water Resources Research Institute at UKnowledge. It has been accepted for inclusion in KWRRRI Research Reports by an authorized administrator of UKnowledge. For more information, please contact UKnowledge@lsv.uky.edu.

DIGITAL COMPUTER MODELING OF LIMESTONE
GROUNDWATER SYSTEMS

Dr. John Thrailkill
Principal Investigator

Graduate Student Assistant: David P. Beiter

Project Number A-030-KY (Completion Report)
Agreement Number 14-31-0001-3217 (FY 1971)
Period of Project - July, 1970 - August, 1971

University of Kentucky Water Resources Institute
Lexington, Kentucky

The work on which this report is based was supported in part by funds provided by the Office of Water Resources Research, United States Department of the Interior, as authorized under the Water Resources Research Act of 1964.

April, 1972

ABSTRACT

Because limestone groundwater flows mainly in discrete openings, limestone aquifers are fundamentally different from aquifers in granular rocks. A digital computer program which simulates flow in a limestone aquifer as a pipe network was written and compared with the Sinkhole Plain aquifer of west-central Kentucky.

A reasonably good fit between observed parameters of the aquifer and those calculated were obtained under assumed conditions of both laminar and turbulent flow in the aquifer. The indicated gross permeability of the aquifer is 5600 meinzers with an assumed aquifer thickness of 100 feet. The location and discharge of springs along the streams bounding the aquifer are predicted.

With further refinements to the computational routines, additional features of the aquifer can be modelled, and more refined predictions can be made of water budget parameters, location of flow paths, and development of the aquifer.

KEY WORDS: limestone aquifer/ground-water basin/digital computer models/ flownets.

TABLE OF CONTENTS

	Page
ABSTRACT	ii
ACKNOWLEDGMENTS	iii
LIST OF TABLES	v
LIST OF FIGURES	vi
 CHAPTERS	
I INTRODUCTION	1
Significance of Research	1
Project Objectives	2
II DESCRIPTION OF PROGRAM KLAM (Kentucky Limestone Aquifer Model)	3
Geometry	4
General Computational Procedure	4
Summary of Program	5
III MODEL 1 - SINKHOLE PLAIN AQUIFER-LAMINAR FLOW	6
General	6
Parameters	8
Results	8
IV MODEL 2 - SINKHOLE PLAIN AQUIFER- TURBULENT FLOW	16
General and Parameters	16
Results	16
V MODEL 3 - NORTH-CENTRAL SINKHOLE PLAIN AQUIFER-LAMINAR FLOW	20
General and Parameters	20
Results	23
VI CONCLUDING SUMMARY	25
PUBLICATIONS AND TRAINING ACCOMPLISHED	26
REFERENCES	27
APPENDIX - KENTUCKY LIMESTONE AQUIFER MODEL	

LIST OF TABLES

Table		Page
1	Comparison of Observed and Calculated Heads for Model 1 (feet)	13
2	Comparison of Observed Flows in the Aquifer and Model 1 Calculated Branch Flows (ft ³ /min)	15
3	Comparison of Observed and Model 1 Calculated Discharges on the Aquifer Boundary (ft ³ /min)	15
4	Comparison of Observed and Calculated Heads for Model 2 (feet)	18
5	Comparison of Observed Flows in the Aquifer and Model 2 Calculated Branch Flows (ft ³ /min)	19
6	Comparison of Observed and Model 2 Calculated Discharges On the Aquifer Boundary (ft ³ /min)	19
7	Comparison of Observed and Model 3 Calculated Flows and Discharges (ft ³ /min)	23

LIST OF ILLUSTRATIONS

Figure		Page
1	Indexing Scheme for Network Parameters	3
2	Sinkhole Plain Aquifer - Location Map	7
3	Sinkhole Plain Aquifer - Network and Values of Specified Parameters	9
4	Sinkhole Plain Aquifer - Model 1 Values of Flow in Branches	10
5	Sinkhole Plain Aquifer - Model 1 Values of Recharge	11
6	Sinkhole Plain Aquifer - Model 1 Values of Head	12
7	Sinkhole Plain Aquifer - Model 2 Values of Head	17
8	North-Central Sinkhole Plain Aquifer - Geologic Map	21
9	North-Central Sinkhole Plain Aquifer - Model 3 Recharge	22
10	North-Central Sinkhole Plain Aquifer - Model 3 Values of Head	24

CHAPTER I

INTRODUCTION

Significance of Research

In many areas, water found in openings in the rock (groundwater) is a major water supply. Where the underlying rock is granular, such as the alluvium along major streams, the study, evaluation, and production of this groundwater is facilitated by the existence of a well-developed theory of the granular groundwater body.

Much of Kentucky, as well as large portions of other states, is underlain by limestone. Groundwater in limestone (and similar rocks) behaves quite differently than that in granular rocks. The permeability is localized in discrete openings, rather than in the intergrain pores of granular rocks, and the flow is often turbulent. The distribution of openings is irregular, and the openings themselves have been and are being enlarged by the flow of water within them.

Although flow assumptions borrowed from granular rock theory (e. g., linear relationship between head loss and flow) are widely applied (with varying degrees of success) to limestones, it is evident that concepts and methods which take into account the nature of the limestone aquifer need to be developed.

The importance of limestone groundwater to Kentucky may be judged by the following summary derived from data in Mull, et al (1971).

Twenty-nine of Kentucky's 120 counties obtain some of their public and industrial water from limestone groundwater. Of these, 7 derive between 10 and 50% of their supply from this source, and six obtain more than half their supply from limestone. In this latter group, Hart, Russell, and Scott counties get more than 80% of their public and industrial water supply from this source, and Allen county receives all of its supply from limestone groundwater.

Currently, the probabilities of obtaining a yield sufficient for public or industrial water supply from a well drilled in a limestone area are quite low. An inspection of the data in Mull, et al (1971) shows only about one-fifth of the wells listed have a yield of 100 gallons/minute or greater. Large flows of water do exist, however, as evidenced by several springs discharging more than 1000 gallons/minute. It is evident that the development of a theory of limestone groundwater which would allow the prediction of the location of these large flows would be of considerable economic benefit.

A better understanding of groundwater in limestone would have benefits other than water supply. The prediction of reservoir leakage and of pollution paths is currently a difficult matter in the absence of a well-developed theory of limestone groundwater.

Project Objectives

The original objectives of the project (as described in the original proposal) were:

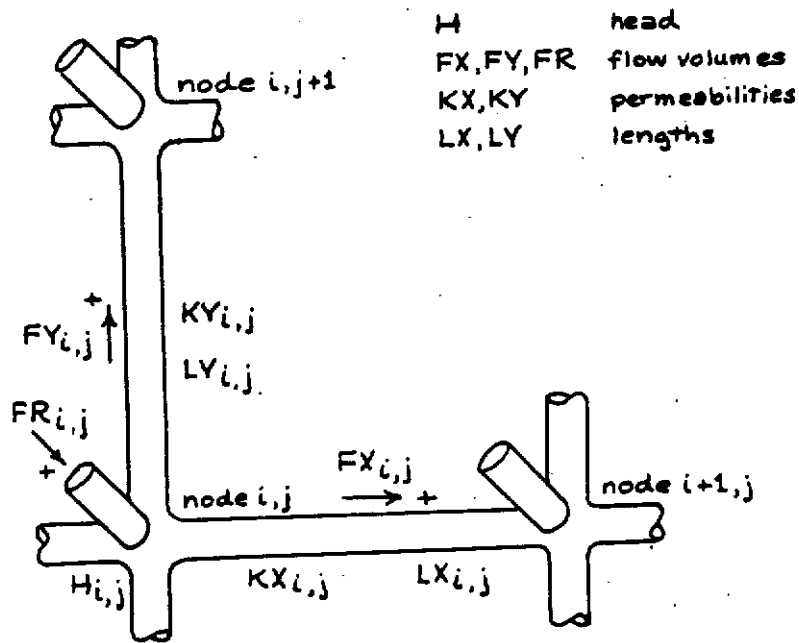
To develop digital computer model techniques to describe limestone groundwater basins in two and three dimensions. Models will be developed for actual limestone aquifer systems with boundary conditions determined by geologic and geochemical data now available or to be acquired as part of the project. Numerical techniques will be used to obtain solutions to the models. Results will be made available on specific systems to guide groundwater surface water development and aid in pollution control. Training of one or two graduate students in the techniques developed will be accomplished.

Midway through the first year of the planned two-year period the project was revised by shortening it to 14 months and reducing the funds to 42% of the original amount. This necessitated a drastic limitation of the project objectives, as discussed in a later section of this report.

CHAPTER II

DESCRIPTION OF PROGRAM KLAM (KENTUCKY LIMESTONE AQUIFER MODEL)

The aquifer is approximated as a 2-dimensional quadrilateral network of pipes. Each branch (pipe segment) has assigned or calculated values of flow volume, flow resistance (permeability), and length. Each node (junction of pipe segments) has assigned or calculated values of head and flow volume introduced from outside the system (recharge flow). The network is numbered as being in the first quadrant. The indexing scheme for nodes and branches is shown in Figure 1, together with the variables and the sign convention for flows.



INDEXING SCHEME FOR NETWORK PARAMETERS

FIGURE 1

Geometry

1. The node flow error (E) at a node is calculated as

$$E_{i,j} = FX_{i,j} + FY_{i,j} - FR_{i,j} - FX_{i-1,j} - FY_{i,j-1}$$

where the symbols are as shown in Figure 1.

2. The branch flow error (E-) in the four branches associated with the node is calculated as

$$EX_{i,j} = \frac{KX_{i,j} \cdot (H_{i,j} - H_{i+1,j})^{\frac{1}{n}}}{LX_{i,j}} - FX_{i,j}$$

for the branch to the right of the node and similarly for the remaining branches. In this expression n is 1 for the laminar flow models and 2 for the turbulent flow models.

3. The flow in each branch (including the recharge flow) about the node is corrected by an amount C- calculated as

$$CX_{i,j} = \frac{-R \cdot (1 - W) \cdot E_{i,j}}{M_{i,j}} + W \cdot EX_{i,j}$$

for the branch to the right of the node and similarly for the remaining branches. R is a relaxation coefficient, W a parameter to allow the relative influence of the two errors to be changed, and M is the number of branches around the node.

4. The head at the node is calculated from the heads at adjacent nodes and the new branch flows by averaging expressions similar to

$$H_{i,j} = \frac{LX_{i,j} \cdot (FX_{i,j})^n}{KX_{i,j}} + H_{i+1,j}$$

5. After the above computations are performed for each node, the entire procedure is iterated until convergence occurs.

Summary of Program

The program is written in Fortran IV and executed on an IBM 360/65 computer. When dimensioned to accommodate a 38 x 27 node network, 118K bytes of storage were required and 100 iterations required about 8 minutes of central processing unit time.

The program consists of a main program and 14 sub-routines.

1. MAIN - reads program parameter card, writes out parameters, and performs iterative computations described above.
2. IN1 - reads data for each node on separate card.
3. IN2 - reads data for all nodes on first card and changes for individual nodes on later cards.
4. IN3 - reads data for each node on cards punched by OUT4.
5. CAL1 - calculates KX and KY if undefined.
6. CAL2 - counts entered value and writes messages.
7. CAL3 - assigns starting values of unspecified flows.
8. CAL4 - assigns starting values of unspecified heads.
9. OUT1 - lists values after specified number of iterations.
10. OUT2 - lists final values in table form.
11. OUT3 - fills arrays for map output of final values.
12. OUT3A - lists final values in map form.
13. OUT3B - writes heading for map output.
14. OUT3C - writes error table for map output.
15. OUT4 - punches cards with final values.

In the initial stages, the program was titled LASP but this was changed to KLAM (Kentucky Limestone Aquifer Model) during the project. A complete listing of the program (slightly expanded for readability) will be found in the Appendix.

CHAPTER III

MODEL 1 - SINKHOLE PLAIN AQUIFER - LAMINAR FLOW

General

The Sinkhole Plain Aquifer was considered to be a continuous aquifer underlying the area shown on Figure 2. Its boundaries were taken to be the streams indicated, which are perennial and of sufficient size to be represented as double lines on 1/24,000 scale maps.

All of the rocks of the area are nearly flat-lying, with a general regional dip to the north of about 10 meters per kilometer. Pennsylvanian rocks crop out near the Green and Barren rivers in the northwest part of the area, otherwise only rocks of Mississippian age outcrop. The Pennsylvanian and upper Mississippian rocks are interbedded sandstones and shales with thin limestones in the Mississippian. The lower Mississippian rocks are almost entirely limestones and dolomites.

North of the south-facing Chester Escarpment, which roughly follows a line connecting the towns of Munfordville, Horse Cave, Cave City, Park City, and Bowling Green, most of the area is underlain by upper Mississippian and Pennsylvanian rocks, with the lower Mississippian limestones and dolomites cropping out only in the bottom of sinkholes and valleys. The portion of the area south of the escarpment, known as the Sinkhole Plain, is underlain by lower Mississippian rocks and is a typical karst, with few surface streams and, in many areas, a very high density of sinkholes. The average altitude is about 220 meters (750 feet) with an average local relief of about 20 meters (65 feet). The relief is fine-textured; a characteristic sinkhole diameter being 100 meters (300 feet).

The eastern part of the plateau north of the Chester escarpment is also a karst, but of a significantly different form. The average altitude of the

plateau tops is about 260 meters (850 feet) and that of the intervening sinkholes is 200 meters (650 feet). The local relief is thus three times that of the Sinkhole Plain. The texture is also much coarser with an average sinkhole diameter of about 1 km.

Parameters

A grid with a standard node interval of 12,500 feet was established (Figure 3). Nodes on the boundary were moved to fall on the bounding streams and the lengths of the branches connecting these nodes to the grid were shortened where necessary. Note that the orientation of the branch does not affect the computations. These boundary nodes were assigned head values taken from 1/24,000 scale topographic maps, and the recharge for these nodes was left unspecified (Figure 3). This model represents boundary geometry and heads prior to the construction and filling of a reservoir on the Barren River in the southermost part of the area.

There is very little surface drainage in the area, and a recharge for the interior nodes was calculated assuming that all non-evapotranspired precipitation (i. e., runoff plus infiltration) enters the aquifer as recharge. In a study of a geologically and climatologically similar area about 120 km to the west (Walker, 1956), the runoff varied from 3.4 inches to 70.7 inches per year over a ten year period. A figure of 30 in/yr was adopted for this model. This is equivalent to a recharge flow of $740 \text{ ft}^3/\text{min}$ at each node.

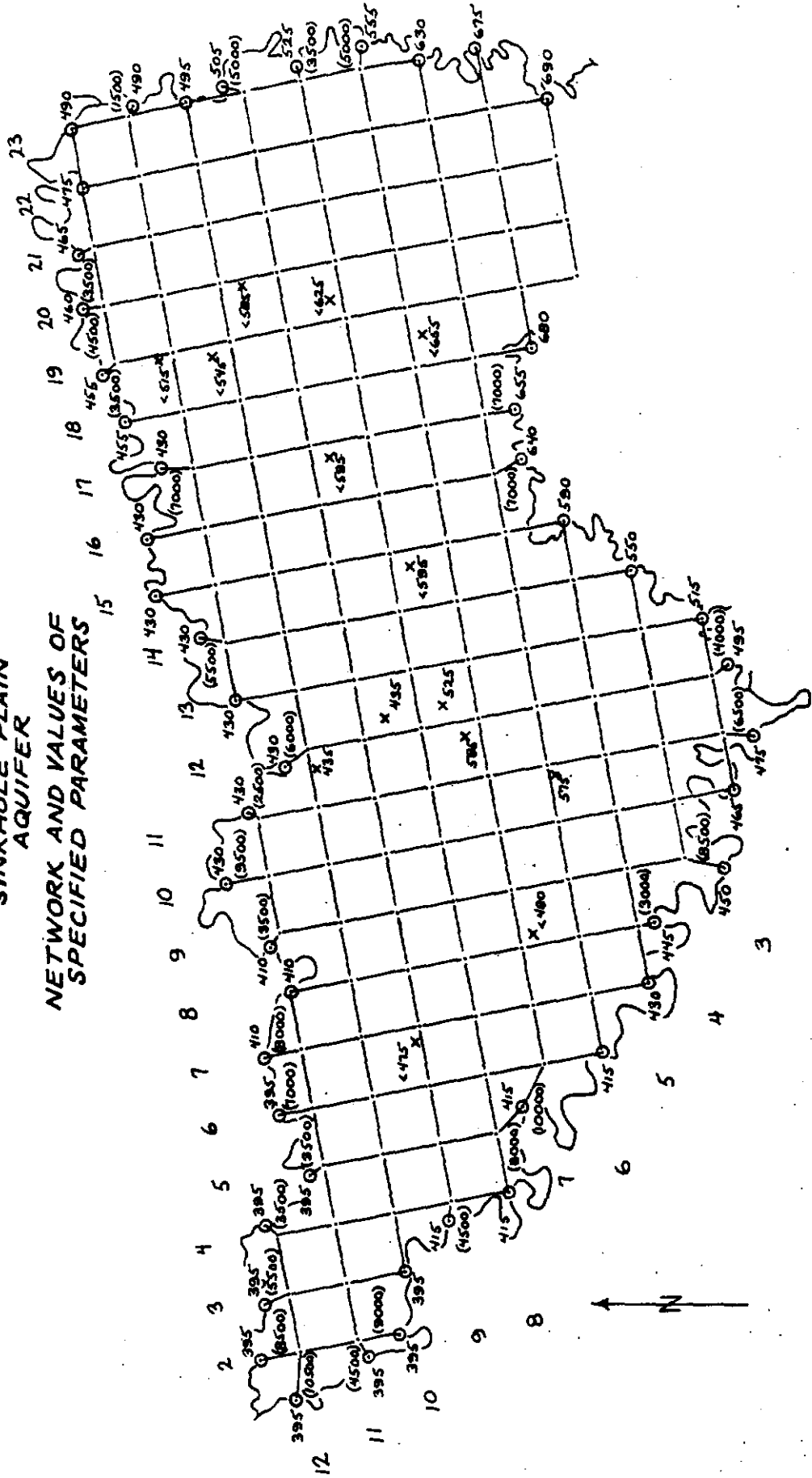
During execution, the permeability (inverse flow resistance) of the branches was adjusted to obtain a fit to observed heads at interior points in the area (Figure 3).

Results

A solution judged to be satisfactory was obtained after 560 iterations. The trial and error method used to determine a permeability is not very efficient, and convergence could have been obtained in only 100 or so iterations if the permeability had been held constant. Final values for the branch flow, recharge flow, and head are shown on Figures 4, 5, and 6 respectively.

SINKHOLE PLAIN
AQUIFER

NETWORK AND VALUES OF
SPECIFIED PARAMETERS



450 head specified at node
 450x head used to adjust mode/
 (5000) length in feet of shortened branch

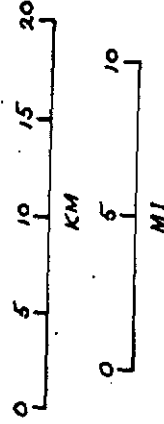
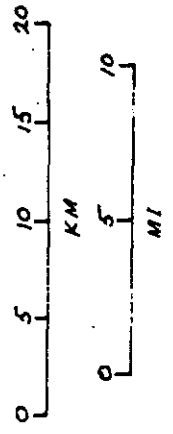
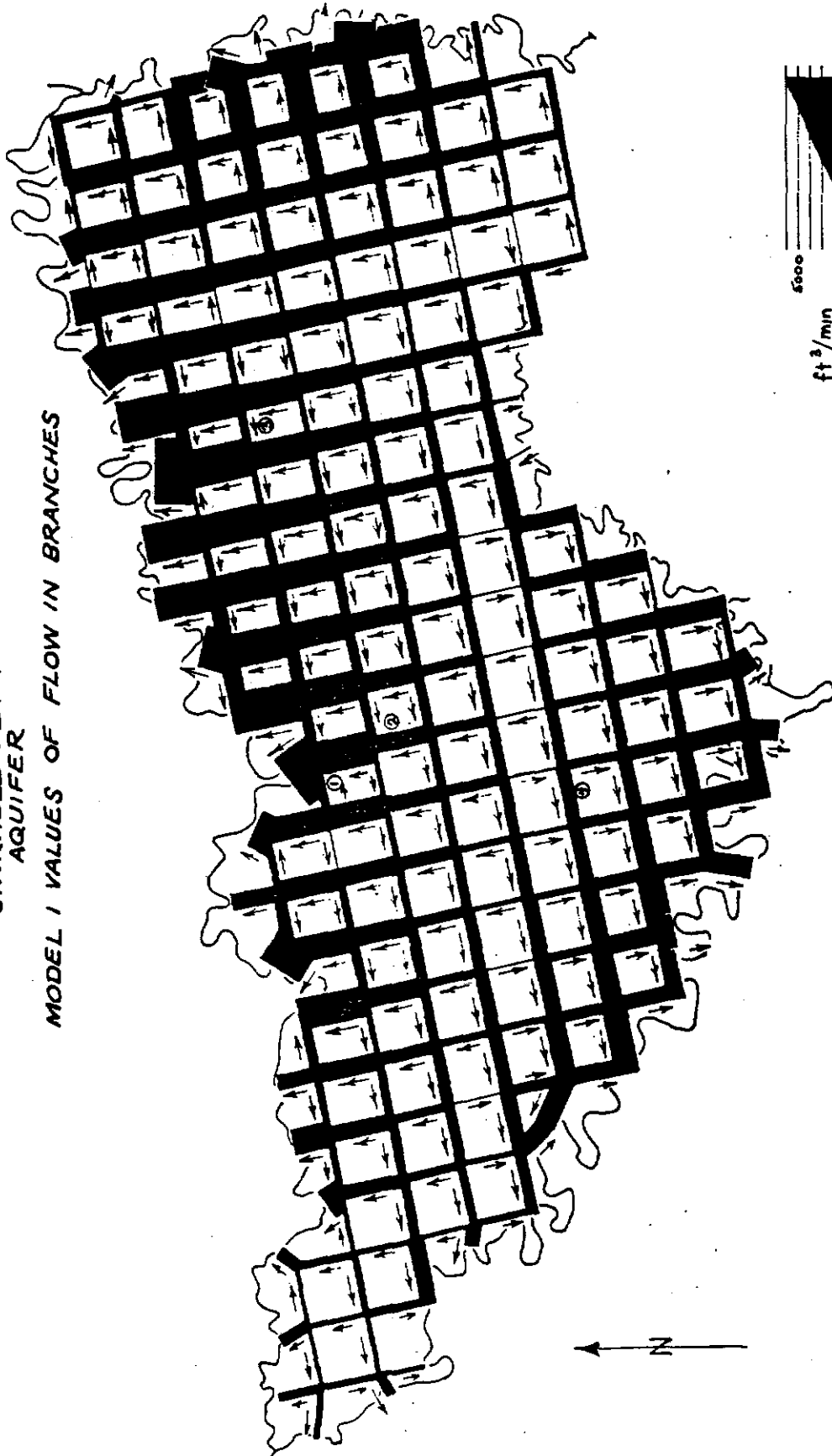


FIGURE 3

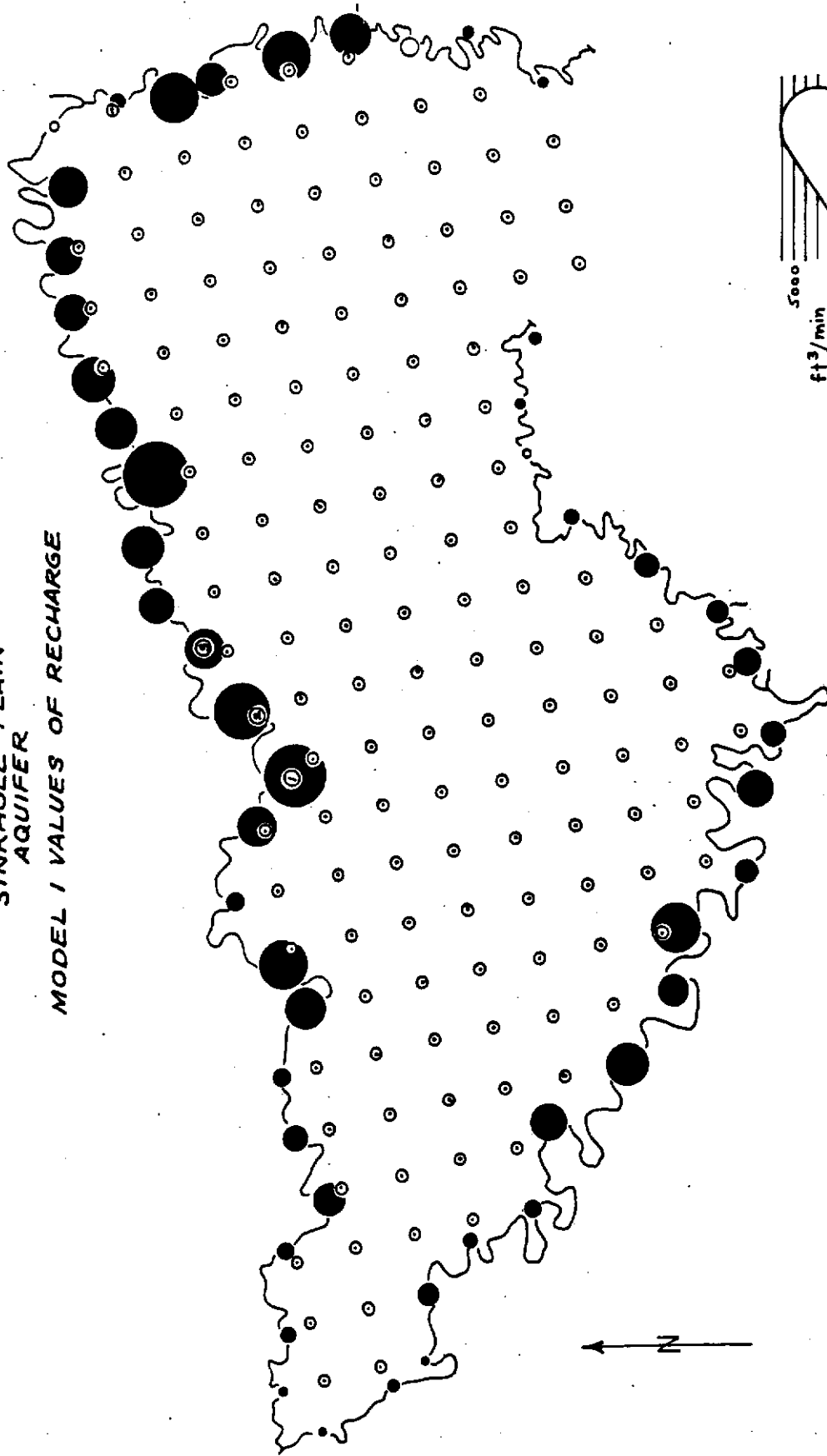
SINKHOLE PLAIN
AQUIFER
MODEL 1 VALUES OF FLOW IN BRANCHES



- ① Cedar Sink 3500 ft³/min (Brown, 1966)
- ② Mill Hole 2700 ft³/min (Brown, 1966)
- ③ Hidden River 4700 ft³/min (Brown, 1966)
- ④ Sink 340 ft³/min (Threlkell, 1970)

FIGURE 4

**SINKHOLE PLAIN
AQUIFER
MODEL 1 VALUES OF RECHARGE**

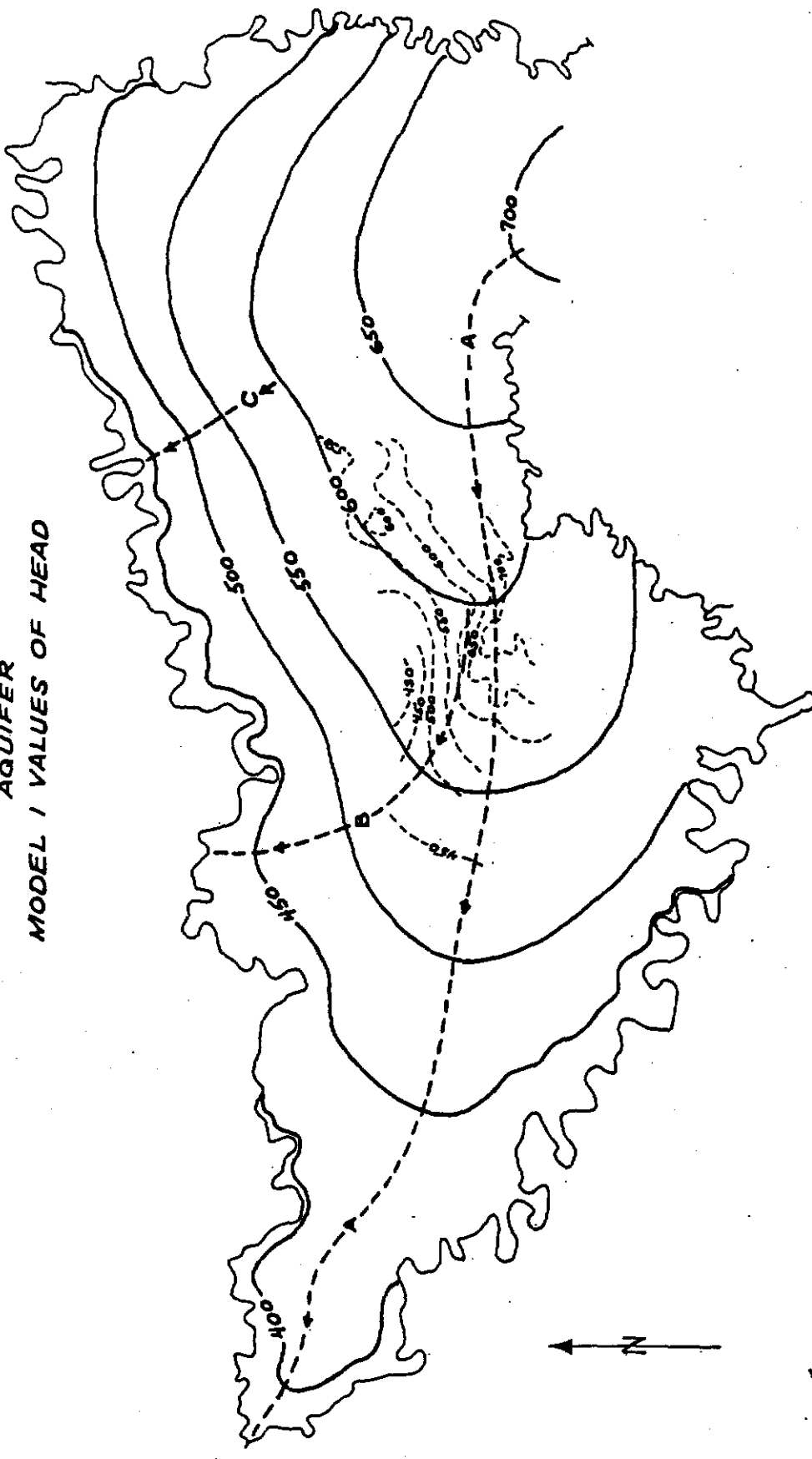


- ① Turnhole / Sand Cave 8000 ft³/min (Brown, 1966)
- ② River Styx / Echo River 95 ft³/min (Brown, 1966)
- ③ Ake Spring 2000 ft³/min (Brown, 1966)

- ⊙ recharge - specified
- recharge - calculated
- discharge - calculated

FIGURE 5

SINKHOLE PLAIN
AQUIFER
MODEL 1 VALUES OF HEAD



-500- Model 1 values of head
 -A-B- Model 1 flowlines
 -500- Water Table contours from Cushman (1968)

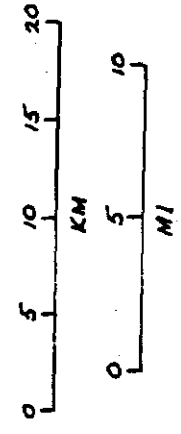


FIGURE 6

The final permeability used was 650,000 ft³/min (the appropriate units are those of discharge), which was judged to provide a satisfactory fit to the observed head at the interior points as shown in Table 1.

TABLE 1
COMPARISON OF OBSERVED AND CALCULATED HEADS
FOR MODEL 1 (feet)

Point	Observed Head (Figure 3)	Head interpolated from Figure 6	Difference (Model 1 - observed)
1	475	465	-
2	480	500	+ 20
3	575	550	- 25
4	435	470	+ 35
5	585	565	- 20
6	435	545	+ 110
7	525	565	+ 40
8	595	600	+ 5
9	595	595	-
10	515	505	-
11	545	555	+ 10
12	655	665	+ 10
13	585	580	-
14	625	630	+ 5

Nine of the fourteen points were sinkholes which apparently did not reach the water table (no water was shown in them on the topographic map) and thus suggest only an upper limit for the head (points 1, 2, 8 through 14 in Table 1).

Assuming active flow occurs in the aquifer to a depth of 100 feet and with the pipe spacing of 12,500 ft, each branch corresponds to a cross-sectional area of the aquifer of $1.25 \times 10^6 \text{ ft}^2$. The pipe flow permeability of $6.5 \times 10^5 \text{ ft}^3/\text{min}$ therefore corresponds to a diffuse flow permeability of .52 ft/min, or 5600 meinzers (1 meinzers is equivalent to $9.3 \times 10^{-5} \text{ ft}/\text{min}$). This is a relatively high permeability, about that of a very well sorted, medium-grained sand (Davis and DeWiest, 1966, p. 164).

Data with which to evaluate the calculated branch flows and recharge (discharge) flows are limited. Flow estimates in streams appearing in four deep sinks or caves have been made, and are compared with the corresponding branch flows in Table 2. The location of these points is shown on Figure 4. Similarly, the calculated discharge is compared with the observed discharge in some large springs on the Green River (Figure 5 and Table 3).

The order-of-magnitude accordance between most observed branch flows and discharges and those calculated as Model 1 (Tables 2 and 3) suggests the model has some validity. The fact that the values calculated depended on the node spacing may indicate that a texture of about that used is present in the aquifer.

A water table map of part of the area (Cushman, 1968) based on well data is shown on Figure 6. Although large differences exist between the observed head and the calculated head, it is interesting to note that the groundwater divide between the Green and Barren Rivers suggested by the observed water table contours nearly coincides with the divide calculated for Model 1 (flow-line A on Figure 6).

Most of the observed discrepancies between Model 1 and observations of the aquifer can probably be explained by the general nature of the Model 1 parameters, such as recharge assumed constant both in time and space, uniform permeability throughout the aquifer, and uniform spacing of flow conduits. The very bad fit between water table contours and calculated heads near the center of the area, for example, can probably best be explained

by much higher permeabilities in this area (which could be used in a refinement of Model 1). It appears, however, that Model 1 provides a reasonably good approximation of the aquifer and that the calculated parameters have some predictive value.

TABLE 2

COMPARISON OF OBSERVED FLOWS IN THE AQUIFER AND MODEL 1
CALCULATED BRANCH FLOWS (ft³/min)

Point (Figure 4)	Apparent direction	Observed flow and reference	Corresponding branch between nodes indicated	Model 1 flow
1	north	3300 (Brown, 1966)	12, 9 and 12, 10	2700 north
2	north	2700 (Brown, 1966)	12, 8 and p2, 9	1600 north
3	north	4700 (Brown, 1966)	18, 9 and 18, 10	2200 north
4	west	340 (Thrallkill, 1970)	10, 6 and 11, 6	1000 west

TABLE 3

COMPARISON OF OBSERVED AND MODEL 1 CALCULATED DISCHARGES
ON THE AQUIFER BOUNDARY (ft³/min)

Point (Figure 5)	Observed discharge and reference	Corresponding node	Model 1 discharge
1	8000 (Brown, 1966)	12, 11	4900
2	95 (Brown, 1966)	13, 11	4700
3	2000 (Brown, 1966)	14, 12	3300

CHAPTER IV

MODEL 2 - SINKHOLE PLAIN AQUIFER TURBULENT FLOW

General and Parameters

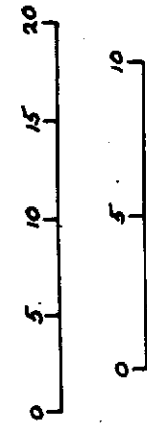
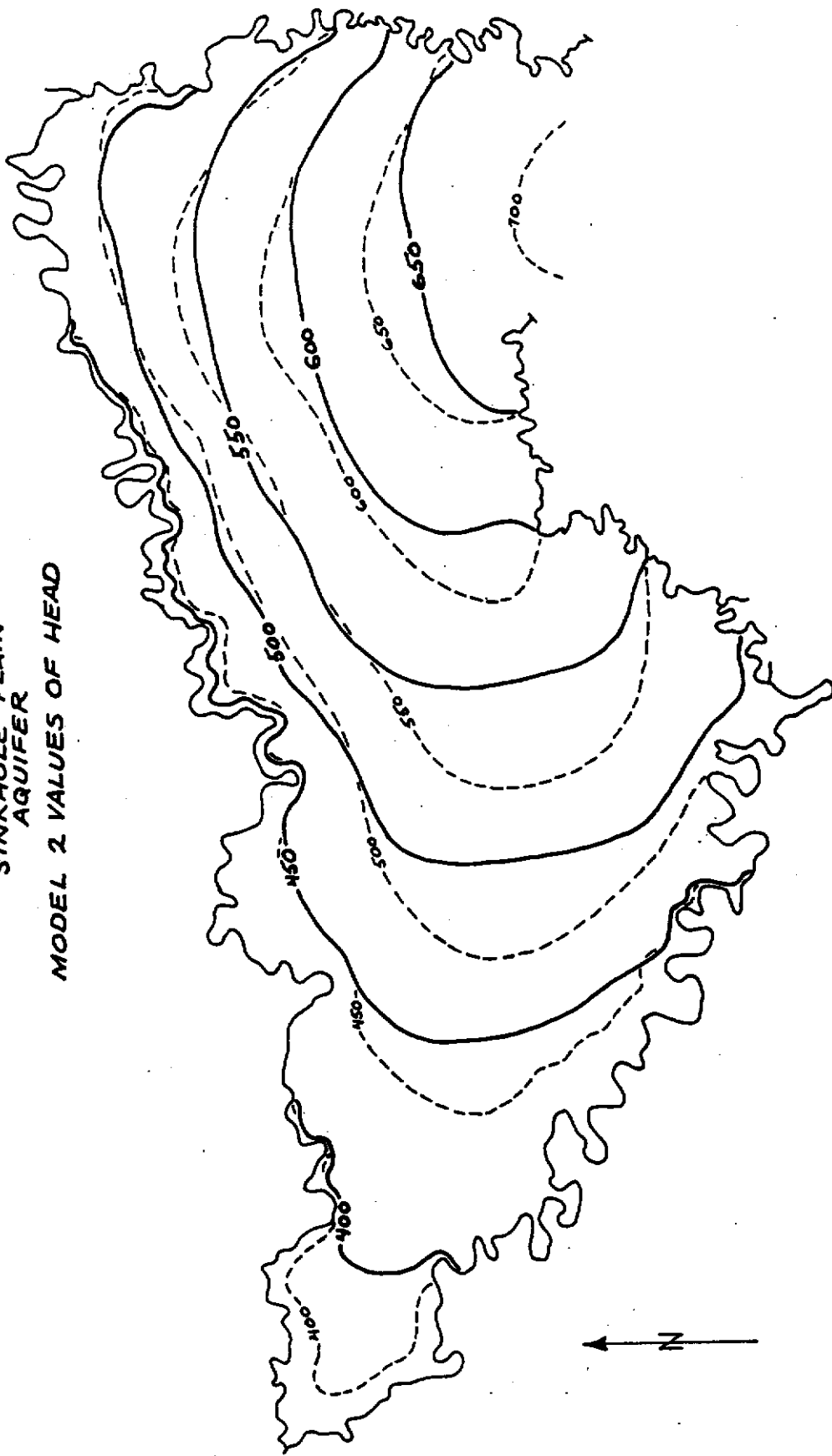
The boundaries and parameters of Model 2 are the same as Model 1 except that turbulent flow in the wholly rough region was specified by assigning the value of 2.0 to the parameter n in steps 2 and 4 of the computational procedure discussed earlier. As with Model 1, the permeability was adjusted during execution to fit observed interior heads.

Results

A solution was obtained after 100 iterations starting with Model 1 values of variables. Head contours (water table elevation) are shown on Figure 7, and comparisons of Model 2, Model 1, and observed head, branch flows, and discharges are given in Tables 4, 5, and 6. The permeability arrived at by the same adjustment procedure used for Model 1 was $2 \times 10^9 \text{ ft}^6/\text{min}^2$, or $(45,000 \text{ ft}^3/\text{min})^2$.

It is apparent that Model 2 is so similar to Model 1 that observed aquifer flows and heads are insufficient to test their relative validity.

SINKHOLE PLAIN
AQUIFER
MODEL 2 VALUES OF HEAD



—500— Model 2 values of head
- - -500- - - Model 1 values of head

FIGURE 7

TABLE 4
COMPARISON OF OBSERVED AND CALCULATED HEADS
FOR MODEL 2 (feet)

Point	Observed head (Figure 3)	Model 2 head interpolated from Figure 7	Difference (Model 2 - observed)	Model 1 difference (from Table 1)
1	475	445	-	-
2	480	470	-	+ 20
3	575	520	- 55	- 25
4	435	480	+ 45	+ 35
5	585	530	- 55	- 20
6	435	530	+ 95	+ 90
7	525	545	+ 20	+ 40
8	595	585	-	+ 5
9	595	590	-	-
10	515	505	-	-
11	545	545	-	+10
12	655	650	-	+ 10
13	585	565	-	-
14	625	610	-	+ 5

TABLE 5

COMPARISON OF OBSERVED FLOWS IN THE AQUIFER AND MODEL 2 CALCULATED BRANCH FLOWS (ft³/min). SEE TABLE 2 FOR REFERENCE FOR OBSERVED FLOW AND BRANCH COORDINATES.

Point (Figure 4)	Apparent direction	Observed flow	Model 2 flow	Model 1 flow (from Table 2)
1	north	3300	2500 north	2700 north
2	north	2700	1600 north	1600 north
3	north	4700	2400 north	2200 north
4	west	340	1700 west	1000 west

TABLE 6

COMPARISON OF OBSERVED AND MODEL 2 CALCULATED DISCHARGES ON THE AQUIFER BOUNDARY (ft³/min). SEE TABLE 3 FOR REFERENCE FOR OBSERVED FLOW AND NODE COORDINATES.

Point (Figure 5)	Observed discharge	Model 2 discharge	Model 1 discharge (from Table 1)
1	8000	4200	4900
2	95	5600	4700
3	2000	3400	3300

CHAPTER V

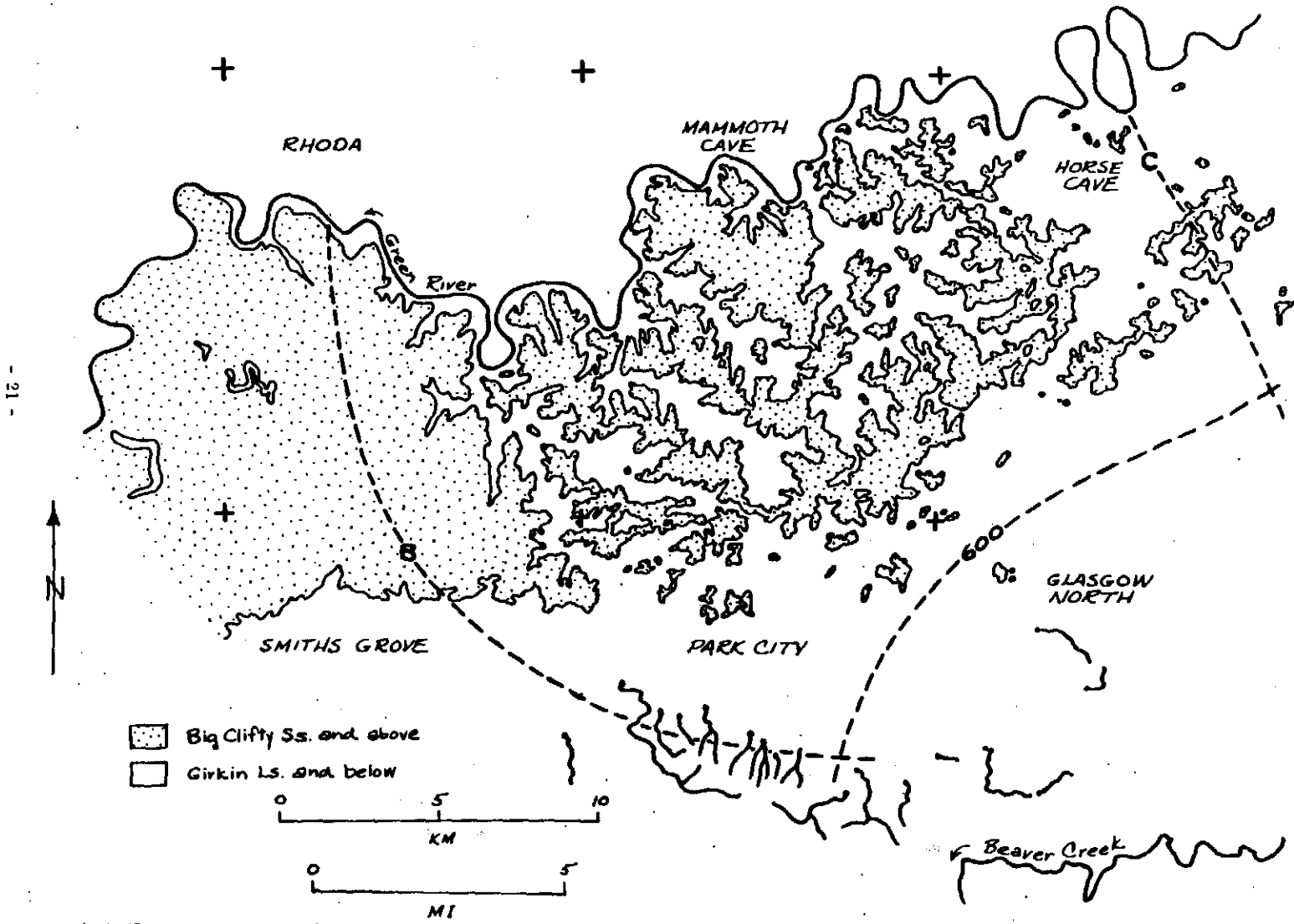
MODEL 3 - NORTH-CENTRAL SINKHOLE PLAIN AQUIFER - LAMINAR FLOW

General and Parameters

In order to investigate flow conditions using a finer grid and with more realistic (and complex) conditions, Model 3 was constructed of the north-central part of the Sinkhole Plain aquifer (Figure 2). Its boundaries were the Green River, the 600 ft. head contour from Model 1, and two Model 1 flow-lines (B and C on Figure 6). A grid spacing of 2500 feet was used.

Other than the finer grid spacing, the principal difference between this model and the previous ones is that it embodies an attempt to represent spatially varying recharge. As outlined earlier, the principal stratigraphic break divides the sedimentary rocks into a thick sequence of limestones and dolomites, of which the Girkin Limestone is the uppermost, overlain by a sequence of thin sandstones, shales, and limestones with the Big Clifty Sandstone at the base directly overlying the Girkin Limestone. The distribution of these two sequences (Gildersleeve, 1963, 1965; Haynes, 1962, 1964a, b, 1966; Klemic, 1963; Richards, 1964) is shown on Figure 8. To at least a good first approximation, no recharge enters the aquifer in areas underlain by the upper sequence (Big Clifty Sandstone and above) due to impermeable shales at the base of the Big Clifty Sandstone. In Model 3, the recharge of 29.6 ft³/min (equivalent to 30 in/year) assigned to each node was introduced at nodes in the lower sequence (open circles on Figure 9) but diverted to the nearest lower sequence node (dotted circles on Figure 9) from nodes located in the outcrop area of the upper sequence (solid circles on Figure 9). Where two or more lower sequence nodes were equi-distant from an upper sequence node, the recharge was divided equally.

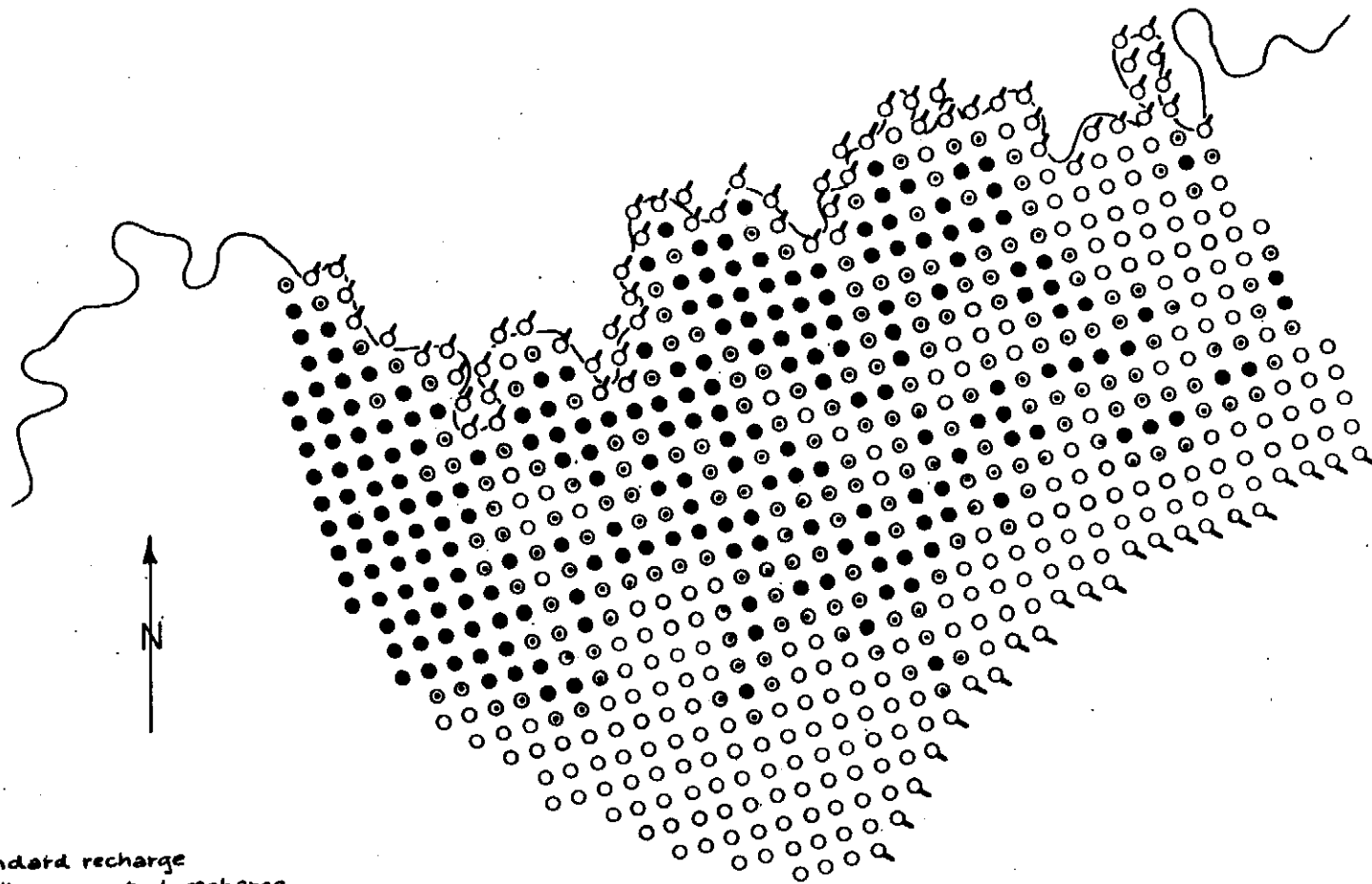
NORTH-CENTRAL
SINKHOLE PLAIN
AQUIFER
GEOLOGIC MAP



- 21 -

FIGURE B

NORTH-CENTRAL
SINKHOLE PLAIN
AQUIFER
MODEL 3 RECHARGE



- 22 -

- standard recharge
- locally augmented recharge
- ⊞ regionally augmented recharge
- zero recharge
- ⊝ unspecified recharge

0 5 10
KM

0 5
MI

FIGURE 9

Aquifer flow entering the model from the southeast was taken to be that calculated in Model 1.

Results

After 1300 iterations, a satisfactorily convergent solution was still not obtained for Model 3 for reasons which will be discussed below. An approximate solution yielded the results shown on Figure 10 and listed in Table 7. A permeability of 130,000 ft³/min was used; the same as that used for Model 1.

TABLE 7
COMPARISON OF OBSERVED AND MODEL 3 CALCULATED
FLOWS AND DISCHARGES (ft³/min)

Point (Figure 10)	Observed flow or discharge (See Figures 4 and 5 for reference)	Corresponding node or branch	Model 3 flow or discharge
1 + 2	discharge: 8000	7, 20 + 8, 20	- 400
3 + 4	discharge: 95	13, 21 + 13, 22	2400
5	discharge: 2000	21, 23	7300
6	flow north: 3300	7, 17 - 7, 18	650 south
7	flow north: 2700	9, 10 - 9, 11	1400 south
8	flow north: 4700	36, 13 - 36, 14	9100 north

NORTH-CENTRAL
SINKHOLE PLAIN
AQUIFER
MODEL 3 VALUES OF HEAD

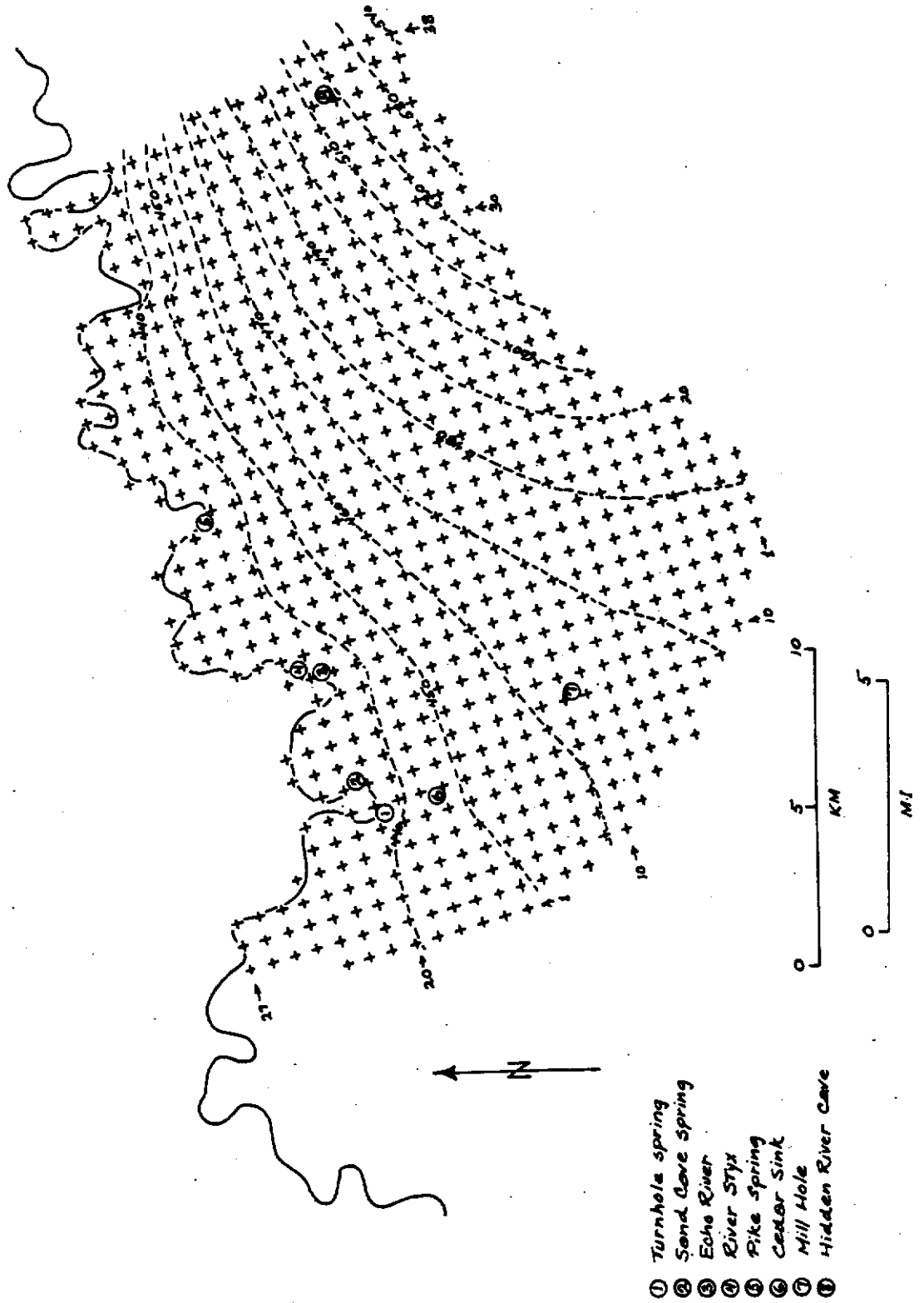


FIGURE 10

CHAPTER VI

CONCLUDING SUMMARY

The results presented in this report are believed to demonstrate the validity of the basic approach investigated. Models 1 and 2 of the entire Sinkhole Plain Aquifer show close correspondence with observed aquifer parameters. It has been shown that Model 1, calculated for laminar flow, and Model 2, calculated for turbulent (wholly rough) flow are sufficiently similar to require detailed observations of the aquifer to discriminate the flow regime in the aquifer, which confirms an earlier conclusion by the writer (Thraillkill, 1968). The existence of large discharges from the aquifer (probably in the form of underwater springs) predicted by the models suggests areas favorable for groundwater exploration and indicates flow paths of groundwater pollution in the area.

As stated in the introduction, the original objectives of the project were significantly reduced in number and scale as a result of time and funding reductions imposed on the project after its initiation. The revised objectives were to (1) write a digital computer program to model the flow in a limestone aquifer, (2) use the program to construct models of a real aquifer of interest, and (3) study the changes in an aquifer with time based on its response to currently available geochemical parameters. Objectives (1) and (2) were accomplished, although there are many routines in the program which require revision to improve their efficiency. The improvements needed in the program and the accomplishment of Objective (3) were prevented by the exhaustion of computer funds, even though a significant amount of non-project funds were also used. It is felt that the major objectives of the project were accomplished and that further work in this direction will provide considerable insight into the nature of the limestone aquifer.

PUBLICATIONS AND TRAINING ACCOMPLISHED

Publications

None as yet (other than progress report)

Training accomplished

One graduate student, David P. Beiter, was supported by and contributed to the project for the academic year 1970-71. Mr. Beiter's research project for the Ph. D. dissertation involves a study of the kinetic factors in the solutional enlargement of flow conduits in limestone.

Some of the results of this project are being utilized in a graduate course in hydrogeology now being offered.

REFERENCES

- Cushman, Robert V. , 1968, Recent Developments in Hydrogeological Investigations in the Karst Area of Central Kentucky. Internat. Assoc. of Hydrogeologists, Mem. 8, p. 236-248.
- Davis, Stanley N. , and DeWiest, Roger J. M. , 1966, Hydrogeology, John Wiley & Sons, New York, 463 p.
- Gildersleeve, Benjamine, 1963, Geology of the Bristow Quadrangle, Kentucky. U. S. G. S. Map GQ-216.
- Gildersleeve, Benjamine, 1965, Geology of the Brownsville Quadrangle, Kentucky. U. S. G. S. Map GQ-411.
- Haynes, Donald D. , 1962, Geology of the Park City Quadrangle, Kentucky. U. S. G. S. Map GQ-183.
- Haynes, Donald D. , 1964a, Geology of the Glasgow North Quadrangle, Kentucky. U. S. G. S. Map GQ-339.
- Haynes, Donald D. , 1964b, Geology of the Mammoth Cave Quadrangle, Kentucky. U. S. G. S. Map GQ-351.
- Haynes, Donald D. , 1966, Geology of the Horse Cave Quadrangle, Kentucky. U. S. G. S. Map GQ-558.
- Klemic, Harry, 1963, Geology of the Rhoda Quadrangle, Kentucky. U. S. G. S. Map GQ-219.
- Mull, Donald S. , Cushman, Robert V. , and Lambert, T. William, 1971, Public and Industrial Water Supplies of Kentucky, 1968-69. Kentucky Geol. Survey, Info. Circ. 20, 107 p.
- Richards, Paul W. , 1964, Geology of the Smiths Grove Quadrangle, Kentucky. U. S. G. S. Map GQ-357.
- Thrailkill, John, 1968, Chemical and Hydrologic Factors in the Excavation of Limestone Caves. Geol. Soc. Amer. Bull. , V. 79, p. 19-46.
- Walker, Eugene H. , 1956, Ground-Water Resources of the Hopkinsville Quadrangle, Kentucky. U. S. G. S. Wat. Sup. Paper 1328, 98 p.

APPENDIX

KENTUCKY LIMESTONE AQUIFER MODEL

C	44	A 1 IN THIS COLUMN PUNCHES FINAL VALUES WITH	MAIN	0410
C		SUBROUTINE OUT4	MAIN	0420
C			MAIN	0430
C	C=C		MAIN	0440
C			MAIN	0450
C	51	A DIGIT (N) IN THIS COLUMN CAUSES OUTPUT WITH OUT1	MAIN	0460
C		EVERY 2**N AND (2**N)+1 ITERATIONS	MAIN	0470
C	52	MULTIPLIER (LEAVE BLANK FOR 1) OF HEAD ASSIGNED TO	MAIN	0480
C		RECHARGE NODES IN SUBROUTINE CAL3 (MFACT)	MAIN	0490
C	53	A 1 IN THIS COLUMN CONTINUES EXECUTION IF NUMBER	MAIN	0500
C		OF ENTERED VARIABLES IS NOT EQUAL TO REQUIRED NUMBER	MAIN	0510
C	54	A DIGIT (Q1) IN THIS COLUMN IS AMOUNT IN TENTHS	MAIN	0520
C		OF HEAD ERROR USED FOR FLOW ADJUSTMENT	MAIN	0530
C	55	A 1 IN THIS COLUMN MAKES Q1 AMOUNT IN HUNDREDTHS,	MAIN	0540
C		A 2 IN THOUSANDTHS, AND A 3 IN TEN-THOUSANDTHS	MAIN	0550
C			MAIN	0560
C	61 THRU 69	A 1 IN THESE COLUMNS WILL CAUSE DELETION OF MAPS	MAIN	0570
C		WRITTEN UNDER SUBROUTINE OUT3 AS FOLLOWS:	MAIN	0580
C			MAIN	0590
C		61 MAP 1 (ENTERED HEAD - BH)	MAIN	0600
C		62 MAP 1 (FINAL HEAD - H)	MAIN	0610
C		63 MAP 3 (ENTERED RECHARGE - BFR)	MAIN	0620
C		64 MAP 4 (FINAL RECHARGE - FR)	MAIN	0630
C		65 MAP 5 (ENTERED BRANCH FLOW - BFX AND BFY)	MAIN	0640
C		66 MAP 6 (FINAL BRANCH FLOW - FX AND FY)	MAIN	0650
C		67 MAP 7 (ENTERED PERMEABILITY - BKX AND BKY)	MAIN	0660
C		68 MAP 8 (FINAL PERMEABILITY - KX AND KY)	MAIN	0670
C		69 MAP 9 (BRANCH LENGTH - BLX AND BLY)	MAIN	0680
C			MAIN	0690
C	C=C		MAIN	0700
C			MAIN	0710
C	SEE INPUT SUBROUTINE FOR REMAINING DATA FORMAT		MAIN	0720
C			MAIN	0730
C	DECLARE COMMON VARIABLES		MAIN	0740
C			MAIN	0750
C	COMMON	H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	MAIN	0760
C	2	BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	MAIN	0770
C	3	KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	MAIN	0780
C	4	I, J, IMAX, JMAX, ITER, CINTER, ERINF, IMAXM1, JMAXM1, XP	MAIN	0790
C		REAL KX, KY	MAIN	0800
C		INITIALIZED	MAIN	0810

C		MAIN	0820
C	DECLARE MAIN VARIABLES	MAIN	0830
C		MAIN	0840
	INTEGER HEADIV, FREBR(5), QI	MAIN	0850
C		MAIN	0860
C	READ PARAMETER CARD	MAIN	0870
C		MAIN	0880
	READ (5,531) IMAX, JMAX, FERROR, HERROR, XP, ORLX, MAXIT, IO1, IO4, IO6,	MAIN	0890
	2 IO2, IO3, IO5, IO7, MML, MFAC, MGO, QI, JQ, ID1, ID2, ID3, ID4,	MAIN	0900
	3 ID5, ID6, ID7, ID8, ID9	MAIN	0910
	531 FORMAT (2I5, 4F5.0, I5, 3I1, 2X, 4I1, 6X, 5I1, 5X, 9I1)	MAIN	0920
	IF (MFAC.EQ.0) MFAC=1	MAIN	0930
C		MAIN	0940
C	WRITE OUT PARAMETERS	MAIN	0950
C		MAIN	0960
	WRITE (6,523)	MAIN	0970
	523 FORMAT ('PARAMETERS USED IN EXECUTION ARE:')	MAIN	0980
	WRITE (6,524) IMAX	MAIN	0990
	524 FORMAT ('0 MAXIMUM VALUE OF I (IMAX) IS ', I5)	MAIN	1000
	WRITE (6,525) JMAX	MAIN	1010
	525 FORMAT ('0 MAXIMUM VALUE OF J (JMAX) IS ', I5)	MAIN	1020
	WRITE (6,526) FERROR	MAIN	1030
	526 FORMAT ('0 MAXIMUM ERROR IN FLOW AT ANY NODE (FERROR) IS ', G10	MAIN	1040
	2.3)	MAIN	1050
	WRITE (6,561) HERROR	MAIN	1060
	561 FORMAT ('0 MAXIMUM ERROR IN HEAD IN ANY BRANCH (HERROR) IS ', G	MAIN	1070
	210.3)	MAIN	1080
	WRITE (6,585) QI	MAIN	1090
	585 FORMAT ('0 DISTRIBUTION FACTOR FOR HEAD ADJUSTMENT (QI) IS ', I	MAIN	1100
	21)	MAIN	1110
	IF (JQ.NE.0) WRITE (6,586) JQ	MAIN	1120
	586 FORMAT ('0 DISTRIBUTION FACTOR (QI) MULTIPLIED BY 10**-'I1)	MAIN	1130
	WRITE (6,559) XP	MAIN	1140
	559 FORMAT ('0 FLOW EXPONENT (XP) IS ', G10.3)	MAIN	1150
	WRITE (6,527) ORLX	MAIN	1160
	527 FORMAT ('0 RELAXATION COEFFICIENT (ORLX) IS ', G10.3)	MAIN	1170
	WRITE (6,528) MAXIT	MAIN	1180
	528 FORMAT ('0 MAXIMUM NUMBER OF ITERATIONS PERMITTED (MAXIT) IS '	MAIN	1190
	2, I5)	MAIN	1200
	IF (IO1.EQ.1) WRITE (6,571)	MAIN	1210
	571 FORMAT ('0 DATA READ WITH SUBROUTINE IN1')	MAIN	1220

	IF (I04.EQ.1) WRITE (6,574)	MAIN	1230
574	FORMAT ('0 DATA READ WITH SUBROUTINE IN2')	MAIN	1240
	IF (I04.EQ.1) WRITE (6,553) MFAC	MAIN	1250
553	FORMAT ('0 MULTIPLIER OF ',I1,' USED WITH CAL3')	MAIN	1260
	IF (I06.EQ.1) WRITE (6,540)	MAIN	1270
540	FORMAT ('0 INITIAL VALUES ENTERED WITH SUBROUTINE IN3')	MAIN	1280
	IF (I02.EQ.1) WRITE (6,572)	MAIN	1290
572	FORMAT ('0 OUTPUT WITH SUBROUTINE OUT1')	MAIN	1300
	IF (I02.EQ.1) WRITE (6,575) MM1,MM1	MAIN	1310
575	FORMAT ('0 OUTPUT EVERY 2**',I1,' AND (2**',I1,')+1 ITERA TIONS')	MAIN	1320
	IF (I03.EQ.1) WRITE (6,573)	MAIN	1330
573	FORMAT ('0 OUTPUT WITH SUBROUTINE OUT2')	MAIN	1340
	IF (I05.EQ.1) WRITE (6,557)	MAIN	1350
557	FORMAT ('0 OUTPUT WITH SUBROUTINE OUT3. THE FOLLOWING MAPS HAV 2E BEEN DELETED:')	MAIN	1360
	IF (I01.EQ.1) WRITE (6,556)	MAIN	1370
556	FORMAT ('0 MAP 1 (ENTERED HEAD - BH)')	MAIN	1380
	IF (I02.EQ.1) WRITE (6,555)	MAIN	1390
555	FORMAT ('0 MAP 2 (FINAL HEAD - H)')	MAIN	1400
	IF (I03.EQ.1) WRITE (6,554)	MAIN	1410
554	FORMAT ('0 MAP 3 (ENTERED RECHARGE - BFR)')	MAIN	1420
	IF (I04.EQ.1) WRITE (6,522)	MAIN	1430
522	FORMAT ('0 MAP 4 (FINAL RECHARGE - FR)')	MAIN	1440
	IF (I05.EQ.1) WRITE (6,521)	MAIN	1450
521	FORMAT ('0 MAP 5 (ENTERED BRANCH FLOW - BFX AND BFY)')	MAIN	1460
	IF (I06.EQ.1) WRITE (6,517)	MAIN	1470
517	FORMAT ('0 MAP 6 (FINAL BRANCH FLOW - FX AND FY)')	MAIN	1480
	IF (I07.EQ.1) WRITE (6,513)	MAIN	1490
513	FORMAT ('0 MAP 7 (ENTERED PERMEABILITY - BKX AND BKY)')	MAIN	1500
	IF (I08.EQ.1) WRITE (6,576)	MAIN	1510
576	FORMAT ('0 MAP 8 (FINAL PERMEABILITY - KX AND KY)')	MAIN	1520
	IF (I09.EQ.1) WRITE (6,577)	MAIN	1530
577	FORMAT ('0 MAP 9 (BRANCH LENGTH - BLX AND BLY)')	MAIN	1540
	IF (I01.NE.1.AND.I02.NE.1.AND.I03.NE.1.AND.I04.NE.1.AND.I05.NE.1. 2 AND.I06.NE.1.AND.I07.NE.1.AND.I08.NE.1.AND.I09.NE.1) WRITE 3 (6,578)	MAIN	1550
	IF (I01.NE.1.AND.I02.NE.1.AND.I03.NE.1.AND.I04.NE.1.AND.I05.NE.1. 2 AND.I06.NE.1.AND.I07.NE.1.AND.I08.NE.1.AND.I09.NE.1) WRITE 3 (6,578)	MAIN	1560
578	FORMAT ('0 NO MAPS DELETED')	MAIN	1570
	IF (I07.EQ.1) WRITE (6,539)	MAIN	1580
539	FORMAT ('0 FINAL VALUES PUNCHED WITH SUBROUTINE OUT4')	MAIN	1590
		MAIN	1600
		MAIN	1610
		MAIN	1620
		MAIN	1630

C INITIALIZE ALL ARRAYS

C

IMAXM1=IMAX-1
JMAXM1=JMAX-1
DO 529 I=1,IMAX
DO 530 J=1,JMAX
H(I,J)=0.
BH(I,J)=0.
FX(I,J)=0.
BFX(I,J)=0.
FY(I,J)=0.
BFY(I,J)=0.
FR(I,J)=0.
BFR(I,J)=0.
KX(I,J)=0.
BKX(I,J)=0.
KY(I,J)=0.
BKY(I,J)=0.
BLX(I,J)=0.
BLY(I,J)=0.

530 CONTINUE

529 CONTINUE

IF (IO1.EQ.1) CALL IN1

IF (IO4.EQ.1) CALL IN2

C

C TEST FOR CORRECT NUMBER OF ENTERED VALUES

C

CALL CAL2 (MGO)

C

C ASSIGN INITIAL VALUES

C

IF (IO6.EQ.1) CALL IN3

IF (IO6.NE.1) CALL CAL3 (MFAC,KCHEK)

C

C

***** BEGIN ITERATIONS *****

C

IF (JQ.EQ.1) Q1=.1*Q1

IF (JQ.EQ.2) Q1=.01*Q1

IF (JQ.EQ.3) Q1=.001*Q1

QH=.1*Q1
QE=1-QH

MAIN 1640
MAIN 1650
MAIN 1660
MAIN 1670
MAIN 1680
MAIN 1690
MAIN 1700
MAIN 1710
MAIN 1720
MAIN 1730
MAIN 1740
MAIN 1750
MAIN 1760
MAIN 1770
MAIN 1780
MAIN 1790
MAIN 1800
MAIN 1810
MAIN 1820
MAIN 1830
MAIN 1840
MAIN 1850
MAIN 1860
MAIN 1870
MAIN 1880
MAIN 1890
MAIN 1900
MAIN 1910
MAIN 1920
MAIN 1930
MAIN 1940
MAIN 1950
MAIN 1960
MAIN 1970
MAIN 1980
MAIN 1990
MAIN 2000
MAIN 2010
MAIN 2020
MAIN 2030
MAIN 2040

<pre> KD2=0 ITER=0 KK=0 550 ITER=ITER+1 CNTR=0 KNTR=0 C C ***** BEGIN CALCULATIONS FOR EACH NODE ***** C DO 511 I=1,IMAX DO 512 J=1,JMAX C C ***** CALCULATE KX AND KY IF UNDEFINED ***** C IF (KCHK.EQ.1) CALL CAL1 C C ***** BEGIN HEAD CALCULATIONS ***** C MFX1=1 MFY1=1 MFX2=1 MFY2=1 IF (FX(I-1,J).LT.0.) MFX1=-1 AFX1=ABS(FX(I-1,J)) IF (FY(I,J-1).LT.0.) MFY1=-1 AFY1=ABS(FY(I,J-1)) IF (FX(I,J).LT.0.) MFX2=-1 AFX2=ABS(FX(I,J)) IF (FY(I,J).LT.0.) MFY2=-1 AFY2=ABS(FY(I,J)) C C CHECK FOR ENTERED VALUE AND SKIP CALCULATIONS C 514 IF (BH(I,J).NE.0.) GO TO 515 C CALCULATE H IF I NE 1, J NE 1, I NE IMAX, J NE JMAX, I-1 KX NE 0, J-1 KY NE 0, KX NE 0, OR KY NE 0 C IF (I.EQ.1.OR.J.EQ.1.OR.I.EQ.IMAX.OR.J.EQ.JMAX) GO TO 516 IF (KX(I,J).LT.0.0001.OR.KY(I,J).LT.0.0001.OR.KX(I-1,J).LT.0.0001 OR.KY(I,J-1).LT.0.0001) GO TO 516 </pre>	<pre> MAIN 2050 MAIN 2060 MAIN 2070 MAIN 2080 MAIN 2090 MAIN 2100 MAIN 2110 MAIN 2120 MAIN 2130 MAIN 2140 MAIN 2150 MAIN 2160 MAIN 2170 MAIN 2180 MAIN 2190 MAIN 2200 MAIN 2210 MAIN 2220 MAIN 2230 MAIN 2240 MAIN 2250 MAIN 2260 MAIN 2270 MAIN 2280 MAIN 2290 MAIN 2300 MAIN 2310 MAIN 2320 MAIN 2330 MAIN 2340 MAIN 2350 MAIN 2360 MAIN 2370 MAIN 2380 MAIN 2390 MAIN 2400 MAIN 2410 MAIN 2420 MAIN 2430 MAIN 2440 MAIN 2450 </pre>
---	--

	IF ((FX(I-1,J).LT.1.0E-70.AND.FX(I-1,J).GT.-1.0E-70).OR.	MAIN	2460
2	(FY(I ,J-1).LT.1.0E-70.AND.FY(I ,J-1).GT.-1.0E-70).OR.	MAIN	2470
3	(FX(I ,J).LT.1.0E-70.AND.FX(I ,J).GT.-1.0E-70).OR.	MAIN	2480
4	(FY(I ,J).LT.1.0E-70.AND.FY(I ,J).GT.-1.0E-70)) GOTO 516	MAIN	2490
	H(I,J)=(((-(BLX(I-1,J)*MFX1*(AFX1**XP))/KX(I-1,J))+H(I-1,J))	MAIN	2500
2	+(((-(BLY(I,J-1)*MFY1*(AFY1**XP))/KY(I,J-1))+H(I,J-1))	MAIN	2510
3	+(((BLX(I,J)*MFX2*(AFX2**XP))/KX(I,J))+H(I+1,J))	MAIN	2520
4	+(((BLY(I,J)*MFY2*(AFY2**XP))/KY(I,J))+H(I,J+1)))/4.	MAIN	2530
C		MAIN	2540
C	SET BRANCH TERMS EQ 0 IF ALL FOUR BRANCHES NOT PRESENT	MAIN	2550
C		MAIN	2560
	516 IF (I.EQ.1) GO TO 518	MAIN	2570
	IF (KX(I-1,J).LT.0.0001) GO TO 518	MAIN	2580
	IF (FX(I-1,J).LT.1.0E-70.AND.FX(I-1,J).GT.-1.0E-70) GO TO 518	MAIN	2590
	GO TO 519	MAIN	2600
	518 AHEAD=0.0	MAIN	2610
	GO TO 520	MAIN	2620
	519 AHEAD= ((-(BLX(I-1,J)*MFX1*(AFX1**XP))/KX(I-1,J))+H(I-1,J))	MAIN	2630
	520 IF (J.EQ.1) GO TO 552	MAIN	2640
	IF (KY(I,J-1).LT.0.0001) GO TO 552	MAIN	2650
	IF (FY(I ,J-1).LT.1.0E-70.AND.FY(I ,J-1).GT.-1.0E-70) GO TO 552	MAIN	2660
	GO TO 563	MAIN	2670
	552 BHEAD=0.0	MAIN	2680
	GO TO 564	MAIN	2690
	563 BHEAD= ((-(BLY(I,J-1)*MFY1*(AFY1**XP))/KY(I,J-1))+H(I,J-1))	MAIN	2700
	564 IF (I.EQ.IMAX.OR.KX(I,J).LT.0.0001) GO TO 565	MAIN	2710
	IF (FX(I ,J).LT.1.0E-70.AND.FX(I ,J).GT.-1.0E-70) GO TO 565	MAIN	2720
	GO TO 566	MAIN	2730
	565 CHEAD=0.0	MAIN	2740
	GO TO 567	MAIN	2750
	566 CHEAD= ((BLX(I,J)*MFX2*(AFX2**XP))/KX(I,J))+H(I+1,J)	MAIN	2760
	567 IF (J.EQ.JMAX.OR.KY(I,J).LT.0.0001) GO TO 568	MAIN	2770
	IF (FY(I ,J).LT.1.0E-70.AND.FY(I ,J).GT.-1.0E-70) GO TO 568	MAIN	2780
	GO TO 569	MAIN	2790
	568 DHEAD=0.0	MAIN	2800
	GO TO 570	MAIN	2810
	569 DHEAD= ((BLY(I,J)*MFY2*(AFY2**XP))/KY(I,J))+H(I,J+1)	MAIN	2820
C		MAIN	2830
C	COUNT NUMBER OF BRANCHES	MAIN	2840
C		MAIN	2850
	570 HEADIV=4	MAIN	2860

IF (AHEAD.EQ.0.0) HEADIV=HEADIV-1	MAIN	2870
IF (BHEAD.EQ.0.0) HEADIV=HEADIV-1	MAIN	2880
IF (CHEAD.EQ.0.0) HEADIV=HEADIV-1	MAIN	2890
IF (DHEAD.EQ.0.0) HEADIV=HEADIV-1	MAIN	2900
IF (HEADIV.EQ.0) GO TO 515	MAIN	2910
C	MAIN	2920
C SUM BRANCH TERMS AND DIVIDE BY NUMBER OF BRANCHES	MAIN	2930
C	MAIN	2940
H(I,J)=(AHEAD+BHEAD+CHEAD+DHEAD)/HEADIV	MAIN	2950
C	MAIN	2960
C ***** CALCULATE ERROR IN FLOW AT NODE *****	MAIN	2970
C	MAIN	2980
C CALCULATE FLOW ERROR AND COMPARE WITH FERROR	MAIN	2990
C	MAIN	3000
515 CONTINUE	MAIN	3010
IF (I.EQ.1) GO TO 534	MAIN	3020
IF (J.EQ.1) GO TO 535	MAIN	3030
ERINF=FX(I,J)+FY(I,J)-FR(I,J)-FX(I-1,J)-FY(I,J-1)	MAIN	3040
GO TO 536	MAIN	3050
534 IF (J.EQ.1) GO TO 537	MAIN	3060
ERINF=FX(I,J)+FY(I,J)-FR(I,J)-FY(I,J-1)	MAIN	3070
GO TO 536	MAIN	3080
537 ERINF=FX(I,J)+FY(I,J)-FR(I,J)	MAIN	3090
GO TO 536	MAIN	3100
535 ERINF=FX(I,J)+FY(I,J)-FR(I,J)-FX(I-1,J)	MAIN	3110
536 ABER=ABS(ERINF)	MAIN	3120
IF (ABER.GT.FERROR) CNTER=CNTER+1	MAIN	3130
C	MAIN	3140
C CALCULATE HEAD ERRORS IN BRANCHES	MAIN	3150
C	MAIN	3160
MFLR=1	MAIN	3170
MFLU=1	MAIN	3180
MFLI=1	MAIN	3190
MFLD=1	MAIN	3200
AFLR=0.	MAIN	3210
AFLU=0.	MAIN	3220
AFLI=0.	MAIN	3230
AFLD=0.	MAIN	3240
CALFLR=0.	MAIN	3250
CALFLU=0.	MAIN	3260
CALFLI=0.	MAIN	3270

	CALFLD=0.	MAIN	3280
	IF (I.EQ.IMAX) GO TO 581	MAIN	3290
	IF ((KX(I,J)*(H(I,J)-H(I+1,J))/BLX(I,J)).LT.0.) MFLR=-1	MAIN	3300
	AFLR=ABS (KX(I,J)*(H(I,J)-H(I+1,J))/BLX(I,J))	MAIN	3310
581	IF (J.EQ.JMAX) GO TO 582	MAIN	3320
	IF ((KY(I,J)*(H(I,J)-H(I,J+1))/BLY(I,J)).LT.0.) MFLU=-1	MAIN	3330
	AFLU=ABS (KY(I,J)*(H(I,J)-H(I,J+1))/BLY(I,J))	MAIN	3340
582	IF (I.EQ.1) GO TO 583	MAIN	3350
	IF ((KX(I-1,J)*(H(I,J)-H(I-1,J))/BLX(I-1,J)).GT.0.) MFLI=-1	MAIN	3360
	AFLI=ABS (KX(I-1,J)*(H(I,J)-H(I-1,J))/BLX(I-1,J))	MAIN	3370
583	IF (J.EQ.1) GO TO 584	MAIN	3380
	IF ((KY(I,J-1)*(H(I,J)-H(I,J-1))/BLY(I,J-1)).GT.0.) MFLD=-1	MAIN	3390
	AFLD=ABS (KY(I,J-1)*(H(I,J)-H(I,J-1))/BLY(I,J-1))	MAIN	3400
584	IF (AFLR.NE.0.) CALFLR=(AFLR**(1./XP))*MFLR	MAIN	3410
	IF (AFLU.NE.0.) CALFLU=(AFLU**(1./XP))*MFLU	MAIN	3420
	IF (AFLI.NE.0.) CALFLL=(AFLI**(1./XP))*MFLI	MAIN	3430
	IF (AFLD.NE.0.) CALFLD=(AFLD**(1./XP))*MFLD	MAIN	3440
C		MAIN	3450
C	DISTRIBUTE FLOW ERROR	MAIN	3460
C		MAIN	3470
	DO 538 L=1,5	MAIN	3480
538	FREBR(L)=1	MAIN	3490
	IF (I.EQ.IMAX.OR.BFX(I,J).NE.0..OR.BKX(I,J).EQ.12345.) FREBR(1)=0	MAIN	3500
	IF (J.EQ.JMAX.OR.BFY(I,J).NE.0..OR.BKY(I,J).EQ.12345.) FREBR(2)=0	MAIN	3510
	IF (I.EQ.1) GO TO 580	MAIN	3520
	IF (BFX(I-1,J).NE.0..OR.BKX(I-1,J).EQ.12345.) FREBR(3)=0	MAIN	3530
580	IF (J.EQ.1) GO TO 544	MAIN	3540
	IF (BFY(I,J-1).NE.0..OR.BKY(I,J-1).EQ.12345.) FREBR(4)=0	MAIN	3550
544	IF (BFR(I,J).NE.0.) FREBR(5)=0	MAIN	3560
	IF (I.EQ.1) FREBR(3)=0	MAIN	3570
	IF (J.EQ.1) FREBR(4)=0	MAIN	3580
	TOTFRE=0.	MAIN	3590
	DO 545 L=1,5	MAIN	3600
545	TOTFRE=TOTFRE+FREBR(L)	MAIN	3610
	IF (TOTFRE.EQ.0.) GO TO 562	MAIN	3620
	IF (FREBR(1).EQ.1) FX(I,J)=FX(I,J)-ORLX*((QF*ERINF/TOTFRE)	MAIN	3630
	+(QH*(CALFLR-FX(I,J))))	MAIN	3640
2	IF (FREBR(2).EQ.1) FY(I,J)=FY(I,J)-ORLX*((QF*ERINF/TOTFRE)	MAIN	3650
	+(QH*(CALFLU-FY(I,J))))	MAIN	3660
2	IF (FREBR(3).EQ.1) FX(I-1,J)=FX(I-1,J)+ORLX*((QF*ERINF/TOTFRE)	MAIN	3670
	+(QH*(CALFLL-FX(I-1,J))))	MAIN	3680

IF (FREBR(4).EQ.1)	FY(I,J-1)=FY(I,J-1)+ORLX*((QF*ERINF/TOTFRE)	MAIN	3690
2	+ (QH*(CALFLD-FY(I,J-1))))	MAIN	3700
IF (FREBR(5).EQ.1)	FR(I,J)=FR(I,J)+(ORLX*(ERINF/TOTFRE))	MAIN	3710
C		MAIN	3720
C CALCULATE HEAD ERRORS AND COMPARE WITH HERROR		MAIN	3730
C		MAIN	3740
HEDERX=0.		MAIN	3750
HEDERY=0.		MAIN	3760
IF (I.EQ.IMAX) GO TO 547		MAIN	3770
IF (KX(I,J).LT.0.0001) GO TO 547		MAIN	3780
HEDERX=((BLX(I,J)*MFX2*(AFX2**XP))/KX(I,J))-H(I,J)+H(I+1,J)		MAIN	3790
547 IF (J.EQ.JMAX) GO TO 546		MAIN	3800
IF (KY(I,J).LT.0.0001) GO TO 546		MAIN	3810
HEDERY=((BLY(I,J)*MFY2*(AFY2**XP))/KY(I,J))-H(I,J)+H(I,J+1)		MAIN	3820
546 CONTINUE		MAIN	3830
ABERX=ABS(HEDERX)		MAIN	3840
ABERY=ABS(HEDERY)		MAIN	3850
IF (ABERX.GT.HERROR) KNTR=KNTR+1		MAIN	3860
IF (ABERY.GT.HERROR) KNTR=KNTR+1		MAIN	3870
IF (KK.EQ.1) CALL OUT3C (KNTR,HEDERX,HEDERY)		MAIN	3880
562 IF (IO2.EQ.1) CALL OUT1 (MM1,HEDERX,HEDERY,KNTR)		MAIN	3890
551 CONTINUE		MAIN	3900
512 CONTINUE		MAIN	3910
511 CONTINUE		MAIN	3920
C		MAIN	3930
C ***** TEST FOR NEXT ITERATION *****		MAIN	3940
C		MAIN	3950
IF (IO3.EQ.1.AND.(ITER.EQ.MAXIT-1.OR.(CNTER.EQ.0.AND.KNTR.EQ.0)))		MAIN	3960
2 CALL OUT2 (KO2)		MAIN	3970
IF (IO7.EQ.1.AND.(ITER.EQ.MAXIT.OR.(CNTER.EQ.0.AND.KNTR.EQ.0)))		MAIN	3980
2 CALL OUT4		MAIN	3990
IF (IO5.EQ.1) GO TO 558		MAIN	4000
IF (ITER.GE.MAXIT) GO TO 548		MAIN	4010
IF (CNTER.EQ.0.AND.KNTR.EQ.0) GO TO 549		MAIN	4020
GO TO 550		MAIN	4030
548 WRITE (6,532) ITER,MAXIT		MAIN	4040
532 FORMAT ('OEXECUTION TERMINATED - NUMBER OF ITERATIONS (' ,I5,') EQU		MAIN	4050
2ALS MAXIMUM SPECIFIED (' ,I5,')')		MAIN	4060
RETURN		MAIN	4070
549 WRITE (6,533) FERROR,HERROR,ITER		MAIN	4080
533 FORMAT ('OEXECUTION COMPLETED - ALL FLOW ERRORS BELOW MAXIMUM (' ,G		MAIN	4090

212.5,') AND ALL HEAD ERRORS BELOW MAXIMUM ('G12.5,') - ',15,' ITE	MAIN	4100
3R.')	MAIN	4110
RETURN	MAIN	4120
558 CONTINUE	MAIN	4130
IF (CNTR.EQ.0.AND.KNTR.EQ.0) GO TO 510	MAIN	4140
IF (ITER.EQ.MAXIT-1) GO TO 510	MAIN	4150
IF (ITER.EQ.MAXIT) GO TO 579	MAIN	4160
GO TO 550	MAIN	4170
510 CALL OUT3B (KK,FERROR,HERROR,MAXIT,KNTR)	MAIN	4180
IF (KK.EQ.1) GO TO 550	MAIN	4190
579 CONTINUE	MAIN	4200
IF (IOS.EQ.1) CALL OUT3 (ID1,ID2,ID3,ID4,ID5,ID6,ID7,ID8,ID9)	MAIN	4210
RETURN	MAIN	4220
END	MAIN	4230
SUBROUTINE IN1	IN1	0010
C	IN1	0020
C *****	IN1	0030
C ***** SUBROUTINE IN1 *****	IN1	0040
C *****	IN1	0050
C	IN1	0060
C STATEMENT NUMBERS USED ARE 401,405,407,408,409,500,509,510	IN1	0070
C	IN1	0080
C *****DATA FORMAT *****	IN1	0090
C	IN1	0100
C DATA CARDS ARE NODE CARDS	IN1	0110
C	IN1	0120
C CAN BE IN ANY ORDER BUT MUST BE ONE FOR EACH I AND J IN RECTANGULAR	IN1	0130
C ARRAY. ANY BLANK OR ZERO PUNCHED FIELDS WILL BE CONSIDERED	IN1	0140
C UNKNOWN. ENTERED ZERO'S MUST BE PUNCHED 12345 IN LAST 5 COLUMNS	IN1	0150
C OR 12345. IN ANY PART OF FIELD. TOTAL NUMBER OF SPECIFIED HEAD,	IN1	0160
C FLOW; AND PERMEABILITY VALUES MUST = (3*IMAX*JMAX)-IMAX-JMAX.	IN1	0170
C	IN1	0180
C BLANK OR ZERO IN LENGTH FIELDS WILL BE CONSIDERED UNIT LENGTH,	IN1	0190
C OTHERWISE PUNCH NUMBER. ZERO (12345) LENGTHS NOT PERMITTED.	IN1	0200
C	IN1	0210
C C=C	IN1	0220
C	IN1	0230
C CARD COLUMNS	IN1	0240
C	IN1	0250
C 1 THRU 5 NODE NUMBER IN ROW (I) - INTEGER	IN1	0260
C 6 THRU 10 ROW NUMBER (J) - INTEGER	IN1	0270

C	11 THRU 20	HEAD AT NODE (BH(I,J))	INI	0280								
C	21 THRU 30	FLOW IN +X DIRECTION FROM NODE (BFX(I,J))	INI	0290								
C	31 THRU 40	FLOW IN +Y DIRECTION FROM NODE (BFY(I,J))	INI	0300								
C	41 THRU 50	RECHARGE FLOW INTO NODE (BFR(I,J))	INI	0310								
C	51 THRU 60	PERMEABILITY IN +X DIRECTION FROM NODE (BKX(I,J))	INI	0320								
C	61 THRU 70	PERMEABILITY IN +Y DIRECTION FROM NODE (BKY(I,J))	INI	0330								
C	71 THRU 75	LENGTH IN +X DIRECTION FROM NODE (BLX(I,J))	INI	0340								
C	76 THRU 80	LENGTH IN +Y DIRECTION FROM NODE (BLY(I,J))	INI	0350								
C			INI	0360								
C	DECLARE COMMON VARIABLES		INI	0370								
C			INI	0380								
	COMMON	H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	INI	0390								
	2	BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	INI	0400								
	3	KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	INI	0410								
	4	I, J, IMAX, JMAX, ITER, CINTER, ERINF, IMAXM1, JMAXM1, XP	INI	0420								
	REAL	KX, KY	INI	0430								
	INTEGER	CINTER	INI	0440								
C			INI	0450								
C	READ NODE CARDS		INI	0460								
C			INI	0470								
	NODMAX=	IMAX*JMAX	INI	0480								
	DO 500	L=1, NODMAX	INI	0490								
	500 READ	(5, 401) I, J, BH(I, J), BFX(I, J), BFY(I, J), BFR(I, J), BKX(I, J),	INI	0500								
	2	BKY(I, J), BLX(I, J), BLY(I, J)	INI	0510								
	401 FORMAT	(2I5, 6F10.0, 2F5.0)	INI	0520								
C			INI	0530								
C	CORRECT LENGTH VALUES		INI	0540								
C			INI	0550								
	DO 405	I=1, IMAX	INI	0560								
	DO 409	J=1, JMAX	INI	0570								
	IF (BLX(I, J).EQ.0.)	BLX(I, J)=1.	INI	0580								
	IF (BLY(I, J).EQ.0.)	BLY(I, J)=1.	INI	0590								
	409 CONTINUE		INI	0600								
	405 CONTINUE		INI	0610								
C			INI	0620								
C	WRITE OUT ENTERED DATA		INI	0630								
C			INI	0640								
	WRITE(6, 408)		INI	0650								
	408 FORMAT(1	I	INI	0660								
	2	BFR	BKX	I	BKY	J	BLX	BH	BFX	BFY	INI	0670
	DO 509	I=1, IMAX	INI	0680								
			INI	0680								

```

DO 510 J=1,JMAX
WRITE(6,407) I,J,BH(I,J),BFX(I,J),BFY(I,J),BFR(I,J),BKX(I,J),
2          BKY(I,J),BLX(I,J),BLY(I,J)
407 FORMAT (' ',12X,15,5X,15,3X,8F10.3)
510 CONTINUE
509 CONTINUE
C
C TEST FOR CORRECT NUMBER OF ENTERED VALUES
C
C CALL CAL2
C
C ASSIGN INITIAL VALUES
C
C CALL CAL3 (MFAC,KCHEK)
RETURN
END
SUBROUTINE IN2
C
C *****
C ***** SUBROUTINE IN2 *****
C *****
C
C HIGHEST STATEMENT NUMBER IS 116
C
C ***** DATA FORMAT *****
C
C FIRST DATA CARD (AFTER PARAMETER CARD) IS GENERAL TO LOAD ALL ARRAYS.
C ENTER VALUES IN THREE OF FIRST SIX FOR CORRECT NUMBER OF ENTERED
C VALUES. BLANK OR ZERO CONSIDERED UNKNOWN, 12345.0 FOR ZERO, DECIMAL
C POINT TO RIGHT OF LAST POSITION UNLESS PUNCHED, ZERO BLX AND BLY
C NOT PERMITTED.
C
C CARD COLUMNS
C
C 1 THRU 10 HEAD (BH)
C 11 20 FLOW IN BRANCH TO RIGHT, POSITIVE IF TO RIGHT (BFX)
C 21 30 FLOW IN BRANCH ABOVE, POSITIVE IF UP (BFY)
C 31 40 RECHARGE FLOW, POSITIVE IF INTO SYSTEM (BFR)
C 41 50 PERMEABILITY IN BRANCH TO RIGHT (BKX)
C 51 60 PERMEABILITY IN BRANCH ABOVE (BKY)
C 61 70 LENGTH OF BRANCH TO RIGHT (BLX)

```

```

IN1 0690
IN1 0700
IN1 0710
IN1 0720
IN1 0730
IN1 0740
IN1 0750
IN1 0760
IN1 0770
IN1 0780
IN1 0790
IN1 0800
IN1 0810
IN1 0820
IN1 0830
IN1 0840
IN2 0010
IN2 0020
IN2 0030
IN2 0040
IN2 0050
IN2 0060
IN2 0070
IN2 0080
IN2 0090
IN2 0100
IN2 0110
IN2 0120
IN2 0130
IN2 0140
IN2 0150
IN2 0160
IN2 0170
IN2 0180
IN2 0190
IN2 0200
IN2 0210
IN2 0220
IN2 0230
IN2 0240
IN2 0250

```

```

C      71      80  LENGTH OF BRANCH ABOVE (BLY)
C
C      C=C
C
C  REMAINING DATA CARDS ARE CHANGE CARDS WITH FOUR NODES PER CARD
C
C  CARD COLUMNS
C
C      1          DIGIT INDICATING VARIABLE:
C      21
C      41          1 FOR BH      5 FOR BKX
C      61          2 FOR BFX     6 FOR BKY
C          3 FOR BFY     7 FOR BLX
C          4 FOR BFR     8 FOR BLY
C
C          0 TO SIGNAL END OF DATA
C
C      2 THRU 5  NODE NUMBER IN ROW (I) - INTEGER
C      22      25
C      42      45
C      62      65
C
C      6,26,46,66  BLANK
C
C      7 THRU 10  ROW NUMBER (J) - INTEGER
C      27      30
C      47      50
C      67      70
C
C      C=C
C
C      11 THRU 20  VALUE OF VARIABLE
C      31      40
C      51      60
C      71      80
C
C  DECLARE COMMON VARIABLES
C
C  COMMON  H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),
2         BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),
3         KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),

```

```

IN2  0260
IN2  0270
IN2  0280
IN2  0290
IN2  0300
IN2  0310
IN2  0320
IN2  0330
IN2  0340
IN2  0350
IN2  0360
IN2  0370
IN2  0380
IN2  0390
IN2  0400
IN2  0410
IN2  0420
IN2  0430
IN2  0440
IN2  0450
IN2  0460
IN2  0470
IN2  0480
IN2  0490
IN2  0500
IN2  0510
IN2  0520
IN2  0530
IN2  0540
IN2  0550
IN2  0560
IN2  0570
IN2  0580
IN2  0590
IN2  0600
IN2  0610
IN2  0620
IN2  0630
IN2  0640
IN2  0650
IN2  0660

```

	4	I, J, IMAX, JMAX, ITER, CNTER, ERINF, IMAXM1, JMAXM1, XP	IN2	0670
		REAL KX, KY	IN2	0680
		INTEGER CNTER	IN2	0690
C			IN2	0700
C	DECLARE	IN2 VARIABLES	IN2	0710
C			IN2	0720
		REAL IG1, IG2, IG3, IG4, IG5, IG6, IG7, IG8	IN2	0730
C			IN2	0740
C	READ	GENERAL DATA CARD	IN2	0750
C			IN2	0760
		READ (5, 108) IG1, IG2, IG3, IG4, IG5, IG6, IG7, IG8	IN2	0770
	108	FORMAT (8F10.0)	IN2	0780
		DO 109 I=1, IMAX	IN2	0790
		DO 110 J=1, JMAX	IN2	0800
		BH(I, J)=IG1	IN2	0810
		BFR(I, J)=IG4	IN2	0820
	110	CONTINUE	IN2	0830
	109	CONTINUE	IN2	0840
		DO 111 I=1, IMAXM1	IN2	0850
		DO 112 J=1, JMAX	IN2	0860
		BFX(I, J)=IG2	IN2	0870
		BKX(I, J)=IG5	IN2	0880
		BLX(I, J)=IG7	IN2	0890
	112	CONTINUE	IN2	0900
	111	CONTINUE	IN2	0910
		DO 113 I=1, IMAX	IN2	0920
		DO 114 J=1, JMAXM1	IN2	0930
		BFY(I, J)=IG3	IN2	0940
		BKY(I, J)=IG6	IN2	0950
		RLY(I, J)=IG8	IN2	0960
	114	CONTINUE	IN2	0970
	113	CONTINUE	IN2	0980
C			IN2	0990
C	READ	CHANGE DATA CARDS	IN2	1000
C			IN2	1010
	116	READ (5, 100) IV1, I1, J1, V1, IV2, I2, J2, V2, IV3, I3, J3, V3, IV4, I4, J4, V4	IN2	1020
	100	FORMAT (4(I1, I4, IX, I4, F10.0))	IN2	1030
		I=I1	IN2	1040
		J=J1	IN2	1050
		IF (IV1.EQ.0) GO TO 115	IN2	1060
		IF (IV1.EQ.1) BH(I, J)=V1	IN2	1070

```

IF (IV1.EQ.2) BFX(I,J)=V1
IF (IV1.EQ.3) BFY(I,J)=V1
IF (IV1.EQ.4) BFR(I,J)=V1
IF (IV1.EQ.5) BKX(I,J)=V1
IF (IV1.EQ.6) BKY(I,J)=V1
IF (IV1.EQ.7) BLX(I,J)=V1
IF (IV1.EQ.8) BLY(I,J)=V1
I=I2
J=J2
IF (IV2.EQ.0) GO TO 115
IF (IV2.EQ.1) BH(I,J)=V2
IF (IV2.EQ.2) BFX(I,J)=V2
IF (IV2.EQ.3) BFY(I,J)=V2
IF (IV2.EQ.4) BFR(I,J)=V2
IF (IV2.EQ.5) BKX(I,J)=V2
IF (IV2.EQ.6) BKY(I,J)=V2
IF (IV2.EQ.7) BLX(I,J)=V2
IF (IV2.EQ.8) BLY(I,J)=V2
I=I3
J=J3
IF (IV3.EQ.0) GO TO 115
IF (IV3.EQ.1) BH(I,J)=V3
IF (IV3.EQ.2) BFX(I,J)=V3
IF (IV3.EQ.3) BFY(I,J)=V3
IF (IV3.EQ.4) BFR(I,J)=V3
IF (IV3.EQ.5) BKX(I,J)=V3
IF (IV3.EQ.6) BKY(I,J)=V3
IF (IV3.EQ.7) BLX(I,J)=V3
IF (IV3.EQ.8) BLY(I,J)=V3
I=I4
J=J4
IF (IV4.EQ.0) GO TO 115
IF (IV4.EQ.1) BH(I,J)=V4
IF (IV4.EQ.2) BFX(I,J)=V4
IF (IV4.EQ.3) BFY(I,J)=V4
IF (IV4.EQ.4) BFR(I,J)=V4
IF (IV4.EQ.5) BKX(I,J)=V4
IF (IV4.EQ.6) BKY(I,J)=V4
IF (IV4.EQ.7) BLX(I,J)=V4
IF (IV4.EQ.8) BLY(I,J)=V4
GO TO 116

```

```

IN2 1080
IN2 1090
IN2 1100
IN2 1110
IN2 1120
IN2 1130
IN2 1140
IN2 1150
IN2 1160
IN2 1170
IN2 1180
IN2 1190
IN2 1200
IN2 1210
IN2 1220
IN2 1230
IN2 1240
IN2 1250
IN2 1260
IN2 1270
IN2 1280
IN2 1290
IN2 1300
IN2 1310
IN2 1320
IN2 1330
IN2 1340
IN2 1350
IN2 1360
IN2 1370
IN2 1380
IN2 1390
IN2 1400
IN2 1410
IN2 1420
IN2 1430
IN2 1440
IN2 1450
IN2 1460
IN2 1470
IN2 1480

```

C		IN2	1490
C	WRITE OUT ENTERED DATA	IN2	1500
C		IN2	1510
	115 WRITE (6,101)	IN2	1520
	101 FORMAT ('1DATA ENTERED USING SUBROUTINE IN2')	IN2	1530
	WRITE (6,102)	IN2	1540
	102 FORMAT ('0',14X,'1',9X,'J',9X,'BH',8X,'BFX',8X,'BFY',7X,'BFR',8X,	IN2	1550
	2 'BKX',8X,'BKY',8X,'BLX',8X,'BLY')	IN2	1560
	DO 103 I=1,IMAX	IN2	1570
	WRITE (6,104)	IN2	1580
	104 FORMAT (' ')	IN2	1590
	DO 105 J=1,JMAX	IN2	1600
	WRITE (6,107) I,J,BH(I,J),BFX(I,J),BFY(I,J),BFR(I,J),BKX(I,J),	IN2	1610
	2 BKY(I,J),BLX(I,J),BLY(I,J)	IN2	1620
	107 FORMAT (' ',12X,15,5X,15,3X,8(G10.3,1X))	IN2	1630
	105 CONTINUE	IN2	1640
	103 CONTINUE	IN2	1650
	RETURN	IN2	1660
	END	IN2	1670
	SUBROUTINE IN3	IN3	0010
C		IN3	0020
C	*****	IN3	0030
C	***** SUBROUTINE IN3 *****	IN3	0040
C	*****	IN3	0050
C		IN3	0060
C	HIGHEST STATEMENT NUMBER IS 104	IN3	0070
C		IN3	0080
C	DECLARE COMMON VARIABLES	IN3	0090
C		IN3	0100
	COMMON H(38,27), BH(38,27), FX(38,27),BFX(38,27), FY(38,27),	IN3	0110
	2 BFY(38,27); FR(38,27),BFR(38,27), KX(38,27),BKX(38,27),	IN3	0120
	3 KY(38,27),BKY(38,27),BLX(38,27),BLY(38,27), IA(38,27),	IN3	0130
	4 I,J,IMAX,JMAX,ITER,CNTER,ERINF,IMAXM1,JMAXM1,XP	IN3	0140
	REAL KX,KY	IN3	0150
	INTEGER CNTER	IN3	0160
C		IN3	0170
C	READ INITIAL VALUES FROM CARDS	IN3	0180
C		IN3	0190
	READ (5,100) NCARD	IN3	0200
	100 FORMAT (I10)	IN3	0210
	DO 101 N=1,NCARD	IN3	0220

READ (5,102) I,J,H(I,J),FX(I,J),FY(I,J),FR(I,J),KX(I,J),KY(I,J)	IN3	0230
102 FORMAT (2I5,6E10.3)	IN3	0240
101 CONTINUE	IN3	0250
C	IN3	0260
C ASSIGN ENTERED VALUES IF DEFINED (TO PERMIT CHANGES)	IN3	0270
C	IN3	0280
DO 103 I=1,IMAX	IN3	0290
DO 104 J=1,JMAX	IN3	0300
IF (BH (I,J).NE.0 .) H (I,J)=BH (I,J)	IN3	0310
IF (BH (I,J).EQ.12345.) H (I,J)=0.	IN3	0320
IF (BFX(I,J).NE.0 .) FX(I,J)=BFX(I,J)	IN3	0330
IF (BFX(I,J).EQ.12345.) FX(I,J)=0.	IN3	0340
IF (BFY(I,J).NE.0 .) FY(I,J)=BFY(I,J)	IN3	0350
IF (BFY(I,J).EQ.12345.) FY(I,J)=0.	IN3	0360
IF (BFR(I,J).NE.0 .) FR(I,J)=BFR(I,J)	IN3	0370
IF (BFR(I,J).EQ.12345.) FR(I,J)=0.	IN3	0380
IF (BKX(I,J).NE.0 .) KX(I,J)=BKX(I,J)	IN3	0390
IF (BKX(I,J).EQ.12345.) KX(I,J)=0.	IN3	0400
IF (BKY(I,J).NE.0 .) KY(I,J)=BKY(I,J)	IN3	0410
IF (BKY(I,J).EQ.12345.) KY(I,J)=0.	IN3	0420
104 CONTINUE	IN3	0430
103 CONTINUE	IN3	0440
RETURN	IN3	0450
END	IN3	0460
SUBROUTINE CAL1	CAL1	0010
C	CAL1	0020
C *****	CAL1	0030
C ***** SUBROUTINE CAL1 *****	CAL1	0040
C *****	CAL1	0050
C	CAL1	0060
C HIGHEST STATEMENT NUMBER IS 101	CAL1	0070
C	CAL1	0080
C DECLARE COMMON VARIABLES	CAL1	0090
C	CAL1	0100
COMMON H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	CAL1	0110
2 BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	CAL1	0120
3 KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	CAL1	0130
4 I, J, IMAX, JMAX, ITER, CNTER, ERINF, IMAXM1, JMAXM1, XP	CAL1	0140
REAL KX, KY	CAL1	0150
INTEGER CNTER	CAL1	0160
IF (I.EQ.IMAX.OR.BKX(I,J).NE.0.) GO TO 100	CAL1	0170

HDIF=H(I,J)-H(I+1,J)	CAL1	0180
IF (HDIF.LT.0.00001.AND.HDIF.GT.-0.00001) HDIF=1.	CAL1	0190
IF (FX(I,J).LT.1.0E-70.AND.FX(I,J).GT.-1.0E-70) GO TO 100	CAL1	0200
AFX2=ABS(FX(I,J))	CAL1	0210
KX(I,J)=ABS((BLX(I,J)*(AFX2**XP))/HDIF)	CAL1	0220
100 IF (J.EQ.JMAX.OR.BKY(I,J).NE.0) GO TO 101	CAL1	0230
HDIF=H(I,J)-H(I,J+1)	CAL1	0240
IF (HDIF.LT.0.00001.AND.HDIF.GT.-0.00001) HDIF=1.	CAL1	0250
IF (FY(I,J).LT.1.0E-70.AND.FY(I,J).GT.-1.0E-70) GO TO 101.	CAL1	0260
AFY2=ABS(FY(I,J))	CAL1	0270
KY(I,J)=ABS((BLY(I,J)*(AFY2**XP))/HDIF)	CAL1	0280
101 RETURN	CAL1	0290
END	CAL1	0300
SUBROUTINE CAL2 (MGO)	CAL2	0010
C *****	CAL2	0020
C *****	CAL2	0030
C ***** SUBROUTINE CAL2 *****	CAL2	0040
C *****	CAL2	0050
C	CAL2	0060
C HIGHEST STATEMENT NUMBER IS 115	CAL2	0070
C OTHER NUMBER AVAILABLE IS 106	CAL2	0080
C	CAL2	0090
C DECLARE COMMON VARIABLES	CAL2	0100
C	CAL2	0110
COMMON H(38,27); BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	CAL2	0120
2 BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	CAL2	0130
3 KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	CAL2	0140
4 I,J,IMAX,JMAX,ITER,CNTER,ERINF,IMAXM1,JMAXM1,XP	CAL2	0150
REAL KX,KY	CAL2	0160
INTEGER CNTER	CAL2	0170
C	CAL2	0180
C COUNT ENTERED VALUES	CAL2	0190
C	CAL2	0200
KOUNT=0	CAL2	0210
DO 100 I=1,IMAX	CAL2	0220
DO 101 J=1,JMAX	CAL2	0230
IF (BH(I,J).NE.0) KOUNT=KOUNT+1	CAL2	0240
IF (BFR(I,J).NE.0.) KOUNT=KOUNT+1	CAL2	0250
101 CONTINUE	CAL2	0260
100 CONTINUE	CAL2	0270
DO 102 I=1,IMAXM1	CAL2	0280

DO 103 J=1,JMAX	CAL2 0290
IF (BFX(I,J).NE.0.) KOUNT=KOUNT+1	CAL2 0300
IF (BKX(I,J).NE.0.) KOUNT=KOUNT+1	CAL2 0310
103 CONTINUE	CAL2 0320
102 CONTINUE	CAL2 0330
DO 104 I=1,IMAX	CAL2 0340
DO 105 J=1,JMAXM1	CAL2 0350
IF (BFY(I,J).NE.0.) KOUNT=KOUNT+1	CAL2 0360
IF (BKY(I,J).NE.0.) KOUNT=KOUNT+1	CAL2 0370
105 CONTINUE	CAL2 0380
104 CONTINUE	CAL2 0390
C	CAL2 0400
C MAKE TESTS	CAL2 0410
C	CAL2 0420
KBV=(3*IMAX*JMAX)-IMAX-JMAX	CAL2 0430
KDIF=ABS(KBV-KOUNT)	CAL2 0440
KK1=0	CAL2 0450
IF (KBV.GT.KOUNT) KK1=1	CAL2 0460
IF (KBV.LT.KOUNT) KK1=2	CAL2 0470
C	CAL2 0480
C WRITE MESSAGES	CAL2 0490
C	CAL2 0500
WRITE (6,107)	CAL2 0510
107 FORMAT ('DATA CHECKED USING SUBROUTINE CAL2')	CAL2 0520
IF (KK1.EQ.0) GO TO 108	CAL2 0530
IF (KK1.EQ.1) GO TO 109	CAL2 0540
GO TO 110	CAL2 0550
108 WRITE (6,111) KOUNT,KBV	CAL2 0560
111 FORMAT (' NUMBER OF ENTERED VALUES (' ,I5,') EQUALS REQUIRED N NUMBER (' ,I5,')')	CAL2 0570
RETURN	CAL2 0580
109 CONTINUE	CAL2 0590
IF (MGO.EQ.1) GO TO 114	CAL2 0600
WRITE (6,112) KOUNT,KBV,KDIF	CAL2 0610
112 FORMAT (' EXECUTION TERMINATED - NUMBER OF ENTERED VALUES (' , 2I5,') IS LESS THAN REQUIRED NUMBER (' ,I5,') BY ' ,I5)	CAL2 0620
CALL EXIT	CAL2 0630
110 CONTINUE	CAL2 0640
IF (MGO.EQ.1) GO TO 114	CAL2 0650
WRITE (6,113) KOUNT,KBV,KDIF	CAL2 0660
113 FORMAT (' EXECUTION TERMINATED - NUMBER OF ENTERED VALUES (' ,	CAL2 0670
	CAL2 0680
	CAL2 0690

215,') EXCEEDS REQUIRED NUMBER (' ,15,') BY ' ,15)	CAL2 0700
CALL EXIT	CAL2 0710
114 CONTINUE	CAL2 0720
WRITE (6,115) KOUNT,KBV	CAL2 0730
115 FORMAT ('0 NUMBER OF ENTERED VALUES (' ,15,') NOT EQUAL TO REQU	CAL2 0740
2IRED NUMBER (' ,15,') - EXECUTION CONTINUES')	CAL2 0750
RETURN	CAL2 0760
END	CAL2 0770
SUBROUTINE CAL3 (MFAC,KCHEK)	CAL3 0010
C *****	CAL3 0020
C *****	CAL3 0030
C ***** SUBROUTINE CAL3 *****	CAL3 0040
C *****	CAL3 0050
C *****	CAL3 0060
C HIGHEST STATEMENT NUMBER IS 137	CAL3 0070
C	CAL3 0080
C DECLARE COMMON VARIABLES	CAL3 0090
C	CAL3 0100
COMMON H(38,27), BH(38,27), FX(38,27),BFX(38,27), FY(38,27),	CAL3 0110
2 BFY(38,27), FR(38,27),BFR(38,27), KX(38,27),BKX(38,27),	CAL3 0120
3 KY(38,27),BKY(38,27),BLX(38,27),BLY(38,27), IA(38,27),	CAL3 0130
4 I,J,IMAX,JMAX,ITER,CNTER,ERINF,IMAXMI,JMAXMI,XP	CAL3 0140
REAL KX,KY	CAL3 0150
INTEGER CNTER	CAL3 0160
C	CAL3 0170
C SUM AND COUNT SPECIFIED NON-ZERO BH	CAL3 0180
C	CAL3 0190
SUMH=0.	CAL3 0200
NUMH=0	CAL3 0210
DO 100 I=1,IMAX	CAL3 0220
DO 101 J=1,JMAX	CAL3 0230
IF (BH(I,J).EQ.12345..OR.BH(I,J).EQ.0.) GO TO 103	CAL3 0240
SUMH=SUMH+BH(I,J)	CAL3 0250
NUMH=NUMH+1	CAL3 0260
103 CONTINUE	CAL3 0270
101 CONTINUE	CAL3 0280
100 CONTINUE	CAL3 0290
C	CAL3 0300
C SUM AND COUNT SPECIFIED K AND SUM L	CAL3 0310
C	CAL3 0320
SUMK=0.	CAL3 0330

```

NUMK=0
SUML=0.
NUML=(2*IMAX*JMAX)-IMAX-JMAX
DO 104 I=1,IMAXM1
DO 105 J=1,JMAX
IF (BKX(I,J).EQ.12345..OR.BKX(I,J).EQ.0.) GO TO 106
SUMK=SUMK+BKX(I,J)
NUMK=NUMK+1
106 SUML=SUML+BLX(I,J)
105 CONTINUE
104 CONTINUE
DO 107 I=1,IMAX
DO 108 J=1,JMAXM1
IF (BKY(I,J).EQ.12345..OR.BKY(I,J).EQ.0.) GO TO 109
SUMK=SUMK+BKY(I,J)
NUMK=NUMK+1
109 SUML=SUML+BLY(I,J)
108 CONTINUE
107 CONTINUE
C
C CALCULATE BFR BALANCE
C
SUMFR=0.
DO 102 I=1,IMAX
DO 110 J=1,JMAX
IF (BFR(I,J).NE.12345..AND.BFR(I,J).NE.0.) SUMFR=SUMFR+BFR(I,J)
110 CONTINUE
102 CONTINUE
C
C COUNT UNSPECIFIED BFR EXCLUDING THOSE WITH INAPPROPRIATE HEAD
C
IF (NUMH.EQ.0) NUMH=1
HEADAV=SUMH/NUMH
NUMFR=0
DO 111 I=1,IMAX
DO 112 J=1,JMAX
IF (SUMFR.LE.0.) GO TO 113
IF (BFR(I,J).NE.0.) GO TO 114
IF (BH(I,J).EQ.12345..AND.HEADAV.GE.0.) NUMFR=NUMFR+1
IF (BH(I,J).NE.0..AND.BH(I,J).NE.12345..AND.BH(I,J).LE.HEADAV)
2 NUMFR=NUMFR+1

```

```

CAL3 0340
CAL3 0350
CAL3 0360
CAL3 0370
CAL3 0380
CAL3 0390
CAL3 0400
CAL3 0410
CAL3 0420
CAL3 0430
CAL3 0440
CAL3 0450
CAL3 0460
CAL3 0470
CAL3 0480
CAL3 0490
CAL3 0500
CAL3 0510
CAL3 0520
CAL3 0530
CAL3 0540
CAL3 0550
CAL3 0560
CAL3 0570
CAL3 0580
CAL3 0590
CAL3 0600
CAL3 0610
CAL3 0620
CAL3 0630
CAL3 0640
CAL3 0650
CAL3 0660
CAL3 0670
CAL3 0680
CAL3 0690
CAL3 0700
CAL3 0710
CAL3 0720
CAL3 0730
CAL3 0740

```

```

GO TO 114
113 IF (BFR(I,J).NE.0.) GO TO 114
    IF (BH(I,J).EQ.12345..AND.HEADAV.LE.0.) NUMFR=NUMFR+1
    IF (BH(I,J).NE.0..AND.BH(I,J).NE.12345..AND.BH(I,J).GE.HEADAV)
        2 NUMFR=NUMFR+1
114 CONTINUE
112 CONTINUE
111 CONTINUE
C
C DISTRIBUTE FR TO BALANCE
C
    IF (NUMFR.EQ.0) NUMFR=1
    FRBAL=SUMFR/NUMFR
    DO 115 I=1,IMAX
    DO 116 J=1,JMAX
    IF (BFR(I,J).EQ.0.) GO TO 137
    FR(I,J)=BFR(I,J)
    IF (BFR(I,J).EQ.12345.) FR(I,J)=0.
    GO TO 118
137 IF (SUMFR.LE.0.) GO TO 117
    IF (BH(I,J).EQ.12345..AND.HEADAV.GE.0.) FR(I,J)=-FRBAL
    IF (BH(I,J).NE.0..AND.BH(I,J).NE.12345..AND.BH(I,J).LE.HEADAV)
        2 FR(I,J)=-FRBAL
    GO TO 118
117 IF (BH(I,J).EQ.12345..AND.HEADAV.LE.0.) FR(I,J)=-FRBAL
    IF (BH(I,J).NE.0..AND.BH(I,J).NE.12345..AND.BH(I,J).GE.HEADAV)
        2 FR(I,J)=-FRBAL
118 CONTINUE
116 CONTINUE
115 CONTINUE
C
C ASSIGN INITIAL VALUES OF H
C
    IF (NUMK.EQ.0) NUMK=1
    KAVER=SUMK/NUMK
    LENGAV=SUML/NUML
    IF (KAVER.EQ.0.) KAVER=1.
    RATIO=LENGAV/KAVER
    IF (RATIO.EQ.0.) RATIO=1.
    DO 119 I=1,IMAX
    DO 120 J=1,JMAX

```

```

CAL3 0750
CAL3 0760
CAL3 0770
CAL3 0780
CAL3 0790
CAL3 0800
CAL3 0810
CAL3 0820
CAL3 0830
CAL3 0840
CAL3 0850
CAL3 0860
CAL3 0870
CAL3 0880
CAL3 0890
CAL3 0900
CAL3 0910
CAL3 0920
CAL3 0930
CAL3 0940
CAL3 0950
CAL3 0960
CAL3 0970
CAL3 0980
CAL3 0990
CAL3 1000
CAL3 1010
CAL3 1020
CAL3 1030
CAL3 1040
CAL3 1050
CAL3 1060
CAL3 1070
CAL3 1080
CAL3 1090
CAL3 1100
CAL3 1110
CAL3 1120
CAL3 1130
CAL3 1140
CAL3 1150

```

IF (BH(I,J).NE.0.) H(I,J)=BH(I,J)	CAL3	1160
IF (BH(I,J).EQ.0..AND.BFR(I,J).NE.0..AND.BFR(I,J).NE.12345.)	CAL3	1170
2 H(I,J)=HEADAV+(MFAC*RATIO*BFR(I,J))	CAL3	1180
IF (BH(I,J).EQ.0..AND.FR(I,J).NE.0.)	CAL3	1190
2 H(I,J)=HEADAV+(MFAC*RATIO*FR(I,J))	CAL3	1200
120 CONTINUE	CAL3	1210
119 CONTINUE	CAL3	1220
CALL CAL4	CAL3	1230
DO 121 I=1,IMAX	CAL3	1240
DO 122 J=1,JMAX	CAL3	1250
IF (H(I,J).EQ.12345.) H(I,J)=0.	CAL3	1260
122 CONTINUE	CAL3	1270
121 CONTINUE	CAL3	1280
C	CAL3	1290
C ASSIGN INITIAL VALUES OF FX, FY, KX, AND KY	CAL3	1300
C	CAL3	1310
KCHEK=0	CAL3	1320
DO 129 I=1,IMAXM1	CAL3	1330
DO 130 J=1,JMAX	CAL3	1340
FX(I,J)=BFX(I,J)	CAL3	1350
IF (BFX(I,J).EQ.12345.) FX(I,J)=0.	CAL3	1360
IF (BKX(I,J).NE.0.) GO TO 135	CAL3	1370
KX(I,J)=KAVER	CAL3	1380
KCHEK=1	CAL3	1390
GO TO 131	CAL3	1400
135 KX(I,J)=BKX(I,J)	CAL3	1410
IF (BKX(I,J).EQ.12345.) KX(I,J)=0.	CAL3	1420
131 CONTINUE	CAL3	1430
130 CONTINUE	CAL3	1440
129 CONTINUE	CAL3	1450
DO 132 I=1,IMAX	CAL3	1460
DO 133 J=1,JMAXM1	CAL3	1470
FY(I,J)=BFY(I,J)	CAL3	1480
IF (BFY(I,J).EQ.12345.) FY(I,J)=0.	CAL3	1490
IF (BKY(I,J).NE.0.) GO TO 136	CAL3	1500
KY(I,J)=KAVER	CAL3	1510
KCHEK=1	CAL3	1520
GO TO 134	CAL3	1530
136 KY(I,J)=BKY(I,J)	CAL3	1540
IF (BKY(I,J).EQ.12345.) KY(I,J)=0.	CAL3	1550
134 CONTINUE	CAL3	1560

133 CONTINUE	CAL3	1570
132 CONTINUE	CAL3	1580
C	CAL3	1590
C WRITE OUT ASSIGNED VALUES	CAL3	1600
C	CAL3	1610
WRITE (6,123) MFAC	CAL3	1620
123 FORMAT ('1INITIAL VALUES CALCULATED USING SUBROUTINE CAL3 WITH HEA	CAL3	1630
2D MULTIPLIER OF ',11)	CAL3	1640
WRITE (6,124)	CAL3	1650
124 FORMAT ('0',14X,'1',9X,'J',10X,'H',12X,'FX',12X,'FY',12X,'FR',12X,	CAL3	1660
2 'KX',12X,'KY')	CAL3	1670
DO 125 I=1,IMAX	CAL3	1680
WRITE (6,126)	CAL3	1690
126 FORMAT ('')	CAL3	1700
DO 127 J=1,JMAX	CAL3	1710
WRITE (6,128) I,J,H(I,J),FX(I,J),FY(I,J),FR(I,J),KX(I,J),KY(I,J)	CAL3	1720
128 FORMAT (' ',12X,15,5X,15,3X,6(G10.3,4X))	CAL3	1730
127 CONTINUE	CAL3	1740
125 CONTINUE	CAL3	1750
RETURN	CAL3	1760
END	CAL3	1770
SUBROUTINE CAL4	CAL4	0010
C	CAL4	0020
C *****	CAL4	0030
C ***** SUBROUTINE CAL4 *****	CAL4	0040
C *****	CAL4	0050
C	CAL4	0060
C HIGHEST STATEMENT NUMBER IS 115	CAL4	0070
C	CAL4	0080
C DECLARE COMMON VARIABLES	CAL4	0090
C	CAL4	0100
COMMON H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	CAL4	0110
2 BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	CAL4	0120
3 KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	CAL4	0130
4 I,J,IMAX,JMAX,ITER,CNTER,ERINF,IMAXM1,JMAXM1,XP	CAL4	0140
REAL KX,KY	CAL4	0150
INTEGER CNTER	CAL4	0160
C	CAL4	0170
C SET IA EQ 1 FOR NODES WITH ASSIGNED VALUES OF H	CAL4	0180
C	CAL4	0190
DO 100 I=1,IMAX	CAL4	0200

```

DO 101 J=1,JMAX
IA(I,J)=0
IF (H(I,J).EQ.0.) GO TO 102
IA(I,J)=1
102 CONTINUE
101 CONTINUE
100 CONTINUE
NLIM=IMAX+JMAX-2
DO 103 I=1,IMAX
DO 104 J=1,JMAX
IF (IA(I,J).NE.1) GO TO 105
DO 106 N=1,NLIM
DO 107 M=1,N
C
C SET THIRD QUADRANT
C
IF (I-M.LT.1) GO TO 108
IF (J-N+M.LT.1) GO TO 108
IF (IA(I-M,J-N+M).EQ.0) GO TO 109
IF (IA(I-M,J-N+M).LE.N+1) GO TO 108
109 H(I-M,J-N+M)=H(I,J)
IA(I-M,J-N+M)=N+1
108 CONTINUE
C
C SET FOURTH QUADRANT
C
IF (I-N+M.LT.1) GO TO 110
IF (J+M.GT.JMAX) GO TO 110
IF (IA(I-N+M,J+M).EQ.0) GO TO 111
IF (IA(I-N+M,J+M).LE.N+1) GO TO 110
111 H(I-N+M,J+M)=H(I,J)
IA(I-N+M,J+M)=N+1
110 CONTINUE
C
C SET FIRST QUADRANT
C
IF (I+M.GT.IMAX) GO TO 112
IF (J+N-M.GT.JMAX) GO TO 112
IF (IA(I+M,J+N-M).EQ.0) GO TO 113
IF (IA(I+M,J+N-M).LE.N+1) GO TO 112
113 H(I+M,J+N-M)=H(I,J)

```

```

CAL4 0210
CAL4 0220
CAL4 0230
CAL4 0240
CAL4 0250
CAL4 0260
CAL4 0270
CAL4 0280
CAL4 0290
CAL4 0300
CAL4 0310
CAL4 0320
CAL4 0330
CAL4 0340
CAL4 0350
CAL4 0360
CAL4 0370
CAL4 0380
CAL4 0390
CAL4 0400
CAL4 0410
CAL4 0420
CAL4 0430
CAL4 0440
CAL4 0450
CAL4 0460
CAL4 0470
CAL4 0480
CAL4 0490
CAL4 0500
CAL4 0510
CAL4 0520
CAL4 0530
CAL4 0540
CAL4 0550
CAL4 0560
CAL4 0570
CAL4 0580
CAL4 0590
CAL4 0600
CAL4 0610

```

IA(I+M,J+N-M)=N+1	CAL4	0620
112 CONTINUE	CAL4	0630
C SET SECOND QUADRANT	CAL4	0640
C	CAL4	0650
IF (I+N-M.GT.IMAX) GO TO 114	CAL4	0660
IF (J-M.LT.1) GO TO 114	CAL4	0670
IF (IA(I+N-M,J-M).EQ.0) GO TO 115	CAL4	0680
IF (IA(I+N-M,J-M).LE.N+1) GO TO 114	CAL4	0690
115 H(I+N-M,J-M)=H(I,J)	CAL4	0700
IA(I+N-M,J-M)=N+1	CAL4	0710
114 CONTINUE	CAL4	0720
107 CONTINUE	CAL4	0730
106 CONTINUE	CAL4	0740
105 CONTINUE	CAL4	0750
104 CONTINUE	CAL4	0760
103 CONTINUE	CAL4	0770
RETURN	CAL4	0780
END	CAL4	0790
SUBROUTINE OUT1 (MM1,HEDERX,HEDERY,KNTR)	CAL4	0800
C	OUT1	0010
C *****	OUT1	0020
C ***** SUBROUTINE OUT1 *****	OUT1	0030
C *****	OUT1	0040
C	OUT1	0050
C HIGHEST STATEMENT NUMBER IS 106	OUT1	0060
C	OUT1	0070
C DECLARE COMMON VARIABLES	OUT1	0080
C	OUT1	0090
COMMON H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	OUT1	0100
2 BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	OUT1	0110
3 KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	OUT1	0120
4 I, J, IMAX, JMAX, ITER, CNTR, ERINF, IMAXM1, JMAXM1, XP	OUT1	0130
REAL KX, KY	OUT1	0140
INTEGER CNTR	OUT1	0150
C	OUT1	0160
C WRITE HEADING FOR PRELIMINARY VALUES	OUT1	0170
C	OUT1	0180
IF (ITER.EQ.1.AND.I.EQ.1.AND.J.EQ.1) GO TO 100	OUT1	0190
GO TO 104	OUT1	0200
100 WRITE (6,106)	OUT1	0210
	OUT1	0220

106	FORMAT ('!OUTPUT WITH SUBROUTINE OUT1')	OUT1	0230
	WRITE (6,102)	OUT1	0240
102	FORMAT('OITERATION	OUT1	0250
	2 FR KX KY ERINF H FEC HEDERX FY HEDERY	OUT1	0260
	3 HEC'//)	OUT1	0270
C		OUT1	0280
C	CHECK FOR WRITE	OUT1	0290
C		OUT1	0300
	104 IF (ITER.EQ.1.AND.I.EQ.1.AND.J.EQ.1) MK1=0	OUT1	0310
	IF (ITER.EQ.1.AND.I.EQ.1.AND.J.EQ.1) MP1=2**MM1	OUT1	0320
	IF (ITER.EQ.1.OR.ITER.EQ.2) GO TO 101	OUT1	0330
	IF (ITER.EQ.MK1.OR.ITER.EQ.MK1+1) GO TO 101	OUT1	0340
	RETURN	OUT1	0350
C		OUT1	0360
C	WRITE VALUES AT NODE	OUT1	0370
C		OUT1	0380
	101 WRITE (6,103) ITER,I,J,H(I,J),FX(I,J),FY(I,J),FR(I,J),KX(I,J),	OUT1	0390
	2 KY(I,J),ERINF,CNTER,HEDERX,HEDERY,KNTR	OUT1	0400
	103 FORMAT (' ',2X,15,5X,15,5X,15,3X,7G10.3,1X,13,1X,2G10.3,1X,13)	OUT1	0410
	IF (I.NE.IMAX.OR.J.NE.JMAX) GO TO 105	OUT1	0420
	IF (ITER.EQ.MK1+1) MK1=MK1+MP1	OUT1	0430
	105 RETURN	OUT1	0440
	END	OUT1	0450
	SUBROUTINE OUT2 (K02)	OUT1	0450
C		OUT2	0010
C	*****	OUT2	0020
C	***** SUBROUTINE OUT2 *****	OUT2	0030
C	*****	OUT2	0040
C		OUT2	0050
C		OUT2	0060
C	HIGHEST STATEMENT NUMBER IS 122	OUT2	0070
C		OUT2	0080
C	DECLARE COMMON VARIABLES	OUT2	0090
C		OUT2	0100
	COMMON H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27);	OUT2	0110
	2 BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	OUT2	0120
	3 KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	OUT2	0130
	4 I, J, IMAX, JMAX, ITER, CNTER, ERINF, IMAXM1, JMAXM1, XP	OUT2	0140
	REAL KX, KY	OUT2	0150
	INTEGER CNTER	OUT2	0160
	IF (K02.EQ.1) GO TO 121	OUT2	0170
	K02=1	OUT2	0180

WRITE (6,120)					OUT2	0190
120 FORMAT ('OUTPUT WITH SUBROUTINE OUT2')					OUT2	0200
WRITE (6,100)					OUT2	0210
100 FORMAT ('0 FINAL VALUES LISTED BELOW. ENTERED VALUE OF 12345 M 2EANS ZERO; ENTERED VALUE OF ZERO MEANS NO VALUE (VALUE TO BE SOLVE 3D FOR)')					OUT2	0220
WRITE (6,122)					OUT2	0230
122 FORMAT ('0 IF TERMINATION DUE TO ITERATION COUNT, VALUES LISTE 2D ARE FOR NEXT TO LAST ITERATION')					OUT2	0240
103 FORMAT (' ')					OUT2	0250
WRITE (6,101)					OUT2	0260
101 FORMAT ('0 I J ENTERED HEAD FINAL HEAD ENTE 2RED RECHARGE FINAL RECHARGE ')					OUT2	0270
WRITE (6,106)					OUT2	0280
106 FORMAT (' 2FLOW INTO NODE IS POSITIVE) ')					OUT2	0290
DO 102 I=1,IMAX					OUT2	0300
WRITE (6,103)					OUT2	0310
DO 104 J=1,JMAX					OUT2	0320
WRITE (6,105) I,J,BH(I,J),H(I,J),BFR(I,J),FR(I,J)					OUT2	0330
105 FORMAT (' ',215,4G20.8)					OUT2	0340
104 CONTINUE					OUT2	0350
102 CONTINUE					OUT2	0360
WRITE (6,107)					OUT2	0370
107 FORMAT ('0 I J ENTERED FLOW IN FINAL FLOW IN ENTE 2RED FLOW IN FINAL FLOW IN ')					OUT2	0380
WRITE (6,108)					OUT2	0390
108 FORMAT (' BRANCH TO RIGHT BRANCH TO RIGHT BR 2ANCH ABOVE BRANCH ABOVE ')					OUT2	0400
WRITE (6,109)					OUT2	0410
109 FORMAT (' (FLOW TO RIGHT IS POSITIVE) 2 (FLOW UP IS POSITIVE) ')					OUT2	0420
DO 110 I=1,IMAX					OUT2	0430
WRITE (6,103)					OUT2	0440
DO 111 J=1,JMAX					OUT2	0450
WRITE (6,105) I,J,BFX(I,J),FX(I,J),BFY(I,J),FY(I,J)					OUT2	0460
111 CONTINUE					OUT2	0470
110 CONTINUE					OUT2	0480
WRITE (6,112)					OUT2	0490
112 FORMAT ('0 I J ENTERED PERM. IN FINAL PERM. IN ENTE 2RED PERM. IN FINAL PERM. IN ')					OUT2	0500
					OUT2	0510
					OUT2	0520
					OUT2	0530
					OUT2	0540
					OUT2	0550
					OUT2	0560
					OUT2	0570
					OUT2	0580
					OUT2	0590

WRITE (6,108)	OUT2 0600
DO 113 I=1,IMAX	OUT2 0610
WRITE (6,103)	OUT2 0620
DO 114 J=1,JMAX	OUT2 0630
WRITE (6,105) I,J,BKX(I,J),KX(I,J),BKY(I,J),KY(I,J)	OUT2 0640
114 CONTINUE	OUT2 0650
113 CONTINUE	OUT2 0660
WRITE (6,115)	OUT2 0670
115 FORMAT ('0 I J ENTERED LENGTH OF ENTERED LENGTH OF ')	OUT2 0680
WRITE (6,116)	OUT2 0690
116 FORMAT (' BRANCH TO RIGHT BRANCH ABOVE ')	OUT2 0700
DO 117 I=1,IMAX	OUT2 0710
WRITE (6,103)	OUT2 0720
DO 118 J=1,JMAX	OUT2 0730
WRITE (6,119) I,J,BLX(I,J),BLY(I,J)	OUT2 0740
119 FORMAT (' ',2I5,2G20.8)	OUT2 0750
118 CONTINUE	OUT2 0760
117 CONTINUE	OUT2 0770
121 RETURN	OUT2 0780
END	OUT2 0790
SUBROUTINE OUT3 (ID1,ID2,ID3,ID4,ID5,ID6,ID7,ID8,ID9)	OUT3 0010
C *****	OUT3 0020
C *****	OUT3 0030
C ***** SUBROUTINE OUT3 *****	OUT3 0040
C *****	OUT3 0050
C	OUT3 0060
C HIGHEST STATEMENT NUMBER IS 134	OUT3 0070
C	OUT3 0080
C DECLARE COMMON VARIABLES	OUT3 0090
C	OUT3 0100
COMMON H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	OUT3 0110
2 BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	OUT3 0120
3 KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	OUT3 0130
4 I,J,IMAX,JMAX,ITER,CNTER,ERINF,IMAXML,JMAXML,XP	OUT3 0140
REAL KX,KY	OUT3 0150
INTEGER CNTER	OUT3 0160
C	OUT3 0170
C DECLARE OUT3 VARIABLES	OUT3 0180
C	OUT3 0190
DIMENSION A(4,16)	OUT3 0200
C	OUT3 0210

```

C INITIALIZE
C
    MAP=1
    ISHEET=1
    JSHEET=1
    M=1
    L=8
    LN=9
C
C SET ARRAY TO ZERO
C
107 CONTINUE
    DO 101 M1=1,4
    DO 102 N=1,16
        A(M1,N)=0.
102 CONTINUE
101 CONTINUE
100 CONTINUE
C
C SET VARIABLES FOR SHEET
C
    LMI7=L-7
    LPL1=L+1
    MPL1=M+1
    MPL2=M+2
    MPL3=M+3
    MPL4=M+4
C
C TRANSFER ACCORDING TO MAP
C
    IF (MAP.EQ.2) GO TO 110
    IF (MAP.EQ.3) GO TO 111
    IF (MAP.EQ.4) GO TO 112
    IF (MAP.EQ.5) GO TO 113
    IF (MAP.EQ.6) GO TO 114
    IF (MAP.EQ.7) GO TO 115
    IF (MAP.EQ.8) GO TO 116
    IF (MAP.EQ.9) GO TO 117
C
C FILL ARRAY FOR MAP 1 AND WRITE
C

```

```

OUT3 0220
OUT3 0230
OUT3 0240
OUT3 0250
OUT3 0260
OUT3 0270
OUT3 0280
OUT3 0290
OUT3 0300
OUT3 0310
OUT3 0320
OUT3 0330
OUT3 0340
OUT3 0350
OUT3 0360
OUT3 0370
OUT3 0380
OUT3 0390
OUT3 0400
OUT3 0410
OUT3 0420
OUT3 0430
OUT3 0440
OUT3 0450
OUT3 0460
OUT3 0470
OUT3 0480
OUT3 0490
OUT3 0500
OUT3 0510
OUT3 0520
OUT3 0530
OUT3 0540
OUT3 0550
OUT3 0560
OUT3 0570
OUT3 0580
OUT3 0590
OUT3 0600
OUT3 0610
OUT3 0620

```

```

      IF (ID1.EQ.1) GO TO 108
      DO 103 L1=LMI7,L
      N=IABS(L1-LN)
      M2=0
      DO 104 M1=M,MPL3
      M2=M2+1
      IF (M1.GT.IMAX.OR.L1.GT.JMAX) GO TO 104
      A(M2,N)=BH(M1,L1)
104 CONTINUE
103 CONTINUE
      GO TO 118
C
C FILL ARRAY FOR MAP 2 AND WRITE
C
110 IF (ID2.EQ.1) GO TO 108
      DO 119 L1=LMI7,L
      N=IABS(L1-LN)
      M2=0
      DO 120 M1=M,MPL3
      M2=M2+1
      IF (M1.GT.IMAX.OR.L1.GT.JMAX) GO TO 120
      A(M2,N)=H(M1,L1)
120 CONTINUE
119 CONTINUE
      GO TO 118
C
C FILL ARRAY FOR MAP 3 AND WRITE
C
111 IF (ID3.EQ.1) GO TO 108
      DO 121 L1=LMI7,L
      N=IABS(L1-LN)
      M2=0
      DO 122 M1=M,MPL3
      M2=M2+1
      IF (M1.GT.IMAX.OR.L1.GT.JMAX) GO TO 122
      A(M2,N)=BFR(M1,L1)
122 CONTINUE
121 CONTINUE
      GO TO 118
C
C FILL ARRAY FOR MAP 4 AND WRITE

```

```

OUT3 0630
OUT3 0640
OUT3 0650
OUT3 0660
OUT3 0670
OUT3 0680
OUT3 0690
OUT3 0700
OUT3 0710
OUT3 0720
OUT3 0730
OUT3 0740
OUT3 0750
OUT3 0760
OUT3 0770
OUT3 0780
OUT3 0790
OUT3 0800
OUT3 0810
OUT3 0820
OUT3 0830
OUT3 0840
OUT3 0850
OUT3 0860
OUT3 0870
OUT3 0880
OUT3 0890
OUT3 0900
OUT3 0910
OUT3 0920
OUT3 0930
OUT3 0940
OUT3 0950
OUT3 0960
OUT3 0970
OUT3 0980
OUT3 0990
OUT3 1000
OUT3 1010
OUT3 1020
OUT3 1030

```

```

C
112 IF (ID4.EQ.1) GO TO 108
    DO 123 L1=LMI7,L
    N=IABS(L1-LN)
    M2=0
    DO 124 M1=M,MPL3
    M2=M2+1
    IF (M1.GT.IMAX.OR.L1.GT.JMAX) GO TO 124
    A(M2,N)=FR(M1,L1)
124 CONTINUE
123 CONTINUE
    GO TO 118

```

```

C
C FILL ARRAY FOR MAP 5 AND WRITE
C

```

```

113 IF (ID5.EQ.1) GO TO 108
    DO 125 L1=LMI7,L
    N=IABS(L1-LN)
    M2=0
    DO 126 M1=M,MPL3
    M2=M2+1
    IF (M1.GT.IMAX.OR.L1.GT.JMAX) GO TO 126
    A(M2,N)=BFY(M1,L1)
    A(M2,N+8)=BFX(M1,L1)
126 CONTINUE
125 CONTINUE
    GO TO 118

```

```

C
C FILL ARRAY FOR MAP 6 AND WRITE
C

```

```

114 IF (ID6.EQ.1) GO TO 108
    DO 127 L1=LMI7,L
    N=IABS(L1-LN)
    M2=0
    DO 128 M1=M,MPL3
    M2=M2+1
    IF (M1.GT.IMAX.OR.L1.GT.JMAX) GO TO 128
    A(M2,N)=FY(M1,L1)
    A(M2,N+8)=FX(M1,L1)
128 CONTINUE
127 CONTINUE

```

```

OUT3 1040
OUT3 1050
OUT3 1060
OUT3 1070
OUT3 1080
OUT3 1090
OUT3 1100
OUT3 1110
OUT3 1120
OUT3 1130
OUT3 1140
OUT3 1150
OUT3 1160
OUT3 1170
OUT3 1180
OUT3 1190
OUT3 1200
OUT3 1210
OUT3 1220
OUT3 1230
OUT3 1240
OUT3 1250
OUT3 1260
OUT3 1270
OUT3 1280
OUT3 1290
OUT3 1300
OUT3 1310
OUT3 1320
OUT3 1330
OUT3 1340
OUT3 1350
OUT3 1360
OUT3 1370
OUT3 1380
OUT3 1390
OUT3 1400
OUT3 1410
OUT3 1420
OUT3 1430
OUT3 1440

```

```

GO TO 118
C
C FILL ARRAY FOR MAP 7 AND WRITE
C
115 IF (ID7.EQ.1) GO TO 108
DO 129 LI=LMI7,L
N=IABS(LI-LN)
M2=0
DO 130 M1=M,MPL3
M2=M2+1
IF (M1.GT.IMAX.OR.LI.GT.JMAX) GO TO 130
A(M2,N)=BKY(M1,L1)
A(M2,N+8)=BKX(M1,L1)
130 CONTINUE
129 CONTINUE
GO TO 118
C
C FILL ARRAY FOR MAP 8 AND WRITE
C
116 IF (ID8.EQ.1) GO TO 108
DO 131 LI=LMI7,L
N=IABS(LI-LN)
M2=0
DO 132 M1=M,MPL3
M2=M2+1
IF (M1.GT.IMAX.OR.LI.GT.JMAX) GO TO 132
A(M2,N)=KY(M1,L1)
A(M2,N+8)=KX(M1,L1)
132 CONTINUE
131 CONTINUE
GO TO 118
C
C FILL ARRAY FOR MAP 9 AND WRITE
C
117 IF (ID9.EQ.1) GO TO 108
DO 133 LI=LMI7,L
N=IABS(LI-LN)
M2=0
DO 134 M1=M,MPL3
M2=M2+1
IF (M1.GT.IMAX.OR.LI.GT.JMAX) GO TO 134

```

```

OUT3 1450
OUT3 1460
OUT3 1470
OUT3 1480
OUT3 1490
OUT3 1500
OUT3 1510
OUT3 1520
OUT3 1530
OUT3 1540
OUT3 1550
OUT3 1560
OUT3 1570
OUT3 1580
OUT3 1590
OUT3 1600
OUT3 1610
OUT3 1620
OUT3 1630
OUT3 1640
OUT3 1650
OUT3 1660
OUT3 1670
OUT3 1680
OUT3 1690
OUT3 1700
OUT3 1710
OUT3 1720
OUT3 1730
OUT3 1740
OUT3 1750
OUT3 1760
OUT3 1770
OUT3 1780
OUT3 1790
OUT3 1800
OUT3 1810
OUT3 1820
OUT3 1830
OUT3 1840
OUT3 1850

```

```

      A(M2,N)=BLY(M1,L1)
      A(M2,N+8)=BLX(M1,L1)
134 CONTINUE
133 CONTINUE
      GO TO 118
C
C INCREMENT JSHEET
C
105 CONTINUE
      IF (L.GT.JMAX) GO TO 106
      L=L+8
      JSHEET=JSHEET+1
      LN=LN+8
      GO TO 107
C
C INCREMENT ISHEET
C
106 CONTINUE
      L=8
      JSHEET=1
      LN=9
      IF (M+4.GT.IMAX) GO TO 108
      M=M+4
      ISHEET=ISHEET+1
      GO TO 107
C
C INCREMENT MAP
C
108 CONTINUE
      IF (MAP.EQ.9) GO TO 109
      ISHEET=1
      JSHEET=1
      M=1
      L=8
      LN=9
      MAP=MAP+1
      GO TO 107
C
C TRANSFER POINT FOR WRITE
C
118 CALL OUT3A (MAP,M,L,MPL1,MPL2,MPL3,LMI7,ISHEET,JSHEET,MPL4,LPL1,A)

```

```

OUT3 1860
OUT3 1870
OUT3 1880
OUT3 1890
OUT3 1900
OUT3 1910
OUT3 1920
OUT3 1930
OUT3 1940
OUT3 1950
OUT3 1960
OUT3 1970
OUT3 1980
OUT3 1990
OUT3 2000
OUT3 2010
OUT3 2020
OUT3 2030
OUT3 2040
OUT3 2050
OUT3 2060
OUT3 2070
OUT3 2080
OUT3 2090
OUT3 2100
OUT3 2110
OUT3 2120
OUT3 2130
OUT3 2140
OUT3 2150
OUT3 2160
OUT3 2170
OUT3 2180
OUT3 2190
OUT3 2200
OUT3 2210
OUT3 2220
OUT3 2230
OUT3 2240
OUT3 2250
OUT3 2260

```



```

GO TO 105
109 RETURN
END
SUBROUTINE OUT3A (MAP,M,L,MPL1,MPL2,MPL3,LMI7,ISHEET,JSHEET,
2          MPL4,LPL1,A)
C
C *****
C ***** SUBROUTINE OUT3A *****
C *****
C
C HIGHEST STATEMENT NUMBER IS 135
C OTHER NUMBERS AVAILABLE ARE 125 THRU 130
C
C DECLARE COMMON VARIABLES
C
COMMON      H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),
2          BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),
3          KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),
4          I, J, IMAX, JMAX, ITER, CNTER, ERINF, IMAXM1, JMAXM1, XP
REAL KX,KY
INTEGER CNTER
C
C DECLARE OUT3A VARIABLES
C
DIMENSION A(4,16)
C
C WRITE MAPS
C
WRITE (6,131)
131 FORMAT ('1',////////)
IF (MAP.EQ.5.OR.MAP.EQ.6.OR.MAP.EQ.7.OR.MAP.EQ.8.OR.MAP.EQ.9)
2 GO TO 124
C
C WRITE NODE MAPS
C
C WRITE FIRST LINE
C
IF (MAP.EQ.1) WRITE (6,100) M,MPL3,LMI7,L
100 FORMAT (' ',6X,'ENTERED HEAD AT NODES (BH) - I FROM ',I2,' TO ',I2
2,' - J FROM ',I2,' TO ',I2,23X,'MAP 1')
IF (MAP.EQ.2) WRITE (6,111) M,MPL3,LMI7,L

```

```

OUT3  2270
OUT3  2280
OUT3  2290
OT3A  0010
OT3A  0020
OT3A  0030
OT3A  0040
OT3A  0050
OT3A  0060
OT3A  0070
OT3A  0080
OT3A  0090
OT3A  0100
OT3A  0110
OT3A  0120
OT3A  0130
OT3A  0140
OT3A  0150
OT3A  0160
OT3A  0170
OT3A  0180
OT3A  0190
OT3A  0200
OT3A  0210
OT3A  0220
OT3A  0230
OT3A  0240
OT3A  0250
OT3A  0260
OT3A  0270
OT3A  0280
OT3A  0290
OT3A  0300
OT3A  0310
OT3A  0320
OT3A  0330
OT3A  0340
OT3A  0350
OT3A  0360
OT3A  0370
OT3A  0380

```

111	FORMAT (' ',6X,'FINAL HEAD AT NODES (H) - I FROM ',12,' TO ',12,'	OT3A	0390
	2- J FROM ',12,' TO ',12,26X,'MAP 2')	OT3A	0400
	IF (MAP.EQ.3) WRITE (6,112) M,MPL3,LMI7,L	OT3A	0410
112	FORMAT (' ',6X,'ENTERED RECHARGE AT NODES (BFR) - I FROM ',12,' TO	OT3A	0420
	2 ',12,' - J FROM ',12,' TO ',12,18X,'MAP 3')	OT3A	0430
	IF (MAP.EQ.4) WRITE (6,113) M,MPL3,LMI7,L	OT3A	0440
113	FORMAT (' ',6X,'FINAL RECHARGE AT NODES (FR) - I FROM ',12,' TO ',	OT3A	0450
	212,' - J FROM ',12,' TO ',12,21X,'MAP 4')	OT3A	0460
C		OT3A	0470
C	WRITE SECOND LINE	OT3A	0480
C		OT3A	0490
	WRITE (6,101) ISHEET,JSHEET	OT3A	0500
	101 FORMAT (' ',85X,'SHEET ',12,'-',12)	OT3A	0510
C		OT3A	0520
C	WRITE THIRD LINE	OT3A	0530
C		OT3A	0540
	IF (MAP.EQ.1) WRITE (6,102)	OT3A	0550
102	FORMAT (' ',11X,'(12345 INDICATES HEAD SPECIFIED AS ZERO, ZERO IND	OT3A	0560
	2ICATES NO HEAD SPECIFIED)')	OT3A	0570
	IF (MAP.EQ.2) WRITE (6,103)	OT3A	0580
	IF (MAP.EQ.3) WRITE (6,114)	OT3A	0590
114	FORMAT (' ',11X,'(12345 INDICATES ZERO SPECIFIED, ZERO IS UNSPECIF	OT3A	0600
	2IED, FLOW OUT IS NEGATIVE)')	OT3A	0610
	IF (MAP.EQ.4) WRITE (6,115)	OT3A	0620
115	FORMAT (' ',11X,'(FLOW OUT OF SYSTEM IS NEGATIVE)')	OT3A	0630
C		OT3A	0640
C	WRITE HEADINGS	OT3A	0650
C		OT3A	0660
	WRITE (6,103)	OT3A	0670
103	FORMAT (' ')	OT3A	0680
	WRITE (6,104) M,MPL1,MPL2,MPL3,MPL4	OT3A	0690
104	FORMAT (' ',6X,'J=',4X,'I= ',12,17X,12,17X,12,17X,12,17X,12)	OT3A	0700
	WRITE (6,103)	OT3A	0710
	WRITE (6,105) LPL1,LPL1	OT3A	0720
105	FORMAT (' ',6X,12,8X,'*',18X,'*',18X,'*',18X,'*',18X,'*',12)	OT3A	0730
	WRITE (6,106)	OT3A	0740
106	FORMAT (' ')	OT3A	0750
C		OT3A	0760
C	WRITE DATA	OT3A	0770
C		OT3A	0780
	DO 109 N=1,8	OT3A	0790

```

IN=IABS(L+1-N)
WRITE (6,107)
107 FORMAT (' ',16X,'|',18X,'|',18X,'|',18X,'|')
WRITE (6,103)
WRITE (6,108) IN,A(1,N),A(2,N),A(3,N),A(4,N),IN
108 FORMAT (' ',6X,I2,' (',G14.7,') - (',G14.7,') - (',G14.7,') - (',G
214.7,') -',8X,'* ',I2)
WRITE (6,106)
109 CONTINUE
WRITE (6,110) M,MPL1,MPL2,MPL3,MPL4
110 FORMAT (' ',15X,I2,17X,I2,17X,I2,17X,I2,17X,I2)
RETURN

C
C BRANCH MAPS
C
C WRITE FIRST LINE
C
124 IF (MAP.EQ.5) WRITE (6,116) M,MPL3,LMI7,L
116 FORMAT (' ',6X,'ENTERED FLOW IN BRANCHES (BFX AND BFY) - I FROM ',
2I2,' TO ',I2,' - J FROM ',I2,' TO ',I2,11X,'MAP 5')
IF (MAP.EQ.6) WRITE (6,117) M,MPL3,LMI7,L
117 FORMAT (' ',6X,'FINAL FLOW IN BRANCHES (FX AND FY) - I FROM ',I2,'
2 TO ',I2,' - J FROM ',I2,' TO ',I2,15X,'MAP 6')
IF (MAP.EQ.7) WRITE (6,118) M,MPL3,LMI7,L
118 FORMAT (' ',6X,'ENTERED PERMEABILITY IN BRANCHES (BKX AND BKY) - I
2 FROM ',I2,' TO ',I2,' - J FROM ',I2,' TO ',I2,3X,'MAP 7')
IF (MAP.EQ.8) WRITE (6,119) M,MPL3,LMI7,L
119 FORMAT (' ',6X,'FINAL PERMEABILITY IN BRANCHES (KX AND KY) - I FRO
2M ',I2,' TO ',I2,' - J FROM ',I2,' TO ',I2,7X,'MAP 8')
IF (MAP.EQ.9) WRITE (6,120) M,MPL3,LMI7,L
120 FORMAT (' ',6X,'LENGTH (ENTERED) OF BRANCHES (BLX AND BLY) - I FRO
2M ',I2,' TO ',I2,' - J FROM ',I2,' TO ',I2,7X,'MAP 9')

C
C WRITE SECOND LINE
C
WRITE (6,101) ISHEET,JSHEET

C
C WRITE THIRD LINE
C
IF (MAP.EQ.5) WRITE (6,121)
121 FORMAT (' ',11X,'(12345 IS 0 SPECIFIED, 0 IS UNSPECIFIED, FLOW DOW

```

OT3A	0800
OT3A	0810
OT3A	0820
OT3A	0830
OT3A	0840
OT3A	0850
OT3A	0860
OT3A	0870
OT3A	0880
OT3A	0890
OT3A	0900
OT3A	0910
OT3A	0920
OT3A	0930
OT3A	0940
OT3A	0950
OT3A	0960
OT3A	0970
OT3A	0980
OT3A	0990
OT3A	1000
OT3A	1010
OT3A	1020
OT3A	1030
OT3A	1040
OT3A	1050
OT3A	1060
OT3A	1070
OT3A	1080
OT3A	1090
OT3A	1100
OT3A	1110
OT3A	1120
OT3A	1130
OT3A	1140
OT3A	1150
OT3A	1160
OT3A	1170
OT3A	1180
OT3A	1190
OT3A	1200

2N OR TO LEFT IS NEGATIVE)')	OT3A	1210
IF (MAP.EQ.6) WRITE (6,122)	OT3A	1220
122 FORMAT (' ',11X, '(FLOW DOWN OR TO LEFT IS NEGATIVE)')	OT3A	1230
IF (MAP.EQ.7) WRITE (6,123)	OT3A	1240
123 FORMAT (' ',11X, '(12345 IS ZERO SPECIFIED, ZERO IS UNSPECIFIED)')	OT3A	1250
IF (MAP.EQ.8.OR.MAP.EQ.9) WRITE (6,103)	OT3A	1260
C	OT3A	1270
C WRITE HEADINGS	OT3A	1280
C	OT3A	1290
WRITE (6,103)	OT3A	1300
WRITE (6,104) M, MPL1, MPL2, MPL3, MPL4	OT3A	1310
WRITE (6,103)	OT3A	1320
WRITE (6,105) LPL1, LPL1	OT3A	1330
WRITE (6,132)	OT3A	1340
132 FORMAT (' ')	OT3A	1350
C	OT3A	1360
C WRITE DATA	OT3A	1370
C	OT3A	1380
DO 133 N=1,8	OT3A	1390
IN=IABS(L+1-N)	OT3A	1400
WRITE (6,134) A(1,N), A(2,N), A(3,N), A(4,N)	OT3A	1410
134 FORMAT (' ',9X, '(',G14.7,') (',G14.7,') (',G14.7,') (',G14.7,	OT3A	1420
2,')')	OT3A	1430
WRITE (6,103)	OT3A	1440
WRITE (6,135) IN, A(1,N+8), A(2,N+8), A(3,N+8), A(4,N+8), IN	OT3A	1450
135 FORMAT (' ',6X, I2,8X, '* (',G14.7,') * (',G14.7,') * (',G14.7,') *	OT3A	1460
2(',G14.7,') * ', I2)	OT3A	1470
WRITE (6,132)	OT3A	1480
133 CONTINUE	OT3A	1490
WRITE (6,110) M, MPL1, MPL2, MPL3, MPL4	OT3A	1500
RETURN	OT3A	1510
END	OT3A	1520
SUBROUTINE OUT3B (KK, FERROR, HERROR, MAXIT, KNTR)	OT3B	0010
C	OT3B	0020
C *****	OT3B	0030
C ***** SUBROUTINE OUT3B *****	OT3B	0040
C *****	OT3B	0050
C	OT3B	0060
C HIGHEST STATEMENT NUMBER IS 112	OT3B	0070
C	OT3B	0080
C DECLARE COMMON VARIABLES	OT3B	0090

```

C
COMMON H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),
2 BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),
3 KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),
4 I, J, IMAX, JMAX, ITER, CNTR, ERINF, IMAXM1, JMAXM1, XP
REAL KX, KY
INTEGER CNTR

C
C WRITE HEADINGS
C
KK=KK+1
IF (KK.NE.1) GO TO 112
WRITE (6,100)
100 FORMAT ('1',/////////)
WRITE (6,101)
101 FORMAT (' ',6X,'OUTPUT WITH SUBROUTINE OUT3')
WRITE (6,102)
102 FORMAT (' ')
ITPL1=ITER+1
IF (CNTR.NE.0.OR.KNTR.NE.0) GO TO 107
WRITE (6,103) FERROR
103 FORMAT (' ',11X,'ALL FLOW ERRORS LESS THAN VALUE SPECIFIED (' ,G10.
23,') AND ALL HEAD ERRORS LESS THAN')
WRITE (6,104) HERROR,ITPL1
104 FORMAT (' ',11X,'VALUE SPECIFIED (' ,G10.3,'). VALUES OF THESE ERRO
2RS FOR FOLLOWING ITERATION (' ,I2,') ARE')
WRITE (6,105)
105 FORMAT (' ',11X,'LISTED BELOW.')
GO TO 106
107 CONTINUE
WRITE (6,108) FERROR
108 FORMAT (' ',11X,'SOME FLOW ERRORS EXCEEDED VALUE SPECIFIED (' ,G10.
23,'); AND/OR SOME HEAD ERRORS')
WRITE (6,109) HERROR
109 FORMAT (' ',11X,'EXCEEDED VALUE SPECIFIED (' ,G10.3,') ON ITERATION
2 PRECEDING MAXIMUM SPECIFIED. VALUES')
WRITE (6,110) MAXIT
110 FORMAT (' ',11X,'OF THESE ERRORS FOR THE MAXIMUM SPECIFIED ITERATI
2ON (' ,I5,') ARE LISTED BELOW')
106 WRITE (6,102)
WRITE (6,111)

```

```

OT38 0100
OT38 0110
OT38 0120
OT38 0130
OT38 0140
OT38 0150
OT38 0160
OT38 0170
OT38 0180
OT38 0190
OT38 0200
OT38 0210
OT38 0220
OT38 0230
OT38 0240
OT38 0250
OT38 0260
OT38 0270
OT38 0280
OT38 0290
OT38 0300
OT38 0310
OT38 0320
OT38 0330
OT38 0340
OT38 0350
OT38 0360
OT38 0370
OT38 0380
OT38 0390
OT38 0400
OT38 0410
OT38 0420
OT38 0430
OT38 0440
OT38 0450
OT38 0460
OT38 0470
OT38 0480
OT38 0490
OT38 0500

```


111	FORMAT (' ',11X,'ENTERED AND FINAL VALUES OF VARIABLES FOLLOW THE ZERROR LISTING')	OT38	0510
112	RETURN	OT38	0520
	END	OT38	0530
	SUBROUTINE OUT3C (KNTR,HEDERX,HEDERY)	OT38	0540
C		OT3C	0010
C	*****	OT3C	0020
C	***** SUBROUTINE OUT3C *****	OT3C	0030
C	*****	OT3C	0040
C		OT3C	0050
C	HIGHEST STATEMENT NUMBER IS 111	OT3C	0060
C		OT3C	0070
C	DECLARE COMMON VARIABLES	OT3C	0080
C		OT3C	0090
	COMMON H(38,27), BH(38,27), FX(38,27), BFX(38,27), FY(38,27),	OT3C	0100
2	BFY(38,27), FR(38,27), BFR(38,27), KX(38,27), BKX(38,27),	OT3C	0110
3	KY(38,27), BKY(38,27), BLX(38,27), BLY(38,27), IA(38,27),	OT3C	0120
4	I, J, IMAX, JMAX, ITER, CINTER, ERINF, IMAXM1, JMAXM1, XP	OT3C	0130
	REAL KX, KY	OT3C	0140
	INTEGER CINTER	OT3C	0150
C		OT3C	0160
C	WRITE HEADING ON FIRST SHEET	OT3C	0170
C		OT3C	0180
	IF (I.EQ.1.AND.J.EQ.1) LPK=1	OT3C	0190
	IF (I.EQ.1.AND.J.EQ.1) LPLIM=2	OT3C	0200
	IF (I.NE.1.OR.J.NE.1) GO TO 100	OT3C	0210
	WRITE (6,101)	OT3C	0220
101	FORMAT ('0')	OT3C	0230
	WRITE (6,102)	OT3C	0240
102	FORMAT ('')	OT3C	0250
100	IF (LPK.NE.LPLIM) GO TO 103	OT3C	0260
	WRITE (6,104)	OT3C	0270
104	FORMAT ('1',//////////)	OT3C	0280
103	IF (I.EQ.1.AND.J.EQ.1) GO TO 111	OT3C	0290
	IF (LPK.EQ.LPLIM) GO TO 111	OT3C	0300
	GO TO 105	OT3C	0310
111	LPK=1	OT3C	0320
	LPLIM=38	OT3C	0330
	IF (I.EQ.1.AND.J.EQ.1) LPLIM=28	OT3C	0340
	WRITE (6,106)	OT3C	0350
106	FORMAT (' ',40X,'CUM. FLOW',36X,'CUM. HEAD')	OT3C	0360
		OT3C	0370

WRITE (6,107)	OT3C 0380
107 FORMAT (' ',15X,'I',6X,'J',9X,'FLOW',4X,'ERRORS GT',7X,'X BRANCH',	OT3C 0390
28X,'Y BRANCH ERRORS GT')	OT3C 0400
WRITE (6,108)	OT3C 0410
108 FORMAT (' ',32X,'ERROR',5X,'LIMIT',8X,'HEAD ERROR',6X,'HEAD ERROR.	OT3C 0420
2 LIMIT')	OT3C 0430
WRITE (6,102)	OT3C 0440
105 IF (J.NE.1) GO TO 109	OT3C 0450
WRITE (6,102)	OT3C 0460
LPK=LPK+1	OT3C 0470
109 WRITE (6,110) I,J,ERINF,CNTER,HEDERX,HEDERY,KNTR	OT3C 0480
110 FORMAT (' ',14X,I2,5X,I2,4X,G14.7,2X,I3,7X,G14.7,2X,G14.7,5X,I3)	OT3C 0490
LPK=LPK+1	OT3C 0500
RETURN	OT3C 0510
END	OT3C 0520
SUBROUTINE OUT4	OUT4 0010
C ***** SUBROUTINE OUT4 *****	OUT4 0020
C *****	OUT4 0030
C *****	OUT4 0040
C *****	OUT4 0050
C	OUT4 0060
C HIGHEST STATEMENT NUMBER IS 103	OUT4 0070
C	OUT4 0080
C DECLARE COMMON VARIABLES	OUT4 0090
C	OUT4 0100
COMMON H(38,27), BH(38,27), FX(38,27),BFX(38,27), FY(38,27),	OUT4 0110
2 BFY(38,27), FR(38,27),BFR(38,27), KX(38,27),BKX(38,27),	OUT4 0120
3 KY(38,27),BKY(38,27),BLX(38,27),BLY(38,27), IA(38,27);	OUT4 0130
4 I,J,IMAX,JMAX,ITER,CNTER,ERINF,IMAXM1,JMAXM1,XP	OUT4 0140
REAL KX,KY	OUT4 0150
INTEGER CNTER	OUT4 0160
C	OUT4 0170
C PUNCH CARDS	OUT4 0180
C	OUT4 0190
NCARD=IMAX*JMAX	OUT4 0200
WRITE (7,100) NCARD	OUT4 0210
100 FORMAT (I10)	OUT4 0220
DO 101 I=1,IMAX	OUT4 0230
DO 102 J=1,JMAX	OUT4 0240
WRITE (7,103) I,J,H(I,J),FX(I,J),FY(I,J),FR(I,J),KX(I,J),KY(I,J)	OUT4 0250
103 FORMAT (2I5,6E10.3)	OUT4 0260

102 CONTINUE
101 CONTINUE
RETURN
END

OUT4 0270
OUT4 0280
OUT4 0290
OUT4 0300

