Department of Hydraulics and Catchment Hydrology,

Agricultural University, Wageningen, 1987

SWM II - NUMERICAL MODEL OF TWO-DIMENSIONAL FLOW IN A VARIABLY SATURATED POROUS MEDIUM

T. Vogel

Ing. Tomas Vogel
Research Institute for
Improvement of Agricultural Soils
Prague, Czechoslovakia

ABSTRACT

This report contains documentation of computer program SWM II designed for numerical simulation of transient two-dimensional flow in a variably saturated porous medium.

The program can be used as a tool for detailed analysis of flow in the vicinity of various technical objects (e.g.: drains, irrigation objects, waste dumps, ...) or for simulation of water regime in soils under the natural atmospheric and groundwater conditions. The program includes procedures for simulation of water uptake by plant roots.

ACKNOWLEDGMENTS

This is the report of the work done during my 3 month stay at the Department of Hydraulics and Catchment Hydrology, Agricultural University, Wageningen, The Netherlands; within the frame of the project Spatial Variability of Hydrological and Soil Physical Characteristics led by Ir. J.N.M. Stricker.

I wish to thank Han Stricker, Jan Hopmans, Piet Warmerdam, Paul Torfs, Jacques Kole and other colleagues from the department for their understanding and kind help.

TABLE OF CONTENTS

1. INTRODUCTION

1.1. PROGRAM DEVELOPMENT

The SWM II computer program is based on numerical model developed by Shlomo P. Neuman (1973). Some subroutines of the original program UNSAT2 (Neuman et al., 1974) have been adopted most of them being more or less altered. The main program has been completely rewritten.

The program SWM II claims to be, in comparison with UNSAT2, more readable (which is motivated by the aim to facilitate possible future changes and extensions), easier to handle (namely because of clear input and output arrangement), and faster in the performance relatively to one iteration (unnecessary comunications between internal and external memories have been avoided). The list of major differences follows:

1. The restart feature necessary in UNSAT2 for implementation of time variable parameters, namely parameters of atmospheric boundary conditions and the value of time increment, has been done away. Time dependent parameters of boundary conditions are read by SWM II from separate input file at prescribed times. The time increment is automatically adjusted by the program.

2. The root zone, treated in UNSAT2 as a system of vertical one-dimensional nodal columns, is in SWM II defined as two-dimensional region without any restrictions on the shape of both the root zone itself and elements of which it is set up. The modified version of the method of Feddes et al. (1978) is used for the sink term. The method is a generalized two-dimensional analogy of the technique

introduced in one-dimensional program SWATRE (Feddes et al., 1978, Belmans et al., 1983).

3. Instead of providing tables for description of material hydraulic characteristics, analytical expressions are introduced such that only the parameters of these expressions are needed to describe fully the material properties. Additional way of coping with space variability of material hydraulic properties is introduced by defining three scaling factors associated with moisture content, pressure head and hydraulic conductivity. This provides a simple tool for generation of variable material properties from either deterministic or stochastic space law.

4. Mass balance and internal flux data, missing in UNSAT2, are computed at prescribed times.

5. Well analysis option included in UNSAT2 is not provided.

Those passages of the documentation of program UNSAT2 (Neuman, 1974, Davis and Neuman, 1983), which remain valid for program SWM II, were adopted in almost unchanged form to provide a potential user of SWM II with all necessary information in one reference manual.

1.2. OVERVIEW OF PROGRAM CAPABILITIES

The program is intended for the analysis of flow in unsaturated, partially saturated, or saturated porous media. SWM II can handle flow regions delineated by irregular boundaries and composed of nonuniform soils having arbitrary degree of local anisotropy. Flow can occur in the vertical plane, in the horizontal plane, or in a three-dimensional region exhibiting radial symmetry about a vertical axis. In addition to conventional prescribed head and flux boundaries, the program can also deal with boundaries controlled by atmospheric conditions. Time variable boundary conditions can be prescribed at specified, not necessarily equidistant, times. Water uptake by plants is computed in a manner that accounts for both soil and atmospheric conditions.

1.3. COMPUTER EQUIPMENT REQUIREMENTS

The program SWM II is intended for use on ATARI ST 1040 microcomputer. Since the program is written in ANSI standard FORTRAN-77, modifications required to allow execution on other computers should be minimal. Computer memory and offline storage are functions of problem size (for details see Chapter 4.).

2. THEORY

2.1. GOVERNING EQUATION

Consider a situation involving two-dimensional isothermal Darcian flow of water in a partially saturated rigid porous medium. Assuming that the role of the air phase is insignificant, the water flow can be described by the modified form of Richards' equation:

$$\frac{d\theta}{dh}\frac{\partial h}{\partial t} = \frac{\partial}{\partial x}\left\{ K \left[K^A_{xx}\frac{\partial h}{\partial x} + K^A_{xz}\frac{\partial h}{\partial z} + K^A_{xz} \right] \right\} + \frac{\partial}{\partial z}\left\{ K \left[K^A_{zx}\frac{\partial h}{\partial x} + K^A_{zz}\frac{\partial h}{\partial z} + K^A_{zz} \right] \right\} - S \qquad (2.1)$$

where $h(x,z,t)$ is the pressure head [L], θ is the volumetric moisture content, K is the hydraulic conductivity [L/T], S is the sink term [1/T], x and z are spatial coordinates [L] (z-coordinate is assumed to be vertically upward), and t is time [T]. The hydraulic properties of porous medium are expressed by the pair of hydraulic characteristics $K(h)$ and $\theta(h)$. The derivative $d\theta/dh$ is refered to as a specific moisture capacity, $C(h)$. The hydraulic conductivity is given by:

$$\mathbf{K} = K(h,x,z)\, K_A \quad , \qquad K(h,x,z) = K_s(x,z)\, K_r(h,x,z) \qquad (2.2)$$

where $\mathbf{K}$ is the hydraulic conductivity tensor, K_r is the relative hydraulic conductivity function (assumed to be direction independent), K_s is the reference value of saturated hydraulic conductivity, and K_A is a dimensionless tensor which describes conductivity anisotropy relatively to the value of K_s.

2.2 SINK TERM

The sink term, S, is the volume of water removed per unit time from a unit volume of soil due to plant transpiration.

Feddes et al., (1978) defined the sink term, S, as:

$$S(h) = \alpha(h)\ S^* \qquad (2.3)$$

where $\alpha(h)$ is the prescribed dimensionless function of soil water pressure head ($\alpha \in \langle 0,1 \rangle$), and S^* is an intensity of the potential transpiration demand [1/T] in the root zone. The assumption that the potential transpiration demand is equally distributed over the two-dimensional rectangular root zone leads to:

$$S^* = \frac{1}{L_x\ L_z}\ L_T\ T^* \qquad (2.4)$$

where T^* is the potential transpiration rate [L/T], L_z is the depth [L] of the root zone, L_x is the width [L] of the root zone, and L_T is the length [L] of the soil surface associated with transpiration (if $L_T=L_x$ then (2.4) reduces to $S^*=T^*/L_z$).

Following generalisation of equation (2.4) can be introduced for a non-uniform distribution of potential transpiration demand in an irregulary shaped root zone:

$$S^* = \beta(x,z)\ L_T\ T^* \qquad (2.5)$$

where $\beta(x,z)$ is the normalized distribution function [$1/L^2$] which describes the space variations of the potential extraction

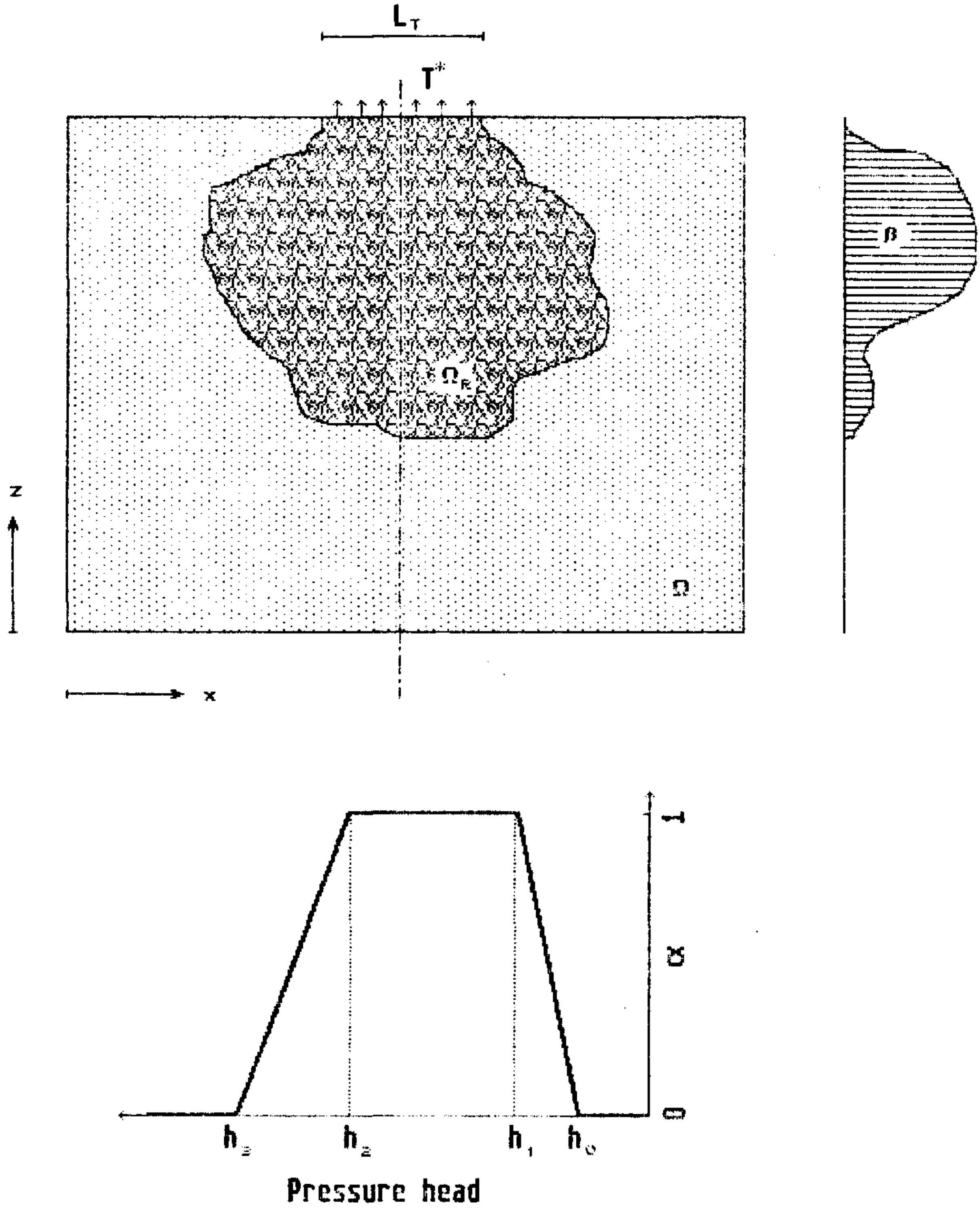


Figure 2.1. Space distribution of transpiration demand in the root zone as described by function $\beta(x,z)$ and the assumed shape of reduction function $\alpha(h)$.

intensity, S^*, in the root zone. The normalization procedure is defined by:

$$\beta(x,z) = \beta'(x,z) \; / \int_{\Omega_R} \beta'(x,z)\, d\Omega \tag{2.6}$$

where Ω_R is the region occupied by the root zone, and $\beta'(x,z)$ is an arbitrary prescribed distribution function. It is obvious that $\beta(x,z)$ normalised by (2.6) fulfils the following condition:

$$\int_{\Omega_R} \beta(x,z)\, d\Omega = 1 \tag{2.7}$$

and thus for S^* the condition holds that:

$$\frac{1}{L_T} \int_{\Omega_R} S^*\, d\Omega = T^* \tag{2.8}$$

Finally the intensity of actual transpiration demand (2.3) can be written in the form:

$$S(h,x,z) = \alpha(h,x,z)\; \beta(x,z)\; L_T\; T^* \tag{2.9}$$

and actual transpiration rate is given by:

$$T = \frac{1}{L_T} \int_{\Omega_R} S\, d\Omega = T^* \int_{\Omega_R} \alpha(h,x,z)\; \beta(x,z)\; d\Omega \tag{2.10}$$

2.3. INITIAL AND BOUNDARY CONDITIONS

Initial condition

To solve Eq. 2.1 it is necessary to know the initial distribution of pressure head within the flow region Ω:

$$h(x,z,0) = h_o(x,z) \qquad \text{for } (x,z) \in \Omega \tag{2.11}$$

where h_o is a prescribed function of x and z.

Boundary conditions

The basic boundary conditions, which describe system-independent interactions on the boundary of the flow region, are specified pressure head (Dirichlet type) boundary condition:

$$h(x,z,t) = \psi(x,z,t) \qquad \text{for } (x,z) \in \Gamma_D \tag{2.12}$$

and specified flux (Neumann type) boundary condition:

$$v_x \, \nu_x + v_z \, \nu_z = -\varsigma(x,z,t) \qquad \text{for } (x,z) \in \Gamma_N \tag{2.13}$$

where Γ_D and Γ_N are the appropriate parts of the boundary, ψ and ς are prescribed functions of x, z and t, ν_x and ν_z are components of an outward unit vector normal to the boundary Γ_N, v_x and v_z are components of Darcian velocity defined as:

$$\begin{aligned} v_x &= -K \left(\hat{K}_{xx} \frac{\partial h}{\partial x} + \hat{K}_{xz} \frac{\partial h}{\partial z} + \hat{K}_{xz} \right) \\ v_z &= -K \left(\hat{K}_{zx} \frac{\partial h}{\partial x} + \hat{K}_{zz} \frac{\partial h}{\partial z} + \hat{K}_{zz} \right) \end{aligned} \tag{2.14}$$

In addition to the basic types of boundary conditions, the flow region may be bounded by the system-dependent boundary conditions for which the a priori specification is not possible.

On the boundaries that correspond to soil-air interfaces and are thus exposed to atmospheric conditions, the potential flux is controlled by external conditions while the actual flux depends on antecedent soil moisture conditions. On these boundaries, conditions may change from prescribed flux to prescribed head type. In the absence of surface ponding, the solution is obtained by minimizing the absolute value of the flux (while maintaining the appropriate sign) subject to the following requirements (Neuman et al., 1974):

$$| v_x \nu_x + v_z \nu_z | \leq E^* \tag{2.15}$$

and

$$h \in \langle h_A , 0 \rangle \tag{2.16}$$

where E^* is the maximum potential rate of infiltration or evaporation under the prevailing atmospheric conditions, h is the actual surface pressure head, and h_A is the minimum pressure head allowed under the prevailing soil conditions (determined from the equilibrium conditions between soil water and atmospheric water vapor). When either end point in (2.16) is reached, a prescribed head boundary conditions is used to calculate the actual surface flux. Methods of calculating E^* and h_A on the basis of atmospheric data have been discussed by Feddes et al. (1974).

Another type of a system-dependent boundary condition that is considered is the seepage face. In this case the length of the seepage face is unknown a priori. Along a seepage face, water leaves the saturated part of the flow region into the atmosphere and the pressure head is thus uniformly equal to zero (atmospheric).

2.4. SOIL HYDRAULIC PROPERTIES

Analytical expressions for soil hydraulic characteristics

Instead of providing tables that describe the soil hydraulic characteristics of each material, analytical expressions are used such that only the parameters of these expressions are needed to describe fully soil hydraulic characteristics.

Modified van Genuchten's (1980) closed form approach based on capillary model of Mualem (1976) is used to fit experimentally determined water retention data and to predict the relative hydraulic conductivity (Vogel and Císlerová 1988). Both hydraulic characteristics are determined by a set of 9 parameters, θ_r, θ_s, θ_a, θ_m, α, n, K_s, K_k, and θ_k:

$$\theta(h) = \theta_a + (\theta_m - \theta_a)\,[\,1 + (-\alpha h)^n\,]^{-m} \qquad \text{for } h < h_s \qquad (2.17)$$

$$\theta(h) = \theta_s \qquad \text{for } h \geq h_s$$

$$K(h) = K_s\, K_r(\theta(h)) \qquad \text{for } h \leq h_k \qquad (2.18)$$

$$K(h) = K_k + (h - h_k)(K_s - K_k)/(h_s - h_k) \qquad \text{for } h \in (h_k, h_s)$$

$$K(h) = K_s \qquad \text{for } h \geq h_s$$

where

$$K_r(\theta) = \left(\frac{\Theta}{\Theta_k}\right)^{1/2} \left[\frac{F(\theta_r) - F(\theta)}{F(\theta_r) - F(\theta_k)}\right]^2 \frac{K_k}{K_s} \qquad (2.19)$$

$$F(\theta) = \left[\,1 - \left(\frac{\theta - \theta_a}{\theta_m - \theta_a}\right)^{1/m}\,\right]^m \qquad (2.20)$$

$$m = 1 - 1/n \quad , \quad n > 1$$

$$\Theta = (\theta - \theta_r)/(\theta_s - \theta_r)$$

$$\Theta_k = (\theta_k - \theta_r)/(\theta_s - \theta_r)$$

and where θ_r and θ_s denote residual and saturated moisture content, respectively, K_s is the saturated hydraulic conductivity. To increase flexibility of used analytical expressions and to allow a non-zero air-entry value, h_s, parameters θ_r and θ_s are replaced by the parameters $\theta_a \leq \theta_r$ and $\theta_m \leq \theta_s$ in the fitting process. Physical meaning of θ_r and θ_s, as measurable quantities, is preserved. The measured value of unsaturated hydraulic conductivity, $K_k \leq K_s$ (corresponding to the moisture content $\theta_k \leq \theta_s$) is used instead of K_s to obtain the desired $K(\theta)$ from predicted $K_r{}^*(\theta)$. To fulfil the original definition of $K_r(\theta)$, as $K_r(\theta) = K(\theta)/K_s$, the expression for $K_r{}^*(\theta)$ is multiplied by K_k/K_s which leads to the equation (2.19).

If $\theta_a=\theta_r$, $\theta_m=\theta_k=\theta_s$, and $K_k=K_s$ then equations (2.17)-(2.20) reduce to the original van Genuchten's model (1980):

$$\theta(h) = \theta_r + (\theta_s-\theta_r) \left[1 + (-\alpha h)^n \right]^{-m} \quad \text{for } h < 0 \qquad (2.21)$$

$$\theta(h) = \theta_s \quad \text{for } h \geq 0$$

$$K(h) = K_s\, K_r(\theta(h)) \quad \text{for } h < 0 \qquad (2.22)$$

$$K(h) = K_s \quad \text{for } h \geq 0$$

where

$$K_r(\theta) = \Theta^{1/2} \left[1 - (1 - \Theta^{1/m})^m \right]^2 \qquad (2.23)$$

Van Genuchten (1980) developed a nonlinear least-squares curve-fitting procedure to estimate the parameters θ_r, θ_s, α, and n from measured $\theta(h)$-data. A later version of this model (RETC) allows the expressions (2.21) and (2.23) to be fitted simultaneously to observed water retention and conductivity data.

Scaling Soil Hydraulic Properties

The purpose of scaling is to simplify the description of space variation of soil hydraulic properties. It is assumed that a set of scale factors, relating the soil hydraulic characteristics, $\theta(h)$ and $K(h)$, at any location to hydraulic characteristics of reference material, can be used to approximate the real spatial variability of a given area. The technique is based on similar media concept introduced by Miller and Miller (1956) for porous media which differ only in the scale of their internal geometry. The concept was extended by Simmons et al. (1979) on materials which differ in morphological properties but show 'scale similar' soil hydraulic characteristics.

For the purposes of numerical modelling, three scaling factors are introduced here. This allows to define the linear model of space variability of soil hydraulic properties in the form:

$$\begin{aligned} h &= \alpha_H \, h^* \\ K(h) &= \alpha_K \, K^*(h^*) \qquad (2.24) \\ \theta(h) &= \theta_r + \alpha_\theta \, [\, \theta^*(h^*) - \theta_r^* \,] \end{aligned}$$

where α_H, α_K and α_θ are mutually independent space variable scaling factors, $K^*(h^*)$ and $\theta^*(h^*)$ are space invariant hydraulic characteristics of reference material.

3. NUMERICAL APPROACH

3.1. SPACE AND TIME DISCRETIZATION

Space discretization

To obtain a solution to Eq. (2.1) subject to appropriate initial and boundary conditions, the Galerkin finite element method with linear basis functions is used. Since the Galerkin approximation process is standard and has already been covered elsewhere (Neuman, 1975), only the basic assumptions and results of the method are repeated here.

The flow region is divided into network of triangular elements. The corners of these elements are taken to be nodal points. The dependent variable, pressure head function $h(x,z,t)$, is approximated by a function $h'(x,z,t)$ defined as:

$$h'(x,z,t) = \sum_{n=1}^{N} \phi_n(x,z)\, h_n(t) \qquad (3.1)$$

where ϕ_n are piecewise linear basis functions satisfying the condition $\phi_n(x_m,z_m)=\delta_{nm}$, δ_{nm} is Kronecker delta ($\delta_{nm}=1$ if $m=n$, and $\delta_{nm}=0$ if $m\neq n$), h_n are the unknown coefficients which represent solution of (2.1) at nodal points, and N is the total number of nodal points.

By applying Galerkin finite element procedure, one obtains a system of time-dependent ordinary differential equations with non-linear coefficients:

$$\sum_{m=1}^{N} A_{nm} + \sum_{m=1}^{N} F_{nm} \frac{dh_m}{dt} = Q_n - B_n - D_n \qquad n=1,2,\ldots,N \qquad (3.2a)$$

where, for a vertical cross-section described by the coordinates x and z:

$$A_{nm} = \sum_{E} \frac{\varkappa}{4A_E} \bar{K} \left[\hat{K}_{xx} b_m b_n + \hat{K}_{xz} (c_m b_n + b_m c_n) + \hat{K}_{zz} c_n c_m \right] \tag{3.2b}$$

$$B_n = \sum_{E} \frac{\varkappa}{2} \bar{K} \left(\hat{K}_{xz} b_n + \hat{K}_{zz} c_n \right) \tag{3.2c}$$

$$F_{nm} = \sum_{E} \frac{\varkappa A_E}{12} \left(3\bar{C} + C_n \right) \delta_{nm} \tag{3.2d}$$

$$Q_n = -\left(\bar{\rho}_n \bar{\lambda}_n + \overset{+}{\rho}_n \overset{+}{\lambda}_n \right) \tag{3.2e}$$

$$D_n = \sum_{E} \frac{\varkappa A_E}{12} \left(3\bar{S}_E + S_n \right) \tag{3.2f}$$

and where:

$$b_i = z_j - z_k \qquad c_i = x_k - x_j$$
$$b_j = z_k - z_i \qquad c_j = x_i - x_k$$
$$b_k = z_i - z_j \qquad c_k = x_j - x_i$$

$$A_E = (c_k b_j - c_j b_k)/2$$

$$\bar{K} = (K_i + K_j + K_k)/3$$

$$\bar{C} = (C_i + C_j + C_k)/3$$

$$\bar{S} = (S_i + S_j + S_k)/3$$

$\varkappa = 1$ for plane flow

$\varkappa = 2\pi \bar{x}$ for axisymmetric flow, $\bar{x} = (x_i + x_j + x_k)/3$

$$\bar{\lambda}_n = \bar{L}_n/2, \qquad \overset{+}{\lambda}_n = \overset{+}{L}_n/2 \qquad \text{for plane flow}$$

$$\bar{\lambda}_n = \bar{L}_n \, \pi(\bar{x}_n + 2x_n)/3, \qquad \overset{+}{\lambda}_n = \overset{+}{L}_n \, \pi(\overset{+}{x}_n + 2x_n)/3 \qquad \text{for axisymmetric flow}$$

In the above equations, the subscripts i, j ,and k represent the three corners of the triangle element, E, A_E is the area of element E. The values of $\bar{K}$, $\bar{C}$, and $\bar{S}$ are average hydraulic conductivity, moisture capacity and root zone exraction intensity over the element E, $\bar{q}_n$ and $\overset{+}{q}_n$ are velocities [L/T] of flow across the boundary sides of elements connected to boundary node n, $\bar{L}_n$ and $\overset{+}{L}_n$ are lengths of appropriate boundary segments, $\bar{x}_n$ and $\overset{+}{x}_n$ are x-coordinates of boundary nodes contiguous with node n. It is assumed that the velocities, q, are uniform over the boundary segments. The components of vector Q_n are zero at all internal nodes which do not act as sources or sinks.

When deriving Equations (3.2), two following important assumptions have been used (Neuman, 1973) in addition to those involved in Galerkin finite element approach. First, the time derivatives of the nodal values of pressure head in Eq. (3.2) are weighted according to:

$$\frac{dh_n}{dt} = \left(\sum_E \int_{\Omega_E} C \frac{dh}{dt} \phi_n \, d\Omega \right) \Big/ \left(\sum_E \int_{\Omega_E} C \phi_n \, d\Omega \right) \qquad (3.3)$$

where Ω_E is the domain of element E. This equivalent to the so-called mass-lumping process improves the convergence of the method. Second, within each element, E, the anisotropy tensor, K_A, is assumed to be constant while the hydraulic conductivity function, K(h,x,z), specific moisture capacity, C(h,x,z), and root extraction intensity S(h,x,z), are assumed to vary, at a given instant of time, linearly according to:

$$\begin{aligned} K(x,z) &= \sum_{i=1}^{3} K(h_i,x_i,z_i)\ \phi_i(x,z) \\ C(x,z) &= \sum_{i=1}^{3} C(h_i,x_i,z_i)\ \phi_i(x,z) \qquad \text{for } (x,z) \in \Omega_E \qquad (3.4) \\ S(x,z) &= \sum_{i=1}^{3} S(h_i,x_i,z_i)\ \phi_i(x,z) \end{aligned}$$

where i stands for the corners of element E. The advantage of this approach is that no numeric integration is needed for evaluation of coefficients in Eq. (3.2).

Time discretization

Integration of Eq. (3.2) is achieved by discretizing the time domain into a sequence of finite intervals and replacing the time derivatives by finite differences. An implicit (backward difference) scheme is applied for both saturated and unsaturated conditions:

$$\sum_{m=1}^{N} \left[A_{nm}(t_j) + \frac{F_{nm}}{\Delta t_j} \right] h_m(t_j) = \sum_{m=1}^{N} \frac{F_{nm}}{\Delta t_j} h_m(t_{j-1}) + Q_n(t_j) - B_n(t_j) - D_n(t_j)$$

$$n=1,2,...,N \qquad (3.5)$$

The specific moisture capacity term F_{nm} is evaluated at half the time step using time averaged values of pressure head:

$$\bar{h}_n = [\ h_n(t_j) - h_n(t_{j-1})\ ]/2 \qquad (3.6)$$

This leads to less oscillations and generally a faster convergence.

3.2. SOLUTION STRATEGY

Iterative process

Due to the nonlinear nature of Eq. (3.5), an iterative process must be used at each time level to obtain solution of (3.5). For each iteration a system of linearized algebraic equations is derived from (3.5) and, after incorporation of boundary conditions, solved using Gaussian elimination. The coefficients in Eq. (3.5) are then re-evaluated and the new equations are again solved. This process is repeated until a satisfactory degree of convergence is obtained for the time level (i.e., until the absolute change in pressure heads between two successive iterations becomes less than the given value of pressure head tolerance). First estimate (zero iteration) of unknown pressure heads at any time level, necessary for evaluation of the coefficients in Eq. (3.5), is obtained using final pressure head values from the previous time level.

Time control

Three different time discretizations are introduced:

I. Time discretization associated with numerical solution.

II. Time discretization associated with specification of boundary conditions.

III. Time discretization associated with output print of nodal point, water balance, and flux information.

The Discretizations II. and III. are mutually independent, with generally variable time increments determined via input data. The Discretization I. starts with prescribed initial time increment, Δt. The time increment is then automatically adjusted at each time level according to following rules (Mls, 1982):

1. The Discretization I. must coincide with Discretiztions II. and III.

2. $\Delta t \in (\Delta t_{min}, \Delta t_{max})$.

3. If a number of iterations, necessary at any time level to meet convergence criterion, is less than or equal to 3, the time increment, Δt, is increased by multiplying by a chozen constant >1 (usually 1.1 - 1.5). If the number of iterations is greater than or equal to 7, Δt is decreased by multiplying by another constant <1 (usually 0.3 - 0.9).

4. If the number of iterations at any time level becomes greater than prescribed maximum number (usually 10 - 20), the iterative process is canceled, time increment is set

equal to $\Delta t/10$, and iterations start again from the beginning.

Treatment of the pressure head boundary condition

Finite element equations corresponding to nodes at which pressure head is prescribed can, in principle, be eliminated from consideration. However, from a programming standpoint it is more convenient to replace such equations by the dummy expression (Neuman, 1974):

$$\sum_{m=1}^{N} \delta_{nm} h_m = h_p \tag{3.7}$$

where δ_{nm} is Kronecker delta, h_p is the prescribed value of pressure head at node n. The values of h_n in all other equations are set equal to h_p and the respective entries of left-hand-side matrix are added to the right-hand-side vector and replaced by zeroes in order to preserve matrix symmetry. After solving for all pressure heads, the value of Q_n can be calculated explicitly from the original finite element equation for node n.

Flux boundary condition

At nodes along prescribed flux boundaries, the values of Q_n are computed according to Eq. (3.2e). Internal nodes that act as Neumann type sources or sinks have values of Q_n equal to the known fluid generation or extraction rate.

Treatment of atmospheric boundary conditions and seepage faces

Atmospheric boundaries are simulated by applying either prescribed head or prescribed flux boundary conditions depending on whether or not Eq. (2.15) and (2.16) are satisfied (Neuman, 1974). If Eq. (2.16) is not satisfied, node n becomes a prescribed head boundary. If during any stage of the computations Eq. (2.15) is not satisfied, i.e., the calculated flux exceeds the specified potential flux, the node is assigned a flux equal to the potential value and is again treated as a prescribed flux boundary.

All nodes expected to be a part of a seepage face during any stage of computations must be identified a priori. During each iteration, the saturated part of potential seepage face is treated as a prescribed pressure head boundary with h=0. At the same time, the unsaturated part is treated as a prescribed flux boundary with Q=0. The length of each part is continually adjusted (Neuman, 1974) durig the iterative process until all calculated values of Q along the saturated part and all calculated values of h along the unsaturated part are negative, indicating, that water is leaving the porous medium through the saturated part of the boundary, and through this part only.

Water balance computation

Water balance computations are performed separately for each specified subregion of the flow region at prescribed print-times. The water balance information for each subregion consists of following items.

1. The actual volume of water in each subregion is given by:

$$V = \sum_{E} \varkappa A_E (\theta_i + \theta_j + \theta_k)/3 \tag{3.8}$$

where θ_i, θ_j, and θ_k are moisture contents evaluated at corner nodes of element E. The summation is taken over all elements within the subregion.

2. The actual rate of inflow/outflow to/from the subregion is

given by:

$$Q = \sum_{E} (V_{new} - V_{old})/\Delta t \tag{3.9}$$

where V_{new} and V_{old} are volumes of water in the subregion computed at actual and previous time level, respectively.

The element velocity computation

The Darcian velocity components are computed during numerical simulation at selected print-times. x- and z-components of the velocities are calculated for each triangular element or subelement according to:

$$\begin{aligned} v_x &= -\bar{K} \left[\frac{1}{2A_E} \left(\overset{x}{\gamma}_i h_i + \overset{x}{\gamma}_j h_j + \overset{x}{\gamma}_k h_k \right) + \hat{K}_{xz} \right] \\ v_z &= -\bar{K} \left[\frac{1}{2A_E} \left(\overset{z}{\gamma}_i h_i + \overset{z}{\gamma}_j h_j + \overset{z}{\gamma}_k h_k \right) + \hat{K}_{zz} \right] \end{aligned} \tag{3.10}$$

where:

$$\overset{x}{\gamma}_n = \hat{K}_{xx} b_n + \hat{K}_{xz} c_n$$

$$\overset{z}{\gamma}_n = \hat{K}_{xz} b_n + \hat{K}_{zz} c_n$$

Treatment of water uptake by plant roots

The set of nodes, n, for which the distribution function, β_n, discribing the space variation of potential root extraction intensity, S^*_n, is greater than zero, is treated as the root zone. The root extraction intensity is assumed to vary linearly within each element which allows the root extraction term D_n to be approximated by Eq. (3.2f). The values of actual root extraction intensity S_n are evaluated using Eq. (2.9). The total rate of transpiration per unit surface area is given by:

$$T = \frac{1}{L_T} \sum_e \varkappa A_e \bar{S} \tag{3.11}$$

the summation is taken over all elements within the flow region.

Interpolation technique for material hydraulic properties

For each defined material an internal table of hydraulic conductivity, K(h), and specific moisture capacity, C(h), are generated from a specified set of material parameters at the beginning of numerical simulation. The values of K_i and C_i are evaluated at prescribed pressure heads h_i generated within a specified interval (h_{Tab1}, h_{TabN}). The generation is performed so that:

$$\log |h_{i+1}| - \log |h_i| = \text{const.} \tag{3.12}$$

During any iteration, the values of hydraulic characteristics, K(h) and C(h), are computed using interpolation in the tables. If an argument h exceeds prescribed interval (h_{Tab1}, h_{TabN}), the hydraulic characteristics are evaluated directly (without interpolation). The used interpolation technique is about 3 times faster than direct evaluation of material functions in the whole range of pressure heads (the amount of saved computing time depends on chozen model of material characteristics).

Implementation of hydraulic conductivity anisotropy

Since the tensor of hydraulic conductivity anisotropy, K_A, is assumed to be symmetric, it is possible to define at any point of porous medium the local system of coordinates for which the matrix of tensor K_A is diagonal (with zeroes everywhere except of diagonal). The diagonal values of K_A are then refered to as a principal components of tensor K_A, and the local coordinate axes are told to be put into principal directions of tensor K_A.

In the program, the anisotropy is described such, that the orientation of the local principal directions may vary from element to element. The principal components, $\hat{K}_{11}$ and $\hat{K}_{22}$, toge ther with an angle, ω, between principal direction associated with $\hat{K}_{11}$ and x-axis of global coordinate system, are specified for each element. At the start of numerical simulation, the transformation of each locally determined tensor K_A is performed to the global system of coordinates, (x,z), using the transformation rule:

$$
\begin{aligned}
\hat{K}_{xx} &= \cos^2\omega\ \hat{K}_{11} + \sin^2\omega\ \hat{K}_{33} \\
\hat{K}_{zz} &= \sin^2\omega\ \hat{K}_{11} + \cos^2\omega\ \hat{K}_{33} \qquad (3.13) \\
\hat{K}_{xz} &= (\ \hat{K}_{11} - \hat{K}_{33})\ \sin\omega\ \cos\omega
\end{aligned}
$$

4. PROBLEM DEFINITION

4.1. CONSTRUCTION OF FINITE ELEMENT MESH

The mesh is constructed by dividing the flow region into quadrilateral and/or triangular elements the shapes of which are defined by the coordinates of the nodes that form their corners.

The element dimensions should be small parallel to directions along which large hydraulic gradients are expected to occur. More or less equidimensional elements are recommended to decrease numerical error. Quadrilaterals are automatically divided by the program into triangles which are then treated as subelements. Triangular elements may also be used subject to the following restrictions.

'Transverse lines' (Neuman, 1974) formed by element boundaries must transect the mesh along the general direction of its shorter dimension. These transverse lines must be continuous and non-intersecting, but they need not be straight. The nodes are numbered sequentially from 1 to NumNP (total number of nodes) proceeding up along each transverse line in the same direction. Elements are numbered in a similar manner. The maximum number of nodes in any one transverse line, IJ, is used to determine the effective size of the finite element matrix (i.e., its band). To minimize memory requirements, IJ should be as small as possible.

No additional restrictions due to the root zone definition are necessary.

To model axisymmetric three-dimensional flow systems, the vertical axis must coincide with, or be to the left of, the leftmost boundary of the mesh.

4.2. CODING MATERIALS AND SUBREGIONS

Materials

An integer code, beginning with 1 and ending with NMAT (the total number of materials), is assigned to each material within the flow region. The appropriate material code is then assigned to each nodal point N.

Interior material interfaces do not coincide with alement boundaries. When corner nodes of any one element are assigned different material numbers then material properties of this element are averaged by finite element algorithm. By that way material interfaces are 'softened'.

For each material a set of parameters of hydraulic characteristics must be specified. This set includes saturated hydraulic conductivity, saturated moisture content, residual moisture content and parameters which determine shape of the water retention curve and the relative conductivity function. For each element, the principal components of conductivity anisotropy tensor must be specified toge ther with an angle between local and global coordinate system.

An additional way of changing material properties is introduced by defining three scaling factors associated with moisture content, pressure head and hydraulic conductivity. The scaling factors are assigned to each nodal point N in the flow region.

Subregions

The water balance is computed separately for specified subregions. These subregions may or may not correspond with diferent material regions. Similarily to material coding an integer code, beginning with 1 and ending with NLay (the total number of subregions), is assigned to each subregion within the flow region. The appropriate material code is then assigned to each element in the flow region.

4.3. CODING BOUNDARY CONDITIONS

For each node, N, a boundary code, Kode(N), must be provided. If node N is to have a prescribed pressure head during a time step (Dirichlet boundary condition), Kode(N) must be set positive during that time step. If the volumetric rate of water entering or leaving the system at node N is prescribed during a time step (Neumann boundary condition), Kode(N) must be set negative or zero.

Constant boundary conditions

The values of constant boundary conditions are specified for particular node, N, by initial setting of pressure head, P(N), in the case of Dirichlet boundary condition, or initial amount of recharge/discharge, Q(N), in the case of Neuman boundary condition. The use of the variables Kode(N), Q(N) and P(N) for the various node types is summarized in following table.

TABLE 4.1. Initial setting of variables Kode(N), Q(N) and P(N) when the constant boundary conditions are applied.

Node Type	Kode(N)	Q(N)	P(N)
Internal; not sink/source	0	0.0	Initial Value
Internal; sink/source (Dirichlet condition)	1	0.0	Prescribed
Internal; sink/source (Neumann condition)	-1	Prescribed	Initial Value
Impermeable Boundary	0	0.0	Initial Value
Specified Head Boundary	1	0.0	Prescribed
Specified Flux Boundary	-1	Prescribed	Initial Value

<u>Variable Boundary conditions</u>

Three kinds of variable boundary conditions can be imposed on the system:

1.'Atmospheric' boundary condition .... Kode(N) = ±4

2. Variable pressure head boundary condition .... Kode(N) = +3

3. Variable flux boundary condition .... Kode(N) = -3

Each of these conditions can be specified for any part of the boundary, but it is not possible to specify more then one boundary condition of each kind. Initial setting of variables Kode(N), Q(N) and P(N) is described in following table.

TABLE 4.2. Initial setting of variables Kode(N), Q(N) and P(N) when the variable boundary conditions are applied.

Node Type	Kode(N)	Q(N)	P(N)
Atmospheric Boundary	-4	0.0	Initial Value
Variable Head Boundary	+3	0.0	Initial Value
Variable Flux Boundary	-3	0.0	Initial Value

When atmospheric boundary condition is specified (Kode(N)=±4), the time sequences of precipitation, Prec, and evaporation, rSoil, must be supplied via input file ATMOSPH.IN. The potential surface flux is then expressed as rAtm=rSoil-Prec. Actual surface flux is determined by the program. Two limiting values of surface pressure head must be also provided: hCritS, specifying maximum allowed pressure head at the surface (usually 0.0), and hCritA, specifying minimum allowed surface pressure head (defined from equilibrium conditions between soil water and atmospheric vapor). The value of Kode(N) is switched automatically by the program from -4 to +4 when either of limiting points is reached. The following table summarizes the use of variables rAtm, hCritS and hCritA during program execution (Width(N) denotes the lenght of the boundary segment associated with node N).

TABLE 4.3. Determination of variables Kode(N), Q(N) and P(N) if the atmospheric boundary condition is applied.

Kode(N)	Q(N)	P(N)	Event
-4	-Width(N)*rAtm	Unknown	rAtm=rSoil-Prec
+4	Unknown	hCritA	Evaporation capacity is exceeded
+4	Unknown	hCritS	Infiltration capacity is exceeded

When variable head or flux boundary condition is specified for certain part of the boundary (Kode(N) = ± 3), the input file ATMOSPH.IN must containe prescribed time sequences of pressure head, ht, or flux, rt, to be imposed on this part of boundary. The values of ht or rt are assigned to particular nodes in specified times according to following table.

TABLE 4.4. Determination of variables Kode(N), Q(N) and P(N) if the variable head or flux boundary condition is applied.

Node Type	Kode(N)	Q(N)	P(N)
Variable Head Boundary	+3	Unknown	ht
Variable Flux Boundary	-3	-Width(N)*rt	Unknown

Water uptake by plant roots

Within the root zone the rate of water removed from the system by plants is calculated by the program via the D-term in the finite element equations. Therefore, the value of Kode(N) must be set equal to 0 for all nodes in this region. Values of potential transpiration, rRoot, must be specified in predefined times via input file ATMOSPH.IN. Actual transpiration is calculated by the program. The sink term parameters are taken from separate input file SINK.IN. The values of function Beta(N) which describes the space variations of the potential extraction intensity must be sapecified for each node in the flow region (see description of input Block E in Chapter 5.). The part of the flow region for which Beta(N)>0 is then treated as a root zone.

Bottom Flux - Bottom Head Relationship q(h)

When the q(h)-relationship is available the special case of variable flux boundary condition, (Kode(N) = - 3), can be used. This case is indicated by setting the logical variable QGWLF from input file ATMOSPH.IN equal to .TRUE. . The value of discharge Q(N), which is assigned to to the node N, is determined by the program as Q(N)=-Width(N)*q(h) where h is set equal to actual value of pressure head P(N) and q(h) is given by:

$$q(h) = -Aqh \exp(Bqh |h-GWLOL|) \quad (4.1)$$

The parameters of function q(h), Aqh and Bqh, must be also specified in the file ATMOSPH.IN togeather with zero groundwater level GWLOL (usually equal to z-coordinate of soil surface).

Seepage faces

Initial setting of variables Kode(N), Q(N) and P(N) for any seepage face node is described in following table. All potential seepage faces must be identified before commencing the numerical simulation by giving a list of nodes within each seepage face (see input Block D).

TABLE 4.5. Initial setting of variables Kode(N), Q(N) and P(N) for seepage faces.

Node Type	Kode(N)	Q(N)	P(N)
Seepage Face (initially saturated)	+2	0.0	0.0
Seepage Face (initially unsaturated)	-2	0.0	Initial Value

4.4 DETERMINING MEMORY REQUIREMENTS

The parameter statement (second statement in the program) is used to define the problem dimensions. All main arrays are dimensioned via adjustable array declarators. This makes it possible to change the problem dimensions without rewritting declarations and recompiling all program subroutines. Different problems can be investigated by changing the problem dimensions in the parameter statement of the main program and attaching the previously compiled subroutines prior to execution. The list of the problem dimensions defined by the parameter statement follows.

TABLE 4.6. List of the problem dimensions.

Dimension Name	Description
NumNPD	Max. number of nodes in finit element mesh.
NumElD	Max. number of elements in finite element mesh.
MBandD	Max. number of nodes in any transverse line + 2.
NumBPD	Max. number of boundary nodes for which Kode(N)≠0
NSeepD	Max. number of seepage faces.
NumSPD	Max. number of nodes in any one seepage face.
NMatD	Max. number of materials.
NTabD	Max. number of items in the table of hydraulic characteristics generated by the program for each material.

5. INPUT DATA

Input to program SWM II is divided into four input files. Each input file consists of one or more input blocks identified by letters from A to I. These blocks must be arranged as follows:

SELECTOR.IN
A. Basic Information
B. Material Information
C. Time Information
D. Seepage Information

GRID.IN
E. Nodal Information
F. Element Information
G. Boundary Geometric Information

ATMOSPH.IN
H. Atmospheric Information

SINK.IN
I. Sink Information

Following are tables describing the data required for each input block. All data are read using list-directed formatting (free format). To facilitate orientation within each input block the comment lines are supplied. The content of these lines is ignored when the program is performed. The comment lines may be left blank but not omitted. All input files are supposed to be placed in the folder SWMII.IN.

All input data must be specified in consistent set of units in both length and time.

BLOCK A - Basic Information

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Char	Hed	Heading.
4	-	-	Comment line.
5	Char	LUnit	Length unit (e.g. 'cm').
5	Char	TUnit	Time unit (e.g. 'min').
6	-	-	Comment line.
7	Int	Kat	Type of flow system being analyzed. 0 if horizontal plane flow 1 if axisymmetric flow 2 if vertical plane flow
8	-	-	Comment line.
9	Int	MaxIt	Maximum number of iterations allowed in any time step (usually 10 - 20).
9	Real	Tol	Maximum desired absolute change [L] in the values of pressure head, h, between two successive iterations in any time step (recommended value is 1 mm).
10	-	-	Comment line.
11	log	CheckF	Set this logical variable equal to .true. if the grid input data are to be printed for checking.
11	log	ShortF	.true. if printing of time-level information on each time level is to be suppressed and the information printed only in specified print times. .false. if this information is to be printed on each time level.
11	log	FluxF	.true. if printing of detailed element flux information and discharge/recharge information is requested.
11	log	AtmInF	.true. if atmospheric control data are supplied via input file ATMOSPH.IN . .false. if the file ATMOSPH.IN is not provided (i.e., in the case of time independent boundary conditions).
11	log	SeepF	Set this logical variable equal to .true. if any seepage face is defined.

BLOCK B - Material Information

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Int	NMat	Number of materials. The materials are identified by material number, MatNum, specified in Block E.
3	Int	NLay	Number of subregions in which the water balance is computed separately. The subregions are identified by subregion number, LayNum, specified in Block F.
3	Real	hTab1	Absolute value of the lower limit [L] of the pressure head interval in which internal table of hydraulic characteristics is generated for each material (hTab1 must be greater than 0.0; recommended value is 0.001 cm).
3	Real	hTabN	Absolute value of the upper limit [L] of the pressure head interval (e.g. 1000 cm). If abs. value of pressure head is out of the interval <htab1,hTabN>, then appropriate values of hydraulic characteristics are computed directly (without interpolation in the table).
3	Int	NPar	Number of parameters specified for each material (i.e., 9 in the case of the modified van Genuchten's model).
4	-	-	Comment line.
5	Real	Par(1,M)	Parameter θ_r of material M.
5	Real	Par(2,M)	Parameter θ_s of material M.
5	Real	Par(3,M)	Parameter θ_a of material M.
5	Real	Par(4,M)	Parameter θ_m of material M.
5	Real	Par(5,M)	Parameter α of material M [1/L].
5	Real	Par(6,M)	Parameter n of material M.
5	Real	Par(7,M)	Parameter K_s of material M [L/T].
5	Real	Par(8,M)	Parameter K_k of material M [L/T].
5	Real	Par(9,M)	Parameter θ_k of material M.
			The same record as above must be provided for each material M (from 1 to NMat).

BLOCK C - Time Information

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Real	dt	Initial time increment Δt [T].
3	Real	dtMin	Δt_{min} [T].
3	Real	dtMax	Δt_{max} [T].
3	Real	DMul	If number of necessary iterations in the end of any time step is less than or equal to 3 then Δt is multiplied by dimensionless number DMul $\geq$ 1.0 (usually DMul should not exceed 1.3).
3	Real	DMul2	If the number of iterations is greater or equal to 7 then Δt is multiplied by DMul2 $\leq$ 1.0 (recommended value is 0.33).
3	Int	MPL	Number of specified print-times in which detailed pressure head, moisture, flux, and water balance information is printed.
4	-	-	Comment line.
5	Real	TPrint(1)	1st specified print-time [T].
5	Real	TPrint(2)	1st specified print-time [T].
.	.	.	.
.	.	.	.
5	Real	TPrint(MPL)	Last specified print-time [T].

BLOCK D - Seepage Face Information

Block D may be omitted if logical variable SeepF (Block A) is set equal to .false. . On each seepage face, I, the nodes with KODE(N) = ±2 are numbered locally in sequence from J=1 to J=NSP(I), starting from the saturated side of the seepage face. Thus, each node on a seepage face is identified by a local number, J, and a global number, NP(I,J).

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Int	NSeep	Number of seepage faces expected to develop.
4	-	-	Comment line.
5	Int	NSP(1)	Number of nodes on the 1st seepage face.
5	Int	NSP(2)	Number of nodes on the 2nd seepage face.
.	.	.	.
.	.	.	.
5	Int	NSP(NSeep)	Number of nodes on the last seepage face.
6	-	-	Comment line.
7	Int	KodeS(1)	Set this variable equal to -1.
7	Int	KodeS(2)	Set this variable equal to -1.
.	.	.	.
.	.	.	.
7	Int	KodeS(NSeep)	Set this variable equal to -1.
8	-	-	Comment line.
9	Int	NP(1,1)	Sequential global number (N) of 1st node on first seepage face.
9	Int	NP(1,2)	Sequential global number (N) of 2nd node on first seepage face.
.	.	.	.
.	.	.	.
9	Int	NP(1,NSP(1))	Sequential global number (N) of last node on first seepage face.
			The same record as above must be provided for each seepage face.

BLOCK E - Nodal Information

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Int	NumNP	Number of nodal points.
3	Int	NumEl	Number of elements (quadrilaterals and/or triangles).
3	Int	IJ	Maximum number of nodes in any transverse line.
3	Int	NumBP	Number of boundary nodes for which Kode(N) is not equal to 0.
4	-	-	Comment line.
5	Int	N	=N (nodal point number).
5	Int	Kode(1)	Code specifying type of boundary condition applying to the nodal point. Legal values include 0,±1,±2,±3, and -4 (see Section 4).
5	Real	X(N)	x-coordinate [L] of the nodal point. This must be the horizontal coordinate.
5	Real	Z(N)	z-coordinate [L] of the nodal point. This must be the vertical coordinate in problems involving vertical planar or axisymmetric flow. In the axisymmetric case, z must coincide with the vertical axis of symmetry.
5	Real	P1(N)	Initial value of pressure head [L] at the nodal point.
5	Real	Q(N)	Prescribed recharge/discharge [L^2/T] at the nodal point, negative out of the system (in axisymmetric case [L^3/T]). If the value of Q(N) is not prescribed set Q(N) equal to zero.
5	Int	MatNum(N)	Number of material whose hydraulic properties are assigned to the node.
5	Real	Beta(N)	Nodal value of the function which describes space distribution of potential transpiration demand in the root zone. If node N is not inside the root zone set Beta(N) equal to zero.
5	Real	Axz(N)	The nodal value of dimensionless scaling factor associated with pressure head.

(for continuation see next page)

BLOCK E (continued)

Record	Type	Symbol	Description
5	Real	Bxz(N)	The nodal value of the scaling factor associated with hydraulic conductivity.
5	Real	Dxz(N)	The nodal value of the scaling factor associated with moisture content.
			In general, one record as that above is required for each nodal point, starting with N=1 and continuing in sequence up to N=NumNP. Some records may be skipped for certain nodes subject to the following conditions.

Consider two nodes, N1 and N2, along a transverse line such that N2 is greater than N1+1. Then only the data for nodes N1 and N2 need be specified provided that all of the following conditions are met simultaneously:

1. All nodes along the transverse line between node N1 and N2 lie equal distances apart along a straight line.
2. The values of P1(N), Beta(N), Axz(N), Bxz(N), and Dxz(N) vary linearly between node N1 and N2.
3. The values of Kode(N), Q(N) and MatNum(N) for N = N1,N1+1,...,N2-1 are the same.

BLOCK F - Element Information

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Int	E	=E (element number)
3	Int	KX(E,1)	Global number of corner node i.
3	Int	KX(E,2)	Global number of corner node j.
3	Int	KX(E,3)	Global number of corner node k.
3	Int	KX(E,4)	Global number of corner node l. Indexes i, j, k, and l, refer to the corner nodes of the element taken in a counter-clockwise direction.
3	Real	Angle(E)	Angle in degrees between $\hat{K}_{11}$ and the x coordinate axis assigned to the element.
3	Real	ConA11(E)	First principal component, $\hat{K}_{11}$, of the dimensionless tensor K_A which describes anisotropy of hydraulic conductivity assigned to the element.
3	Real	ConA33(E)	Second principal component, $\hat{K}_{33}$.
3	Int	LayNum(E)	Subregion number assigned to the element.
			In general, one record is required for each element, starting with E=1 and continuing in sequence up to E=NumEl. Some records may be skipped for certain elements subject to the following conditions.

Consider two elements, E1 and E2, between two given transverse lines such that E2 is greater than E1. Then only the data for element E1 need be specified (data for elements E1+1 through E2 may be omitted) provided that all of the following conditions are met simultaneously:

1. All elements between E1 and E2 are quadrilaterals, including E1 and E2.
2. All elements, E1,...,E2, are assigned the same values of Angle(E), ConA11(E), ConA33(E), and LayNum(E).

BLOCK G - Boundary Geometric Information

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Int	KXB(1)	Global number of 1st of the set of sequentially numbered boundary nodes for which Kode(N) is not equal to zero.
3	Int	KXB(2)	As above for 2nd boundary node.
.	.	.	.
.	.	.	.
3	Int	KXB(NumBP)	As above for the last boundary node.
4	-	-	Comment line.
5	Real	Width(1)	Width of the boundary [L] associated with boundary node KXB(1). Width(N) includes half the boundary length of each element in the immediate vicinity of the node KXB(N) which is assigned the kind of boundary condition specified by Kode(N). In the case of axisymmetric flow, Width(N) is the area of boundary strip [L^2] which node N represents).
5	Real	Width(2)	As above for node KXB(2).
.	.	.	.
.	.	.	.
5	Real	Width(NumBP)	As above for node KXB(NumBP).
6	-	-	Comment line.
7	Real	rLen	Length associated with transpiration (length of soil surface [L] or, in the case of axisymmetric flow, surface area [L^2]). For problems without transpiration set this variable equal zero.

BLOCK H - Atmospheric Information

Block H is not read if logical variable AtmInF (Block A) is set equal to .false. .

Record	Type	Symbol	Description
1,2,3,4	-	-	Comment lines.
5	Log	SinkF	Set this variable equal to .true. if water extraction from the root zone is defined.
5	Log	Dummy	Set this variable equal to .false. .
5	Log	qGWLF	Set this variable equal to .true. if the discharge-groundwater level relationship q(GWL) is applied as bottom boundary condition.
6	-	-	Comment line.
7	Real	GWLOL	The value of z-coordinate [L] for which GWL=0 (usually z-coordinate of soil surface). If GWL is not specified set GWLOL=0.
7	Real	Aqh	The value of parameter a in the q(GWL)-relationship (set = 0 if qGWLF =.false.).
7	Real	Bqh	The value of parameter b in the q(GWL)-relationship (set = 0 if qGWLF =.false.).
8	-	-	Comment line.
9	Real	tInit	Starting time [T] of the simulation.
9	Real	MaxAL	Number of atmospheric data records.
10	-	-	Comment line.
11	Real	hCritS	Maximum allowed pressure head at the soil surface [L].
12	-	-	Comment line.
13	Real	tAtm(i)	Time for which the i-th tmospheric data record is provided [T].
13	Real	Prec(i)	Precipitation [L/T] (in absolute value).
13	Real	rSoil(i)	Potential soil evaporation rate [L/T] (in absolute value).
13	Real	rRoot(i)	Potential transpiration [L/T] (in absolute value) .

(for continuation see next page)

BLOCK H (continued)

Record	Type	Symbol	Description
13	Real	hCritA(i)	Absolute value of minimum allowed pressure head at the soil surface [L].
13	Real	rt(i)	Bottom flux [L/T] or other time variable prescribed flux boundary condition (positive out of the system) indicated by Kode(N)=-3 (set equal to 0 if no Kode(N)=-3 boundary condition is specified).
13	Real	ht(i)	Groundwater level [L] (usually negative) or other time variable prescribed head boundary condition indicated by Kode(N)=+3 (set equal to 0 if no Kode(N)=+3 boundary condition is specified). The number of atmospheric-data records is MaxAL (i = 1,2,...,MaxAL)

BLOCK I - Sink Information

Block I is not read if logical variable SinkF (Block H) is set equal to .false. .

Record	Type	Symbol	Description
1,2	-	-	Comment lines.
3	Real	P0	The value of pressure head [L] below which roots start to extract water from the soil (starting point).
3	Real	P2H	The value of pressure head [L] below which the roots cannot extract water optimally anymore, for potential transpiration rate equal to r2H (limiting point).
3	Real	P2L	As above, but for potential transpiration rate equal to r2L.
3	Real	P3	The value of pressure head [L] below which no water uptake by roots is possible (wilting point).
3	Real	r2H	[L/T], usually 0.5 cm/day.
3	Real	r2L	[L/T], usually 0.1 cm/day.
4	-	-	Comment line.
5	Real	POptm(1)	The value of pressure head [L] below which roots start to extract water optimally for material number 1.
5	Real	POptm(2)	As above, for material number 2.
.	.	.	.
.	.	.	.
5	Real	POptm(NMat)	As above, for material number NMat.

6. OUTPUT DATA

The program output consists of 12 output files which are organized into 3 groups:

T-level information
- H_MEAN.OUT
- V_MEAN.OUT
- CUM_Q.OUT
- RUN_INF.OUT

P-level information
- H.OUT
- TH.OUT
- Q.OUT
- VZ.OUT
- VX.OUT
- BOUNDARY.OUT
- BALANCE.OUT

A-level information
- A_LEVEL.OUT

In addition, some of the input data are printed to the file CHECK.OUT. This file contains a complete description of the finite element mesh, the boundary node of each node, and the hydraulic characteristics of each material (finite element mesh data are printed only if logical variable CheckF in input Block A is set equal to .TRUE.). All output files are directed to the folder SWMII.OUT.

<u>T-level information</u>

This group of output files contains information which is printed at the end of each time step (printing can be suppressed by setting logical variable ShortF from input Block A equal to .TRUE., the information is then printed only at selected print-times). Following data are printed to particular files:

H_MEAN.OUT (mean pressure heads)

- **hAtm:** Mean value of pressure head calculated over a set of nodal points which are assigned Kode(n)=±4 (i.e., along part of the boundary controled by atmospheric conditions).

- **hRoot:** Mean value of pressure head over the region for which Beta(n)>0 (i.e., within the root zone).

- **hKode3:** Mean value of pressure head calculated over a set of nodal points which are assigned Kode(n)=±3 (i.e., along part of the boundary on which groundwater level, bottom flux, or other time variable pressure head or/and flux boundary conditions are imposed).

- **hKode1:** Mean value of pressure head calculated over a set of nodal points which are assigned Kode(n)=±1 (i.e., along part of the boundary on which time independent pressure head or/and flux boundary conditions are imposed).

- **hSeep:** Mean value of pressure head calculated over a set of nodal points which are assigned Kode(n)=±2 (i.e., along seepage faces).

V_MEAN.OUT (mean fluxes, positive out of the system)

- **rAtm:** Potential surface flux along the atmospheric boundary (Kode(n)=±4).

- **rRoot:** Potential transpiration.

- **vAtm:** Mean value of actual surface flux along the atmospheric boundary (Kode(n)=±4).

- **vRoot:** Actual transpiration.

- **vKode3:** Mean value of bottom or another boundary flux along part of the boundary on which groundwater level, bottom flux, or other time variable pressure head or/and flux boundary conditions are imposed (Kode(n)=±3).

- **vKode1:** Mean value of boundary flux along part of the boundary on which time independent pressure head or/and flux boundary conditions are imposed (Kode(n)=±1).

- **vSeep:** Mean value of boundary flux along seepage faces (Kode(n)=±2).

CUM_Q.OUT (mean cumulative fluxes per unit of particular part of a boundary as described above for mean fluxes, positive out of the system)

RUN_INF.OUT (time and iteration information)

TLevel: Time-level

Time: Simular time associated with the time-level

dt: Δt

Iter: Number of iterations

ItCum: Cumulative number of iterations

Real time: Computation time spent on the simulation

P-level information

This information is printed only at prescribed print-times. Following output files are supplied:

H.OUT (Nodal values of pressure head)

TH.OUT (Nodal values of moisture content)

Q.OUT (Discharge/recharge from/to the system assigned to the boundary nodes or internal sink/source nodes)

VZ.OUT (Element values of z-components of Darcian velocity vector)

VX.OUT (Element values of x-components of Darcian velocity vector)

BOUNDARY.OUT (This file contains information about each boundary node, n, for which $Kode(n) \neq 0$, including: discharge/recharge, $Q(n)$, boundary flux, $v(n)$, pressure head, $h(n)$, and moisture content $\theta(n)$)

BALANCE.OUT (This file contains the volume of water inside, and inflow/outflow to/from, each specified subregion, togeather with mean pressure head over the subregion)

A-level information

This information is printed at each time associated with specification of time variable boundary conditions. A-level information is directed to the output file A_LEVEL.OUT. Following data are printed.

CumQAP: Mean cumulative potential surface flux per unit of the atmospheric boundary (Kode(n)=±4).

CumQRP: Mean cumulative potential transpiration.

CumQA: Mean cumulative value of actual surface flux per unit of the atmospheric boundary (Kode(n)=±4).

CumQR: Mean cumulative value of actual transpiration.

CumQ3: Mean cumulative value of bottom or another boundary flux per unit of the boundary on which groundwater level, bottom flux, or other time variable pressure head or/and flux boundary conditions are imposed (Kode(n)=±3).

hAtm: Mean value of pressure head calculated over a set of nodal points which are assigned Kode(n)=±4.

hRoot: Mean value of pressure head over the region for which Beta(n)>0 (i.e., within the root zone).

hKode3: Mean value of pressure head calculated over a set of nodal points which are assigned Kode(n)=±3.

7. EXAMPLE PROBLEMS

7.1. EXAMPLE PROBLEM #1 - COLUMN TEST

This example is based on data from laboratory experiments as presented by Skaggs et al. (1970). Since the same example is used by Davis and Neuman (1983) as a test problem for program UNSAT2 it may serve for comparison between SWM II and UNSAT2 codes.

A graphical representation of the soil column and the finite element mesh used in the numerical simulation are presented in Figure 7.1. The retention curve and relative hydraulic conductivity function of sand mix material are presented in Figure 7.2. Initially the sand was assumed to be dry having a pressure head of -150 cm. The soil hydraulic properties were assumed to be homogenous and isotropic with a saturated hydraulic conductivity of 0.000722 cm/sec. One-dimensional vertical flow occurs due to infiltration at the soil surface from the ponded water. The bottom of the column was left open and the sides of the column were impervious.

Tables 7.1. and 7.2. provide an entire listing of the input data. Nodes 1 and 2 at the soil surface have been assigned a Kode(N) value of 1 indicating that these two nodes have a prescribed pressure head which is equal to the depth of ponded water (0.75 cm). Nodes 111 and 112 at the column bottom have been specified as Kode(N)=-2 indicating that they lie on an unsaturated portion of the boundary that may become a seepage face during later stages of the computations. For the remaining nodes, values for Kode(N) and flux, Q(N), were set equal to zero. This indicates that flow does not occur either into or out of the system of these nodes.

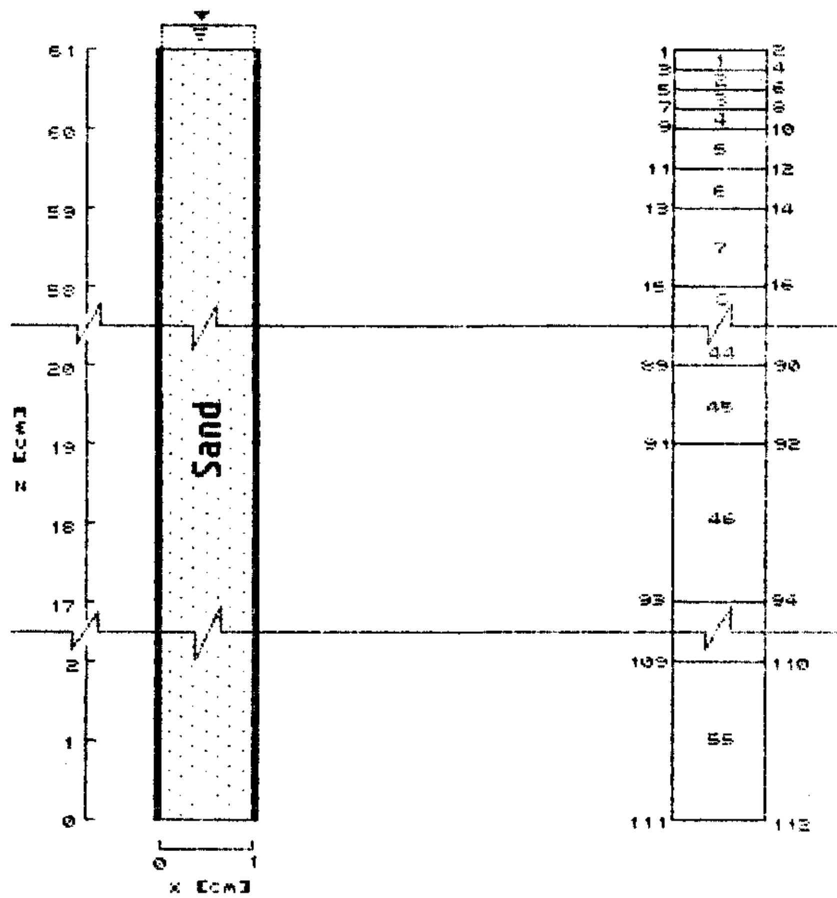
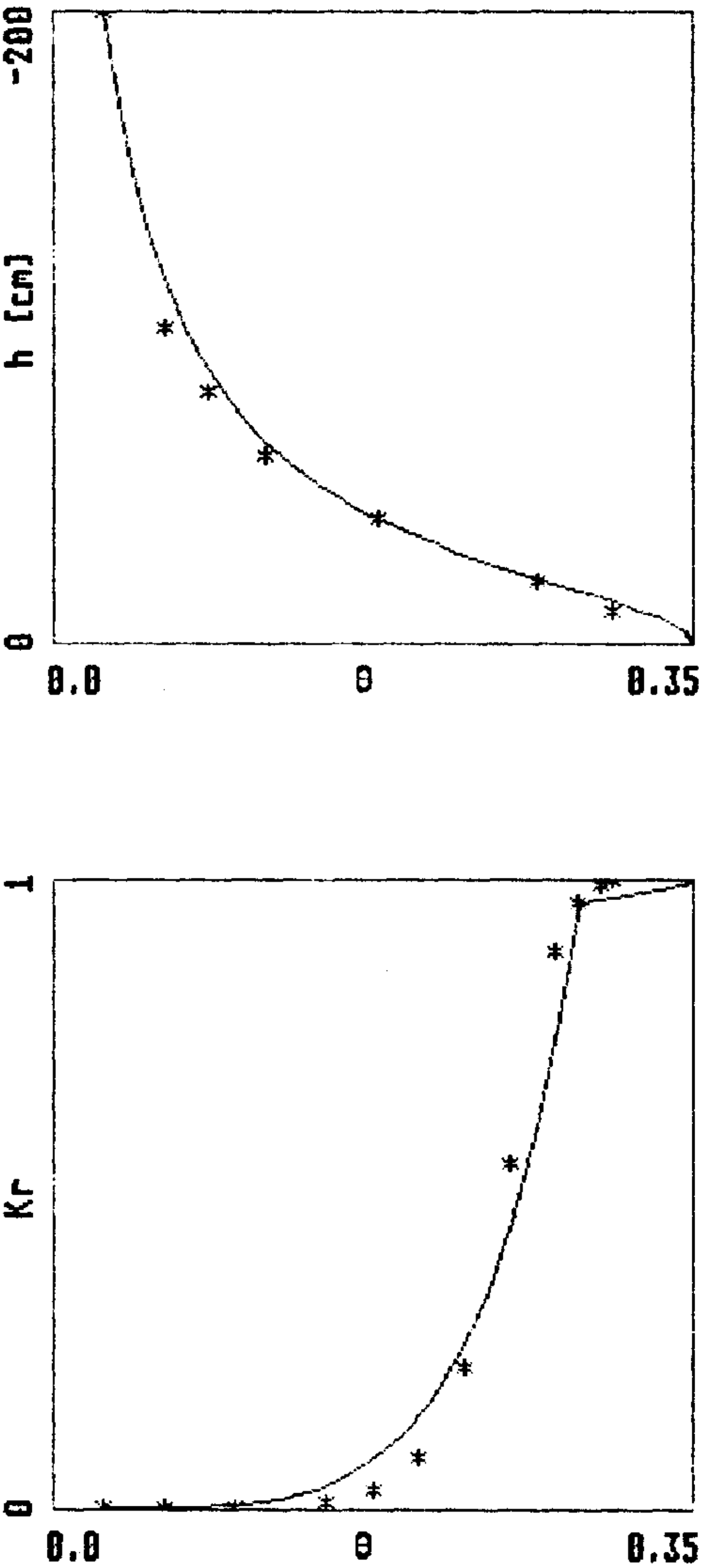


Figure 7.1. Flow system and finite element mesh for Example Problem #1



Fifure 7.2. Retention curve, h(θ), and relative hydraulic conductivity function, $K_r(\theta)$, for Example Problem #1, * denotes UNSAT2 input data (Davis and Neuman, 1983).

The simulation was executed for 5400 seconds which corresponds to the length of the experiment. The infiltration rates and cumulative infiltration rates resulting from the column simulation are presented in Figure 7.3. Also shown on the figure are the results of UNSAT2 simulation (Davis and Neuman 1983).

Tabulation of the pressure head profiles from the SWM II simulation for time intervals of 60, 900, 1800, 2700, 3600, and 5400 seconds is shown in Table 7.3. Moisture content, recharge/discharge, and Darcian flux profiles at the end of simulation are tabulated in Table 7.4., 7.5., and 7.6. respectively. Mean boundary pressure heads, mean boundary fluxes, and mean cumulative boundary fluxes are tabulated in Tables 7.7., 7.8., and 7.9. The information about changes of the time increment, number of iterations necessary on selected time levels to reach the convergence, and computation time which was spent on simulation presents Table 7.10.

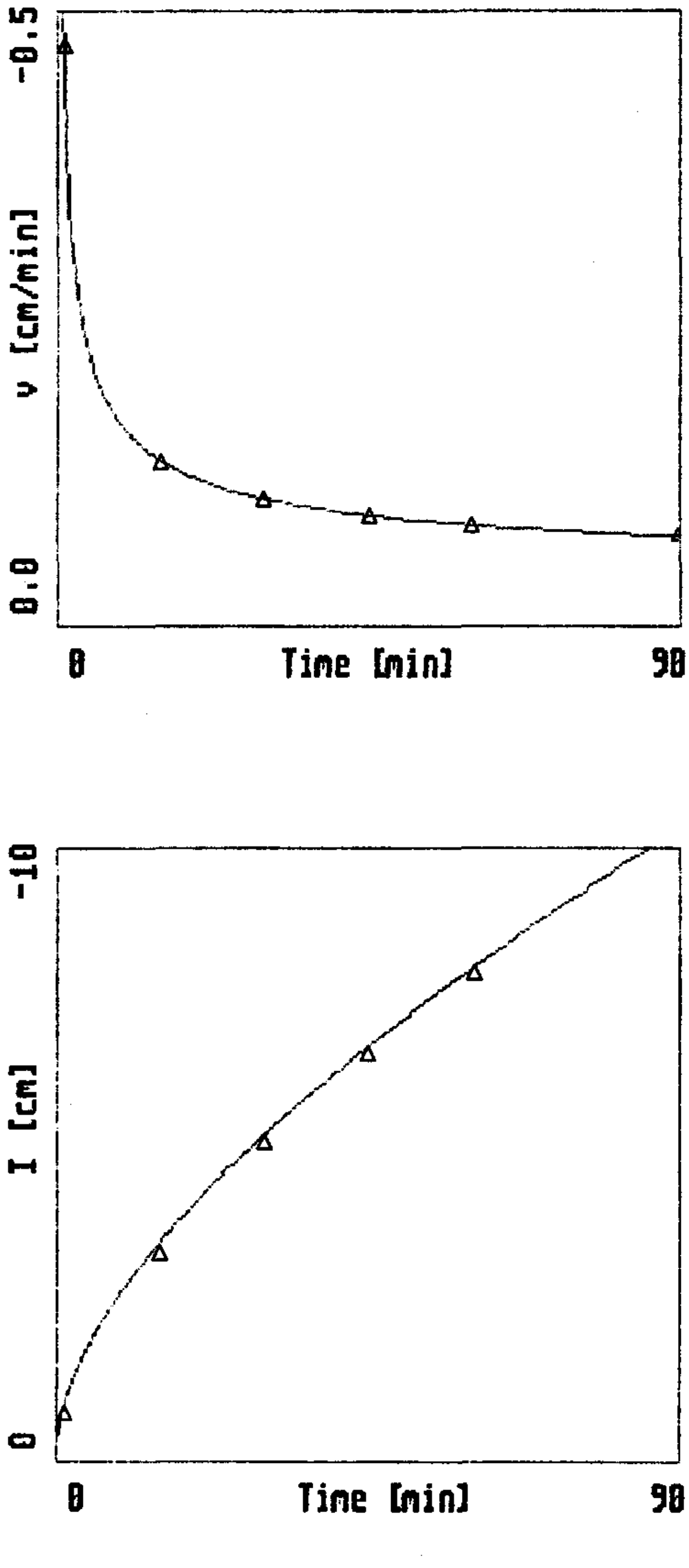


Figure 7.3. Infiltration Rate, v, and Cumulative Infiltration Rate, I, simulated by SWM II '___' and UNSAT2 'Δ'

TABLE 7.1. Input Data for Example Problem #1, Input File 'SELECTOR.IN'.

```
*** BLOCK A: BASIC INFORMATION *****************************************
Heading
  ' Example problem #1 - Column Test'
LUnit   TUnit        (indicated units are obligatory for all input data)
  'cm'  'sec'
Kat (0:horizontal plane, 1:axisymmetric vertical flow, 2:vertical plane)
  2
MaxIt   Tol          (maximum number of iterations and p. head tolerance)
  20    .10
CheckF  ShortF  FluxF   AtmInF  SeepF
  f       t       t       f       t
*** BLOCK B: MATERIAL INFORMATION **************************************
NMat    NLay    hTab1   hTabN   NPar
  1       1     .001    200.     9
thr     ths     tha     thm     Alfa    n       Ks      Kk      thk
 .02    .350    -.02    .350    .0410   1.964  .000722 .000695 .2875
*** BLOCK C: TIME INFORMATION ******************************************
dt      dtMin   dtMax   DMul    DMul2   MPL
  1.    .01     60.     1.1     .33      6
TPrint(1),TPrint(2),...,TPrint(MPL)                   (print-time array)
  60 900 1800 2700 3600 5400
*** BLOCK D: SEEPAGE INFORMATION (only if SeepF =.true.) ***************
NSeep                                           (number of seepage faces)
  1
NSP(1),NSP(2),.......,NSP(NSeep)          (number of nodes in each s.f.)
  2
KodeS(1),KodeS(2),...,KodeS(NSeep)  (set all elements of this array =-1)
 -1
NP(i,1),NP(i,2),.....,NP(i,NSP(i))      (nodal number array of i-th s.f.)
  111     112
*** END OF INPUT FILE 'SELECTOR.IN' ************************************
```

TABLE 7.2. Input Data for Example Problem #1, Input File 'GRID.IN'.

```
*** BLOCK E: NODAL INFORMATION ******************************************
NumNP NumEl  IJ   NumBP
 112   55    2     4
  N  Kode    X     Z        P1       Q       M  Beta  Axz   Bxz   Dxz
  1   1    0.00  61.00     0.75  0.00E+00  1  0.00  1.00  1.00  1.00
  2   1    1.00  61.00     0.75  0.00E+00  1  0.00  1.00  1.00  1.00
  3   0    0.00  60.75  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
  4   0    1.00  60.75  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
  5   0    0.00  60.50  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
  6   0    1.00  60.50  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
  7   0    0.00  60.25  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
  8   0    1.00  60.25  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
  9   0    0.00  60.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 10   0    1.00  60.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 11   0    0.00  59.50  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 12   0    1.00  59.50  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 13   0    0.00  59.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 14   0    1.00  59.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 15   0    0.00  58.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 16   0    1.00  58.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 17   0    0.00  57.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 18   0    1.00  57.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 19   0    0.00  56.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 20   0    1.00  56.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 21   0    0.00  55.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 22   0    1.00  55.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 23   0    0.00  54.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 24   0    1.00  54.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 25   0    0.00  53.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 26   0    1.00  53.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 27   0    0.00  52.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 28   0    1.00  52.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 29   0    0.00  51.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 30   0    1.00  51.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 31   0    0.00  50.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 32   0    1.00  50.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 33   0    0.00  49.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 34   0    1.00  49.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 35   0    0.00  48.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 36   0    1.00  48.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 37   0    0.00  47.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 38   0    1.00  47.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 39   0    0.00  46.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 40   0    1.00  46.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 41   0    0.00  45.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 42   0    1.00  45.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 43   0    0.00  44.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 44   0    1.00  44.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 45   0    0.00  43.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 46   0    1.00  43.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 47   0    0.00  42.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 48   0    1.00  42.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 49   0    0.00  41.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 50   0    1.00  41.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 51   0    0.00  40.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
 52   0    1.00  40.00  -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
```

TABLE 7.2. Input Data for Example Problem #1, Input File 'GRID.IN'.
(Continuation)

53	0	0.00	39.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
54	0	1.00	39.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
55	0	0.00	38.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
56	0	1.00	38.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
57	0	0.00	37.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
58	0	1.00	37.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
59	0	0.00	36.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
60	0	1.00	36.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
61	0	0.00	35.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
62	0	1.00	35.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
63	0	0.00	34.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
64	0	1.00	34.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
65	0	0.00	33.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
66	0	1.00	33.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
67	0	0.00	32.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
68	0	1.00	32.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
69	0	0.00	31.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
70	0	1.00	31.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
71	0	0.00	30.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
72	0	1.00	30.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
73	0	0.00	29.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
74	0	1.00	29.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
75	0	0.00	28.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
76	0	1.00	28.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
77	0	0.00	27.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
78	0	1.00	27.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
79	0	0.00	26.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
80	0	1.00	26.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
81	0	0.00	25.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
82	0	1.00	25.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
83	0	0.00	24.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
84	0	1.00	24.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
85	0	0.00	23.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
86	0	1.00	23.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
87	0	0.00	22.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
88	0	1.00	22.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
89	0	0.00	21.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
90	0	1.00	21.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
91	0	0.00	20.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
92	0	1.00	20.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
93	0	0.00	18.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
94	0	1.00	18.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
95	0	0.00	16.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
96	0	1.00	16.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
97	0	0.00	14.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
98	0	1.00	14.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
99	0	0.00	12.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
100	0	1.00	12.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
101	0	0.00	10.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
102	0	1.00	10.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
103	0	0.00	8.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
104	0	1.00	8.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
105	0	0.00	6.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
106	0	1.00	6.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
107	0	0.00	4.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00
108	0	1.00	4.00	-150.00	0.00E+00	1	0.00	1.00	1.00	1.00

TABLE 7.2. Input Data for Example Problem #1, Input File 'GRID.IN'.
(Continuation)

```
109  0    0.00    2.00   -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
110  0    1.00    2.00   -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
111 -2    0.00    0.00   -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
112 -2    1.00    0.00   -150.00  0.00E+00  1  0.00  1.00  1.00  1.00
*** BLOCK F: ELEMENT INFORMATION ****************************************
  E   i   j   k   l   Angle  ConA11 ConA33 LayNum
  1   1   3   4   2   0.00   1.00   1.00   1
  2   3   5   6   4   0.00   1.00   1.00   1
  3   5   7   8   6   0.00   1.00   1.00   1
  4   7   9  10   8   0.00   1.00   1.00   1
  5   9  11  12  10   0.00   1.00   1.00   1
  6  11  13  14  12   0.00   1.00   1.00   1
  7  13  15  16  14   0.00   1.00   1.00   1
  8  15  17  18  16   0.00   1.00   1.00   1
  9  17  19  20  18   0.00   1.00   1.00   1
 10  19  21  22  20   0.00   1.00   1.00   1
 11  21  23  24  22   0.00   1.00   1.00   1
 12  23  25  26  24   0.00   1.00   1.00   1
 13  25  27  28  26   0.00   1.00   1.00   1
 14  27  29  30  28   0.00   1.00   1.00   1
 15  29  31  32  30   0.00   1.00   1.00   1
 16  31  33  34  32   0.00   1.00   1.00   1
 17  33  35  36  34   0.00   1.00   1.00   1
 18  35  37  38  36   0.00   1.00   1.00   1
 19  37  39  40  38   0.00   1.00   1.00   1
 20  39  41  42  40   0.00   1.00   1.00   1
 21  41  43  44  42   0.00   1.00   1.00   1
 22  43  45  46  44   0.00   1.00   1.00   1
 23  45  47  48  46   0.00   1.00   1.00   1
 24  47  49  50  48   0.00   1.00   1.00   1
 25  49  51  52  50   0.00   1.00   1.00   1
 26  51  53  54  52   0.00   1.00   1.00   1
 27  53  55  56  54   0.00   1.00   1.00   1
 28  55  57  58  56   0.00   1.00   1.00   1
 29  57  59  60  58   0.00   1.00   1.00   1
 30  59  61  62  60   0.00   1.00   1.00   1
 31  61  63  64  62   0.00   1.00   1.00   1
 32  63  65  66  64   0.00   1.00   1.00   1
 33  65  67  68  66   0.00   1.00   1.00   1
 34  67  69  70  68   0.00   1.00   1.00   1
 35  69  71  72  70   0.00   1.00   1.00   1
 36  71  73  74  72   0.00   1.00   1.00   1
 37  73  75  76  74   0.00   1.00   1.00   1
 38  75  77  78  76   0.00   1.00   1.00   1
 39  77  79  80  78   0.00   1.00   1.00   1
 40  79  81  82  80   0.00   1.00   1.00   1
 41  81  83  84  82   0.00   1.00   1.00   1
 42  83  85  86  84   0.00   1.00   1.00   1
 43  85  87  88  86   0.00   1.00   1.00   1
 44  87  89  90  88   0.00   1.00   1.00   1
 45  89  91  92  90   0.00   1.00   1.00   1
 46  91  93  94  92   0.00   1.00   1.00   1
 47  93  95  96  94   0.00   1.00   1.00   1
 48  95  97  98  96   0.00   1.00   1.00   1
 49  97  99 100  98   0.00   1.00   1.00   1
 50  99 101 102 100   0.00   1.00   1.00   1
```

TABLE 7.2. Input Data for Example Problem #1, Input File 'GRID.IN'.
(Continuation)

```
 51 101 103 104 102   0.00   1.00   1.00   1
 52 103 105 106 104   0.00   1.00   1.00   1
 53 105 107 108 106   0.00   1.00   1.00   1
 54 107 109 110 108   0.00   1.00   1.00   1
 55 109 111 112 110   0.00   1.00   1.00   1
*** BLOCK 6: BOUNDARY GEOMETRIC INFORMATION ******************************
KXB(1),KXB(2),...,KXB(NumBP)              (boundary node number array)
 1     2     111     112
Width(1),Width(2),...,Width(NumBP)                       (width array)
 0.50   0.50  0.50    0.50
rLen                                          (length of the surface)
 1.00
*** END OF INPUT FILE 'GRID.IN' *****************************************
```

TABLE 7.3. Output Data, Example Problem #1, Output File 'H.OUT'.

Example problem #1 - Column Test

Program SWM II
Date: 26. 4.1988 Time: 13:40
Time independent boundary conditions
Vertical plane flow, V = L*L
Units: L = cm , T = sec

Time *** 60.0000 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	61.0	0.8	0.8
3	0.0	60.7	-1.7	-1.7
5	0.0	60.5	-4.2	-4.2
7	0.0	60.2	-6.6	-6.7
9	0.0	60.0	-9.0	-9.3
11	0.0	59.5	-13.7	-14.3
13	0.0	59.0	-19.1	-20.1
15	0.0	58.0	-41.6	-40.7
17	0.0	57.0	-129.3	-124.3
19	0.0	56.0	-150.0	-149.9
21	0.0	55.0	-150.0	-150.0
23	0.0	54.0	-150.0	-150.0
25	0.0	53.0	-150.0	-150.0
27	0.0	52.0	-150.0	-150.0
29	0.0	51.0	-150.0	-150.0
31	0.0	50.0	-150.0	-150.0
33	0.0	49.0	-150.0	-150.0
35	0.0	48.0	-150.0	-150.0
37	0.0	47.0	-150.0	-150.0
39	0.0	46.0	-150.0	-150.0
41	0.0	45.0	-150.0	-150.0
43	0.0	44.0	-150.0	-150.0
45	0.0	43.0	-150.0	-150.0
47	0.0	42.0	-150.0	-150.0
49	0.0	41.0	-150.0	-150.0
51	0.0	40.0	-150.0	-150.0
53	0.0	39.0	-150.0	-150.0
55	0.0	38.0	-150.0	-150.0
57	0.0	37.0	-150.0	-150.0
59	0.0	36.0	-150.0	-150.0
61	0.0	35.0	-150.0	-150.0
63	0.0	34.0	-150.0	-150.0
65	0.0	33.0	-150.0	-150.0
67	0.0	32.0	-150.0	-150.0
69	0.0	31.0	-150.0	-150.0
71	0.0	30.0	-150.0	-150.0
73	0.0	29.0	-150.0	-150.0
75	0.0	28.0	-150.0	-150.0
77	0.0	27.0	-150.0	-150.0
79	0.0	26.0	-150.0	-150.0
81	0.0	25.0	-150.0	-150.0
83	0.0	24.0	-150.0	-150.0
85	0.0	23.0	-150.0	-150.0

TABLE 7.3. Output Data, Example Problem #1, Output File 'H.OUT'.
(Continuation)

87	0.0	22.0	-150.0	-150.0
89	0.0	21.0	-150.0	-150.0
91	0.0	20.0	-150.0	-150.0
93	0.0	18.0	-150.0	-150.0
95	0.0	16.0	-150.0	-150.0
97	0.0	14.0	-150.0	-150.0
99	0.0	12.0	-150.0	-150.0
101	0.0	10.0	-150.0	-150.0
103	0.0	8.0	-150.0	-150.0
105	0.0	6.0	-150.0	-150.0
107	0.0	4.0	-150.0	-150.0
109	0.0	2.0	-150.0	-150.0
111	0.0	0.0	-150.0	-150.0

Time *** 900.0000 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	61.0	0.8	0.8
3	0.0	60.7	0.2	0.2
5	0.0	60.5	-0.3	-0.3
7	0.0	60.2	-0.8	-0.8
9	0.0	60.0	-1.3	-1.3
11	0.0	59.5	-2.4	-2.4
13	0.0	59.0	-3.4	-3.4
15	0.0	58.0	-5.5	-5.5
17	0.0	57.0	-7.6	-7.6
19	0.0	56.0	-9.7	-9.7
21	0.0	55.0	-11.8	-11.8
23	0.0	54.0	-13.8	-13.8
25	0.0	53.0	-15.7	-15.8
27	0.0	52.0	-17.9	-17.9
29	0.0	51.0	-20.7	-20.7
31	0.0	50.0	-24.8	-24.9
33	0.0	49.0	-31.9	-32.1
35	0.0	48.0	-47.2	-47.0
37	0.0	47.0	-94.8	-89.2
39	0.0	46.0	-147.9	-146.1
41	0.0	45.0	-150.0	-150.0
43	0.0	44.0	-150.0	-150.0
45	0.0	43.0	-150.0	-150.0
47	0.0	42.0	-150.0	-150.0
49	0.0	41.0	-150.0	-150.0
51	0.0	40.0	-150.0	-150.0
53	0.0	39.0	-150.0	-150.0
55	0.0	38.0	-150.0	-150.0
57	0.0	37.0	-150.0	-150.0
59	0.0	36.0	-150.0	-150.0
61	0.0	35.0	-150.0	-150.0
63	0.0	34.0	-150.0	-150.0
65	0.0	33.0	-150.0	-150.0
67	0.0	32.0	-150.0	-150.0
69	0.0	31.0	-150.0	-150.0
71	0.0	30.0	-150.0	-150.0
73	0.0	29.0	-150.0	-150.0

TABLE 7.3. Output Data, Example Problem #1, Output File 'H.OUT'.
(Continuation)

75	0.0	28.0	-150.0	-150.0
77	0.0	27.0	-150.0	-150.0
79	0.0	26.0	-150.0	-150.0
81	0.0	25.0	-150.0	-150.0
83	0.0	24.0	-150.0	-150.0
85	0.0	23.0	-150.0	-150.0
87	0.0	22.0	-150.0	-150.0
89	0.0	21.0	-150.0	-150.0
91	0.0	20.0	-150.0	-150.0
93	0.0	18.0	-150.0	-150.0
95	0.0	16.0	-150.0	-150.0
97	0.0	14.0	-150.0	-150.0
99	0.0	12.0	-150.0	-150.0
101	0.0	10.0	-150.0	-150.0
103	0.0	8.0	-150.0	-150.0
105	0.0	6.0	-150.0	-150.0
107	0.0	4.0	-150.0	-150.0
109	0.0	2.0	-150.0	-150.0
111	0.0	0.0	-149.8	-149.8

Time ### 1800.0000 ###

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	61.0	0.8	0.8
3	0.0	60.7	0.4	0.4
5	0.0	60.5	0.1	0.1
7	0.0	60.2	-0.3	-0.3
9	0.0	60.0	-0.6	-0.6
11	0.0	59.5	-1.3	-1.3
13	0.0	59.0	-2.0	-2.0
15	0.0	58.0	-3.4	-3.4
17	0.0	57.0	-4.8	-4.8
19	0.0	56.0	-6.1	-6.1
21	0.0	55.0	-7.5	-7.5
23	0.0	54.0	-8.9	-8.9
25	0.0	53.0	-10.3	-10.3
27	0.0	52.0	-11.6	-11.6
29	0.0	51.0	-12.9	-12.9
31	0.0	50.0	-14.2	-14.2
33	0.0	49.0	-15.5	-15.5
35	0.0	48.0	-16.8	-16.8
37	0.0	47.0	-18.3	-18.3
39	0.0	46.0	-20.1	-20.2
41	0.0	45.0	-22.7	-22.7
43	0.0	44.0	-26.2	-26.3
45	0.0	43.0	-31.9	-32.0
47	0.0	42.0	-42.3	-42.4
49	0.0	41.0	-67.6	-65.9
51	0.0	40.0	-131.7	-125.1
53	0.0	39.0	-149.7	-149.5
55	0.0	38.0	-150.0	-150.0
57	0.0	37.0	-150.0	-150.0
59	0.0	36.0	-150.0	-150.0
61	0.0	35.0	-150.0	-150.0

TABLE 7.3. Output Data, Example Problem #1, Output File 'H.OUT'.
(Continuation)

63	0.0	34.0	-150.0	-150.0
65	0.0	33.0	-150.0	-150.0
67	0.0	32.0	-150.0	-150.0
69	0.0	31.0	-150.0	-150.0
71	0.0	30.0	-150.0	-150.0
73	0.0	29.0	-150.0	-150.0
75	0.0	28.0	-150.0	-150.0
77	0.0	27.0	-150.0	-150.0
79	0.0	26.0	-150.0	-150.0
81	0.0	25.0	-150.0	-150.0
83	0.0	24.0	-150.0	-150.0
85	0.0	23.0	-150.0	-150.0
87	0.0	22.0	-150.0	-150.0
89	0.0	21.0	-150.0	-150.0
91	0.0	20.0	-150.0	-150.0
93	0.0	18.0	-150.0	-150.0
95	0.0	16.0	-150.0	-150.0
97	0.0	14.0	-150.0	-150.0
99	0.0	12.0	-150.0	-150.0
101	0.0	10.0	-150.0	-150.0
103	0.0	8.0	-150.0	-150.0
105	0.0	6.0	-150.0	-150.0
107	0.0	4.0	-150.0	-150.0
109	0.0	2.0	-150.0	-150.0
111	0.0	0.0	-149.6	-149.7

Time *** 2700.0001 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	61.0	0.8	0.8
3	0.0	60.7	0.5	0.5
5	0.0	60.5	0.2	0.2
7	0.0	60.2	0.0	0.0
9	0.0	60.0	-0.3	-0.3
11	0.0	59.5	-0.8	-0.8
13	0.0	59.0	-1.4	-1.4
15	0.0	58.0	-2.4	-2.4
17	0.0	57.0	-3.5	-3.5
19	0.0	56.0	-4.6	-4.6
21	0.0	55.0	-5.7	-5.7
23	0.0	54.0	-6.7	-6.7
25	0.0	53.0	-7.8	-7.8
27	0.0	52.0	-8.9	-8.9
29	0.0	51.0	-9.9	-9.9
31	0.0	50.0	-11.0	-11.0
33	0.0	49.0	-12.0	-12.0
35	0.0	48.0	-13.0	-13.0
37	0.0	47.0	-14.0	-14.0
39	0.0	46.0	-15.0	-15.0
41	0.0	45.0	-15.9	-15.9
43	0.0	44.0	-16.9	-17.0
45	0.0	43.0	-18.1	-18.1
47	0.0	42.0	-19.5	-19.5
49	0.0	41.0	-21.3	-21.3

TABLE 7.3. Output Data, Example Problem #1, Output File 'H.OUT'.
(Continuation)

51	0.0	40.0	-23.7	-23.7
53	0.0	39.0	-26.9	-27.0
55	0.0	38.0	-31.9	-32.0
57	0.0	37.0	-40.3	-40.4
59	0.0	36.0	-58.2	-57.8
61	0.0	35.0	-108.8	-102.6
63	0.0	34.0	-148.5	-147.1
65	0.0	33.0	-150.0	-150.0
67	0.0	32.0	-150.0	-150.0
69	0.0	31.0	-150.0	-150.0
71	0.0	30.0	-150.0	-150.0
73	0.0	29.0	-150.0	-150.0
75	0.0	28.0	-150.0	-150.0
77	0.0	27.0	-150.0	-150.0
79	0.0	26.0	-150.0	-150.0
81	0.0	25.0	-150.0	-150.0
83	0.0	24.0	-150.0	-150.0
85	0.0	23.0	-150.0	-150.0
87	0.0	22.0	-150.0	-150.0
89	0.0	21.0	-150.0	-150.0
91	0.0	20.0	-150.0	-150.0
93	0.0	18.0	-150.0	-150.0
95	0.0	16.0	-150.0	-150.0
97	0.0	14.0	-150.0	-150.0
99	0.0	12.0	-150.0	-150.0
101	0.0	10.0	-150.0	-150.0
103	0.0	8.0	-150.0	-150.0
105	0.0	6.0	-150.0	-150.0
107	0.0	4.0	-150.0	-150.0
109	0.0	2.0	-150.0	-150.0
111	0.0	0.0	-149.4	-149.5

Time *** 3600.0001 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	61.0	0.8	0.8
3	0.0	60.7	0.5	0.5
5	0.0	60.5	0.3	0.3
7	0.0	60.2	0.1	0.1
9	0.0	60.0	-0.1	-0.1
11	0.0	59.5	-0.6	-0.6
13	0.0	59.0	-1.0	-1.0
15	0.0	58.0	-1.9	-1.9
17	0.0	57.0	-2.8	-2.8
19	0.0	56.0	-3.7	-3.7
21	0.0	55.0	-4.6	-4.6
23	0.0	54.0	-5.5	-5.5
25	0.0	53.0	-6.4	-6.4
27	0.0	52.0	-7.3	-7.3
29	0.0	51.0	-8.1	-8.1
31	0.0	50.0	-9.0	-9.0
33	0.0	49.0	-9.9	-9.9
35	0.0	48.0	-10.7	-10.7
37	0.0	47.0	-11.6	-11.6

TABLE 7.3. Output Data, Example Problem #1, Output File 'H.OUT'.
(Continuation)

39	0.0	46.0	-12.4	-12.4
41	0.0	45.0	-13.3	-13.3
43	0.0	44.0	-14.1	-14.1
45	0.0	43.0	-14.9	-14.9
47	0.0	42.0	-15.7	-15.7
49	0.0	41.0	-16.5	-16.5
51	0.0	40.0	-17.4	-17.4
53	0.0	39.0	-18.4	-18.4
55	0.0	38.0	-19.6	-19.6
57	0.0	37.0	-21.1	-21.1
59	0.0	36.0	-23.1	-23.1
61	0.0	35.0	-25.7	-25.7
63	0.0	34.0	-29.4	-29.5
65	0.0	33.0	-35.0	-35.2
67	0.0	32.0	-45.1	-45.2
69	0.0	31.0	-67.8	-66.7
71	0.0	30.0	-125.9	-119.2
73	0.0	29.0	-149.4	-149.0
75	0.0	28.0	-150.0	-150.0
77	0.0	27.0	-150.0	-150.0
79	0.0	26.0	-150.0	-150.0
81	0.0	25.0	-150.0	-150.0
83	0.0	24.0	-150.0	-150.0
85	0.0	23.0	-150.0	-150.0
87	0.0	22.0	-150.0	-150.0
89	0.0	21.0	-150.0	-150.0
91	0.0	20.0	-150.0	-150.0
93	0.0	18.0	-150.0	-150.0
95	0.0	16.0	-150.0	-150.0
97	0.0	14.0	-150.0	-150.0
99	0.0	12.0	-150.0	-150.0
101	0.0	10.0	-150.0	-150.0
103	0.0	8.0	-150.0	-150.0
105	0.0	6.0	-150.0	-150.0
107	0.0	4.0	-150.0	-150.0
109	0.0	2.0	-149.9	-149.9
111	0.0	0.0	-149.3	-149.4

Time *** 5400.0002 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	61.0	0.8	0.8
3	0.0	60.7	0.6	0.6
5	0.0	60.5	0.4	0.4
7	0.0	60.2	0.2	0.2
9	0.0	60.0	0.1	0.1
11	0.0	59.5	-0.3	-0.3
13	0.0	59.0	-0.6	-0.6
15	0.0	58.0	-1.3	-1.3
17	0.0	57.0	-1.9	-1.9
19	0.0	56.0	-2.6	-2.6
21	0.0	55.0	-3.3	-3.3
23	0.0	54.0	-4.0	-4.0
25	0.0	53.0	-4.7	-4.7

TABLE 7.3. Output Data, Example Problem #1, Output File 'H.OUT'.
(Continuation)

27	0.0	52.0	-5.4	-5.4
29	0.0	51.0	-6.0	-6.0
31	0.0	50.0	-6.7	-6.7
33	0.0	49.0	-7.4	-7.4
35	0.0	48.0	-8.1	-8.1
37	0.0	47.0	-8.7	-8.7
39	0.0	46.0	-9.4	-9.4
41	0.0	45.0	-10.1	-10.1
43	0.0	44.0	-10.7	-10.7
45	0.0	43.0	-11.4	-11.4
47	0.0	42.0	-12.0	-12.0
49	0.0	41.0	-12.6	-12.6
51	0.0	40.0	-13.3	-13.3
53	0.0	39.0	-13.9	-13.9
55	0.0	38.0	-14.5	-14.5
57	0.0	37.0	-15.1	-15.1
59	0.0	36.0	-15.7	-15.7
61	0.0	35.0	-16.3	-16.3
63	0.0	34.0	-16.9	-16.9
65	0.0	33.0	-17.6	-17.6
67	0.0	32.0	-18.4	-18.4
69	0.0	31.0	-19.3	-19.3
71	0.0	30.0	-20.5	-20.5
73	0.0	29.0	-21.9	-21.9
75	0.0	28.0	-23.7	-23.8
77	0.0	27.0	-26.1	-26.1
79	0.0	26.0	-29.3	-29.3
81	0.0	25.0	-34.0	-34.0
83	0.0	24.0	-41.5	-41.7
85	0.0	23.0	-56.3	-56.2
87	0.0	22.0	-94.2	-90.0
89	0.0	21.0	-145.0	-141.7
91	0.0	20.0	-149.9	-149.9
93	0.0	18.0	-150.0	-150.0
95	0.0	16.0	-150.0	-150.0
97	0.0	14.0	-150.0	-150.0
99	0.0	12.0	-150.0	-150.0
101	0.0	10.0	-150.0	-150.0
103	0.0	8.0	-150.0	-150.0
105	0.0	6.0	-150.0	-150.0
107	0.0	4.0	-150.0	-150.0
109	0.0	2.0	-149.9	-149.9
111	0.0	0.0	-149.0	-149.1

TABLE 7.4. Output at the End of Simulation, Example Problem #1, Output File 'TH.OUT'.

Time *** 5400.0002 ***

n	x(n)	z(n)	th(n)	th(n+1) ...
1	0.0	61.0	0.350	0.350
3	0.0	60.7	0.350	0.350
5	0.0	60.5	0.350	0.350
7	0.0	60.2	0.350	0.350
9	0.0	60.0	0.350	0.350
11	0.0	59.5	0.350	0.350
13	0.0	59.0	0.350	0.350
15	0.0	58.0	0.349	0.349
17	0.0	57.0	0.349	0.349
19	0.0	56.0	0.348	0.348
21	0.0	55.0	0.346	0.346
23	0.0	54.0	0.345	0.345
25	0.0	53.0	0.343	0.343
27	0.0	52.0	0.341	0.341
29	0.0	51.0	0.339	0.339
31	0.0	50.0	0.336	0.336
33	0.0	49.0	0.334	0.334
35	0.0	48.0	0.331	0.331
37	0.0	47.0	0.328	0.328
39	0.0	46.0	0.325	0.325
41	0.0	45.0	0.322	0.322
43	0.0	44.0	0.318	0.318
45	0.0	43.0	0.315	0.315
47	0.0	42.0	0.312	0.312
49	0.0	41.0	0.308	0.308
51	0.0	40.0	0.305	0.305
53	0.0	39.0	0.302	0.302
55	0.0	38.0	0.298	0.298
57	0.0	37.0	0.295	0.295
59	0.0	36.0	0.292	0.292
61	0.0	35.0	0.288	0.288
63	0.0	34.0	0.285	0.285
65	0.0	33.0	0.281	0.281
67	0.0	32.0	0.276	0.276
69	0.0	31.0	0.271	0.271
71	0.0	30.0	0.264	0.264
73	0.0	29.0	0.256	0.256
75	0.0	28.0	0.247	0.247
77	0.0	27.0	0.235	0.234
79	0.0	26.0	0.219	0.219
81	0.0	25.0	0.199	0.198
83	0.0	24.0	0.171	0.171
85	0.0	23.0	0.132	0.132
87	0.0	22.0	0.077	0.081
89	0.0	21.0	0.045	0.047
91	0.0	20.0	0.043	0.043
93	0.0	18.0	0.043	0.043
95	0.0	16.0	0.043	0.043
97	0.0	14.0	0.043	0.043
99	0.0	12.0	0.043	0.043
101	0.0	10.0	0.043	0.043
103	0.0	8.0	0.043	0.043

TABLE 7.4. Output at the End of Simulation, Example Problem #1, Output File 'TH.OUT'.
(Continuation)

105	0.0	6.0	0.043	0.043
107	0.0	4.0	0.043	0.043
109	0.0	2.0	0.043	0.043
111	0.0	0.0	0.044	0.044

TABLE 7.5. Output at the End of Simulation, Example Problem #1, Output File 'Q.OUT'.

Time *** 5400.0002 ***

n	x(n)	z(n)	Q(n)	Q(n+1) ...
1	0.0	61.0	0.60E-03	0.60E-03
3	0.0	60.7	0.00E+00	0.00E+00
5	0.0	60.5	0.00E+00	0.00E+00
7	0.0	60.2	0.00E+00	0.00E+00
9	0.0	60.0	0.00E+00	0.00E+00
11	0.0	59.5	0.00E+00	0.00E+00
13	0.0	59.0	0.00E+00	0.00E+00
15	0.0	58.0	0.00E+00	0.00E+00
17	0.0	57.0	0.00E+00	0.00E+00
19	0.0	56.0	0.00E+00	0.00E+00
21	0.0	55.0	0.00E+00	0.00E+00
23	0.0	54.0	0.00E+00	0.00E+00
25	0.0	53.0	0.00E+00	0.00E+00
27	0.0	52.0	0.00E+00	0.00E+00
29	0.0	51.0	0.00E+00	0.00E+00
31	0.0	50.0	0.00E+00	0.00E+00
33	0.0	49.0	0.00E+00	0.00E+00
35	0.0	48.0	0.00E+00	0.00E+00
37	0.0	47.0	0.00E+00	0.00E+00
39	0.0	46.0	0.00E+00	0.00E+00
41	0.0	45.0	0.00E+00	0.00E+00
43	0.0	44.0	0.00E+00	0.00E+00
45	0.0	43.0	0.00E+00	0.00E+00
47	0.0	42.0	0.00E+00	0.00E+00
49	0.0	41.0	0.00E+00	0.00E+00
51	0.0	40.0	0.00E+00	0.00E+00
53	0.0	39.0	0.00E+00	0.00E+00
55	0.0	38.0	0.00E+00	0.00E+00
57	0.0	37.0	0.00E+00	0.00E+00
59	0.0	36.0	0.00E+00	0.00E+00
61	0.0	35.0	0.00E+00	0.00E+00
63	0.0	34.0	0.00E+00	0.00E+00
65	0.0	33.0	0.00E+00	0.00E+00
67	0.0	32.0	0.00E+00	0.00E+00
69	0.0	31.0	0.00E+00	0.00E+00
71	0.0	30.0	0.00E+00	0.00E+00
73	0.0	29.0	0.00E+00	0.00E+00
75	0.0	28.0	0.00E+00	0.00E+00
77	0.0	27.0	0.00E+00	0.00E+00
79	0.0	26.0	0.00E+00	0.00E+00

TABLE 7.5. Output at the End of Simulation, Example Problem #1, Output File 'Q.OUT'. (Continuation)

81	0.0	25.0	0.00E+00	0.00E+00
83	0.0	24.0	0.00E+00	0.00E+00
85	0.0	23.0	0.00E+00	0.00E+00
87	0.0	22.0	0.00E+00	0.00E+00
89	0.0	21.0	0.00E+00	0.00E+00
91	0.0	20.0	0.00E+00	0.00E+00
93	0.0	18.0	0.00E+00	0.00E+00
95	0.0	16.0	0.00E+00	0.00E+00
97	0.0	14.0	0.00E+00	0.00E+00
99	0.0	12.0	0.00E+00	0.00E+00
101	0.0	10.0	0.00E+00	0.00E+00
103	0.0	8.0	0.00E+00	0.00E+00
105	0.0	6.0	0.00E+00	0.00E+00
107	0.0	4.0	0.00E+00	0.00E+00
109	0.0	2.0	0.00E+00	0.00E+00
111	0.0	0.0	0.00E+00	0.00E+00

TABLE 7.6. Output at the End of Simulation, Example Problem #1, Output File 'VZ.OUT'.

Time *** 5400.0002 ***

e	x(e)	z(e)	vz(e)	vz(e+1) ...
1	0.5	60.9	-0.12E-02	
2	0.5	60.6	-0.12E-02	
3	0.5	60.4	-0.12E-02	
4	0.5	60.1	-0.12E-02	
5	0.5	59.7	-0.12E-02	
6	0.5	59.2	-0.12E-02	
7	0.5	58.5	-0.12E-02	
8	0.5	57.5	-0.12E-02	
9	0.5	56.5	-0.12E-02	
10	0.5	55.5	-0.12E-02	
11	0.5	54.5	-0.12E-02	
12	0.5	53.5	-0.12E-02	
13	0.5	52.5	-0.12E-02	
14	0.5	51.5	-0.12E-02	
15	0.5	50.5	-0.12E-02	
16	0.5	49.5	-0.12E-02	
17	0.5	48.5	-0.12E-02	
18	0.5	47.5	-0.12E-02	
19	0.5	46.5	-0.12E-02	
20	0.5	45.5	-0.12E-02	
21	0.5	44.5	-0.12E-02	
22	0.5	43.5	-0.12E-02	
23	0.5	42.5	-0.12E-02	
24	0.5	41.5	-0.11E-02	
25	0.5	40.5	-0.11E-02	
26	0.5	39.5	-0.11E-02	
27	0.5	38.5	-0.11E-02	
28	0.5	37.5	-0.11E-02	

TABLE 7.6. Output at the End of Simulation, Example Problem #1, Output File 'VZ.OUT'.
(Continuation)

29	0.5	36.5	-0.11E-02
30	0.5	35.5	-0.11E-02
31	0.5	34.5	-0.11E-02
32	0.5	33.5	-0.11E-02
33	0.5	32.5	-0.10E-02
34	0.5	31.5	-0.10E-02
35	0.5	30.5	-0.10E-02
36	0.5	29.5	-0.98E-03
37	0.5	28.5	-0.94E-03
38	0.5	27.5	-0.90E-03
39	0.5	26.5	-0.84E-03
40	0.5	25.5	-0.77E-03
41	0.5	24.5	-0.66E-03
42	0.5	23.5	-0.52E-03
43	0.5	22.5	-0.31E-03
44	0.5	21.5	-0.42E-04
45	0.5	20.5	-0.75E-06
46	0.5	19.0	-0.88E-07
47	0.5	17.0	-0.84E-07
48	0.5	15.0	-0.84E-07
49	0.5	13.0	-0.84E-07
50	0.5	11.0	-0.84E-07
51	0.5	9.0	-0.84E-07
52	0.5	7.0	-0.84E-07
53	0.5	5.0	-0.84E-07
54	0.5	3.0	-0.80E-07
55	0.5	1.0	-0.52E-07

TABLE 7.7. Output Data, Example Problem #1, Output File 'H_MEAN.OUT'.

Time [T]	hAtm [L]	hRoot [L]	hKode3 [L]	hKode1 [L]	hSeep [L]	t-level
60.0000	0.0	0.0	0.0	0.8	-150.0	66
900.0000	0.0	0.0	0.0	0.8	-149.8	157
1800.0000	0.0	0.0	0.0	0.8	-149.6	203
2700.0001	0.0	0.0	0.0	0.8	-149.5	237
3600.0001	0.0	0.0	0.0	0.8	-149.3	268
5400.0002	0.0	0.0	0.0	0.8	-149.1	321

TABLE 7.8. Output Data, Example Problem #1, Output File 'V_MEAN.OUT'.

Time [T]	rAtm [L/T]	rRoot [L/T]	vAtm [L/T]	vRoot [L/T]	vKode3 [L/T]	vKode1 [L/T]	vSeep [L/T]	t-level
60.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.783E-02	0.000E+00	66
900.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.223E-02	0.000E+00	157
1800.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.171E-02	0.000E+00	203
2700.0001	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.149E-02	0.000E+00	237
3600.0001	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.136E-02	0.000E+00	268
5400.0002	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.121E-02	0.000E+00	321

TABLE 7.9. Output Data, Example Problem #1, Output File 'CUM_Q.OUT'.

Time [T]	CumQAP [L]	CumQRP [L]	CumQA [L]	CumQR [L]	CumQ3 [L]	CumQ1 [L]	CumQS [L]	t-level
60.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.812E+00	0.000E+00	66
900.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.358E+01	0.000E+00	157
1800.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.531E+01	0.000E+00	203
2700.0001	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.673E+01	0.000E+00	237
3600.0001	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.801E+01	0.000E+00	268
5400.0002	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.103E+02	0.000E+00	321

TABLE 7.10. Output Data, Example Problem #1, Output File 'RUN_INF.OUT'.

TLevel	Time	dt	Iter	ItCum
66	0.600E+02	0.275E+01	4	264
157	0.900E+03	0.154E+02	4	608
203	0.180E+04	0.219E+02	4	789
237	0.270E+04	0.284E+02	4	928
268	0.360E+04	0.290E+02	4	1052
321	0.540E+04	0.357E+02	4	1263

Real time [min] 86.0666656494140 64

7.2. EXAMPLE PROBLEM #2 - GRASS FIELD PROBLEM

The atmospheric and groundwater level data from Hupselse Beek watershed were used to simulate water movement in one-dimensional soil profile. Calculations were performed over the period of April 1 to September 30 of the fairly dry year 1982. The results of computations are compared with those obtained using simulation model of the water balance SWATRE (Feddes et al., 1978, Belmans et al., 1983).

The soil profile (see Figure 7.4.) consists of two layers, an A-horizon of 40 cm thickness and BC-horizon extending to a depth of 300 cm. The depth of the root zone is 30 cm. The mean scaled hydraulic functions used to characterize the soil properties of each layer (Hopmans and Stricker, 1987, Cislerova 1987) are presented on Figure 7.5.

As a boundary conditions for the top of the system, actual precipitation and potential transpiration intensities of a grass crop were incorporated as day's totals spred uniformly over day intervals. As a bottom boundary condition, the bottom flux - groundwater level relationship q(h) was used (see Eq. 4.1.). Initially, the groundwater level was set at 55 cm below the soil surface. The initial moisture profile was taken to be in equlibrium with the groundwater level.

The input values of actual precipitation and potential transpiration are presented in Figure 7.6. . The cumulative values of calculated actual transpiration and cumulative discharge from the bottom of the soil profile are shown in Figure 7.7. . The variation of mean pressure head at the soil surface and within the root zone during simulated seasone are presented in Figure 7.8. . Variations of the calculated groundwater table with time are shown in Figure 7.9. . A complete print of input data is provided in Tables 7.11. through 7.14. . Selected output data are tabelated in Tables 7.15. through 7.18. .

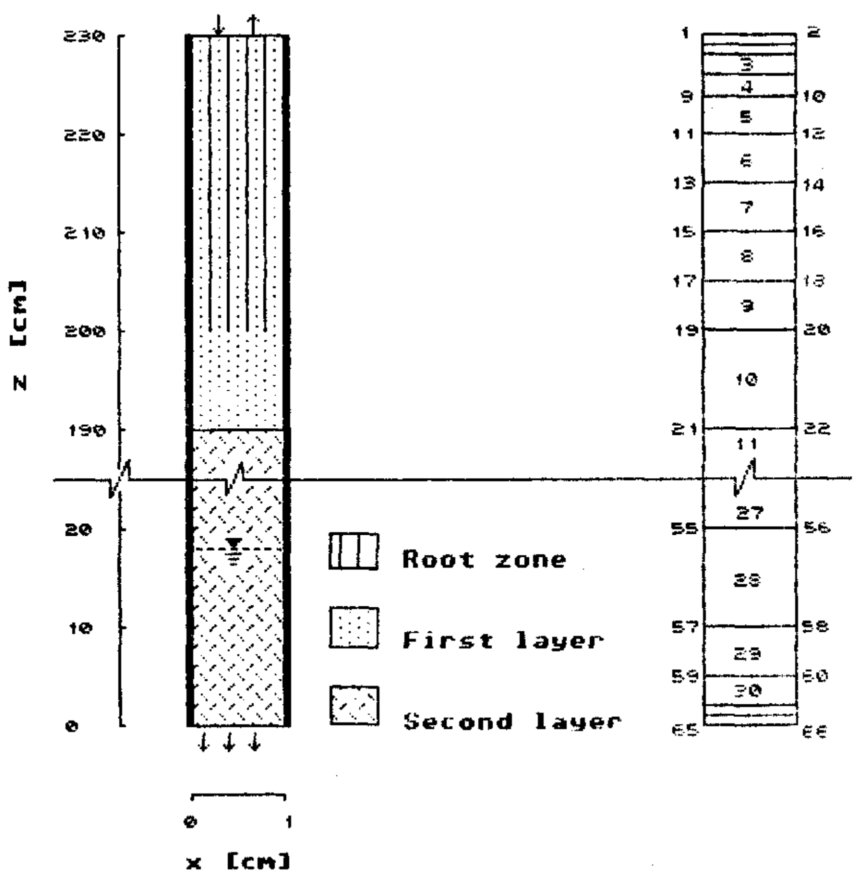


Figure 7.4. Flow system and finite element mesh for Example Problem #2.

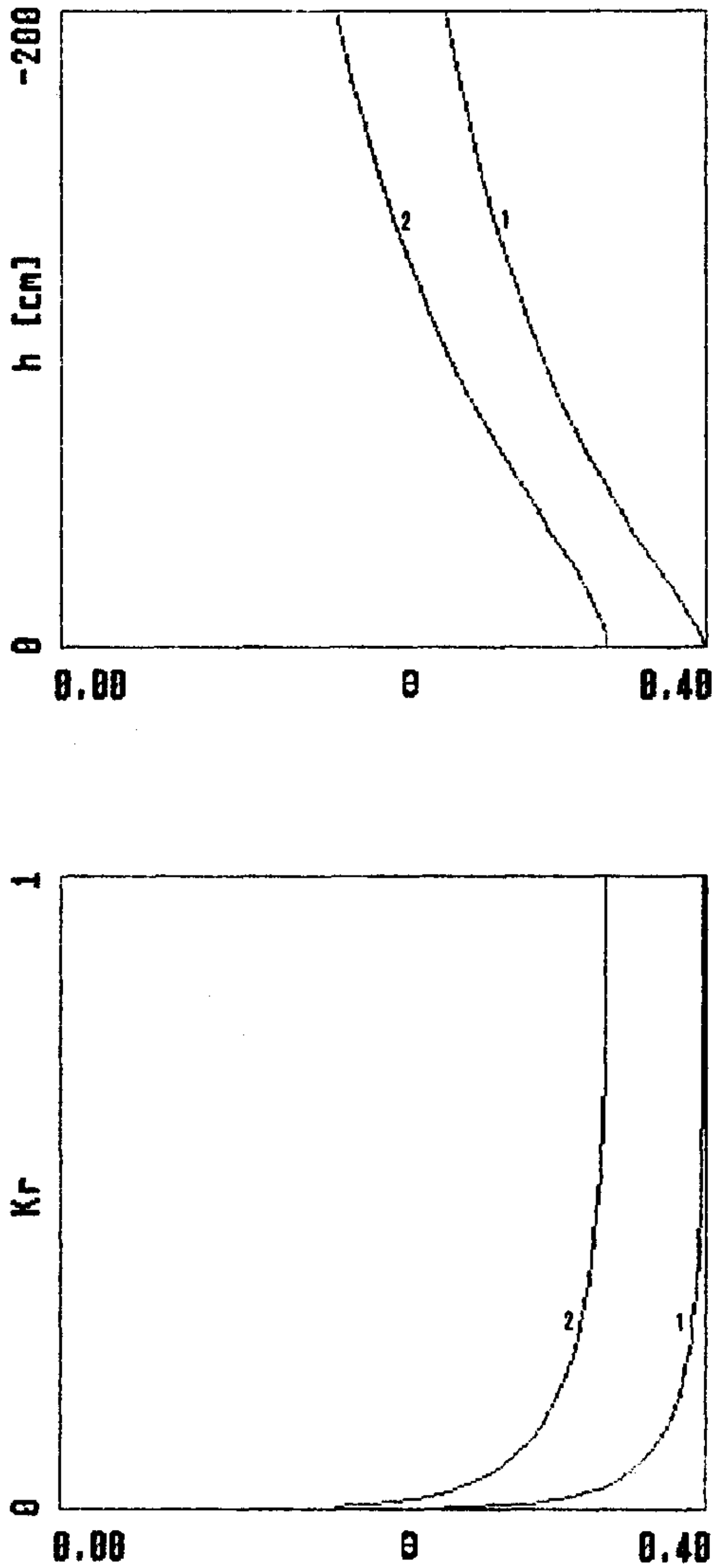


Figure 7.5. Soil hydraulic characteristics of 1st and 2nd layer for Example Problem #2.

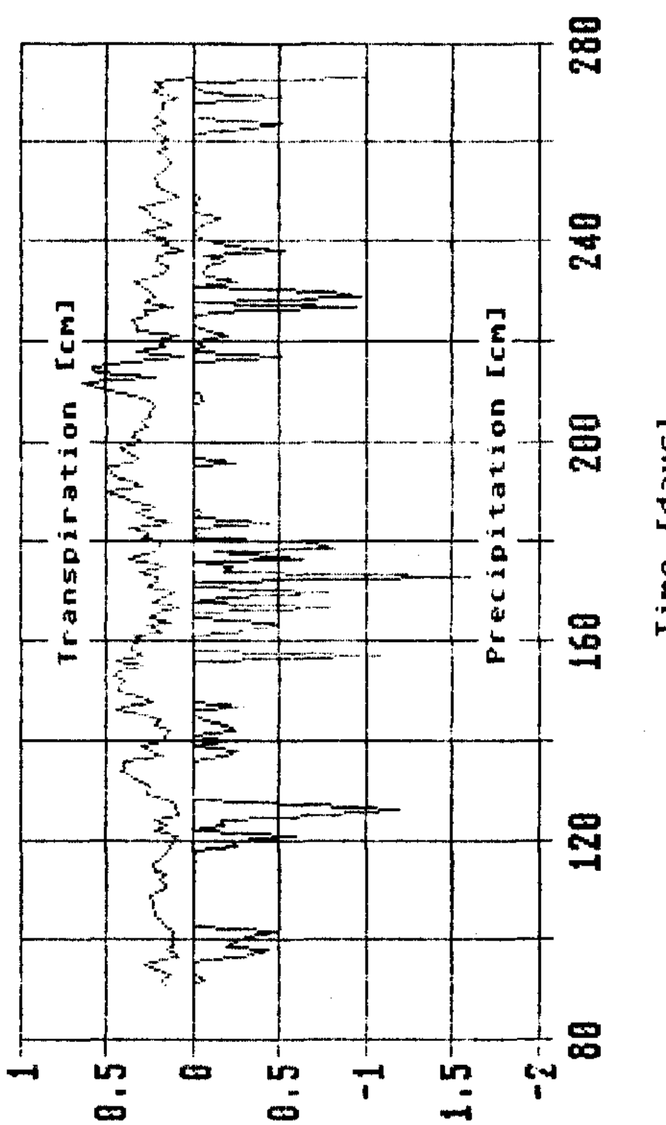


Figure 7.6. Actual precipitation and potential transpiration for Example Problem #2.

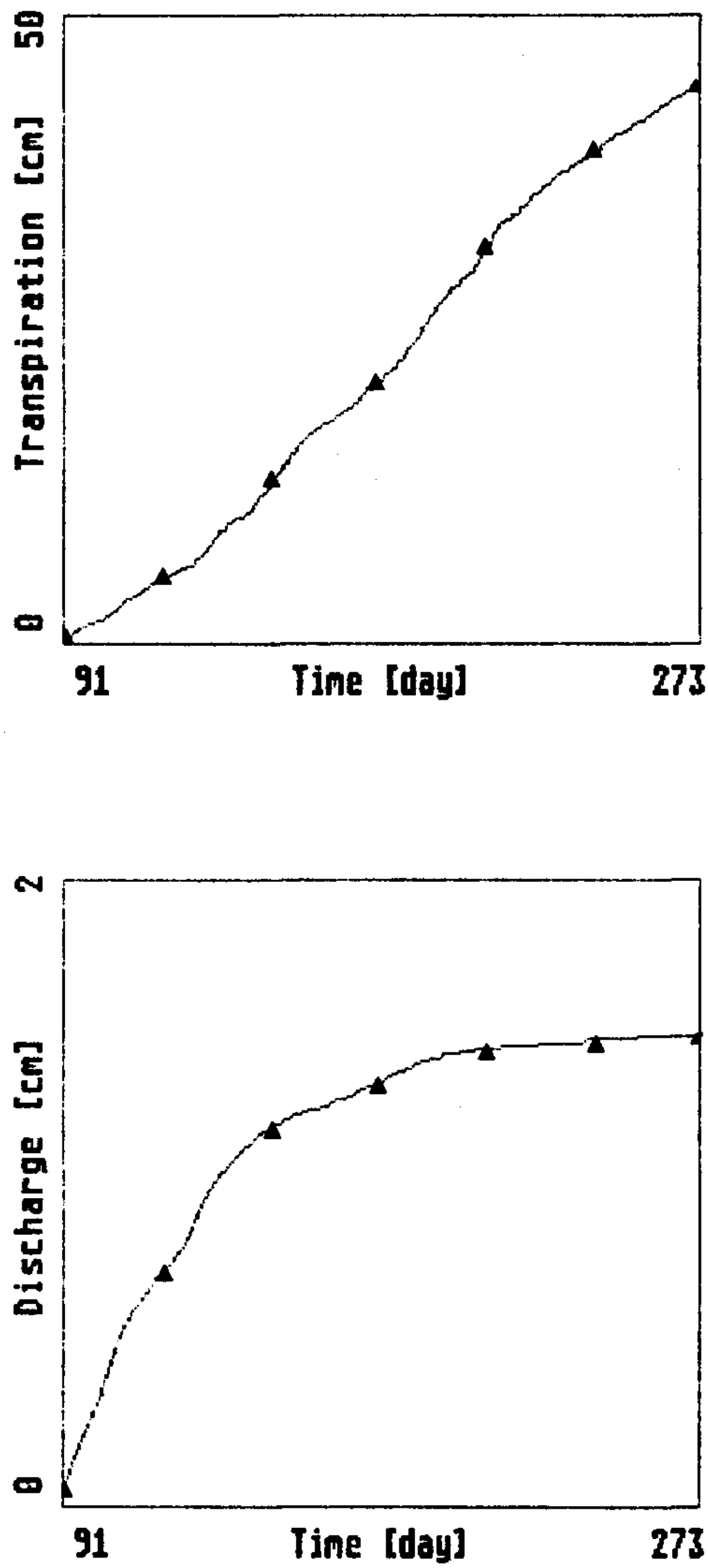


Figure 7.7. Cumulative values of calculated actual transpiration and bottom discharge for Example Problem #2 as simulated by SWM II '___' and SWATRE '▲'.

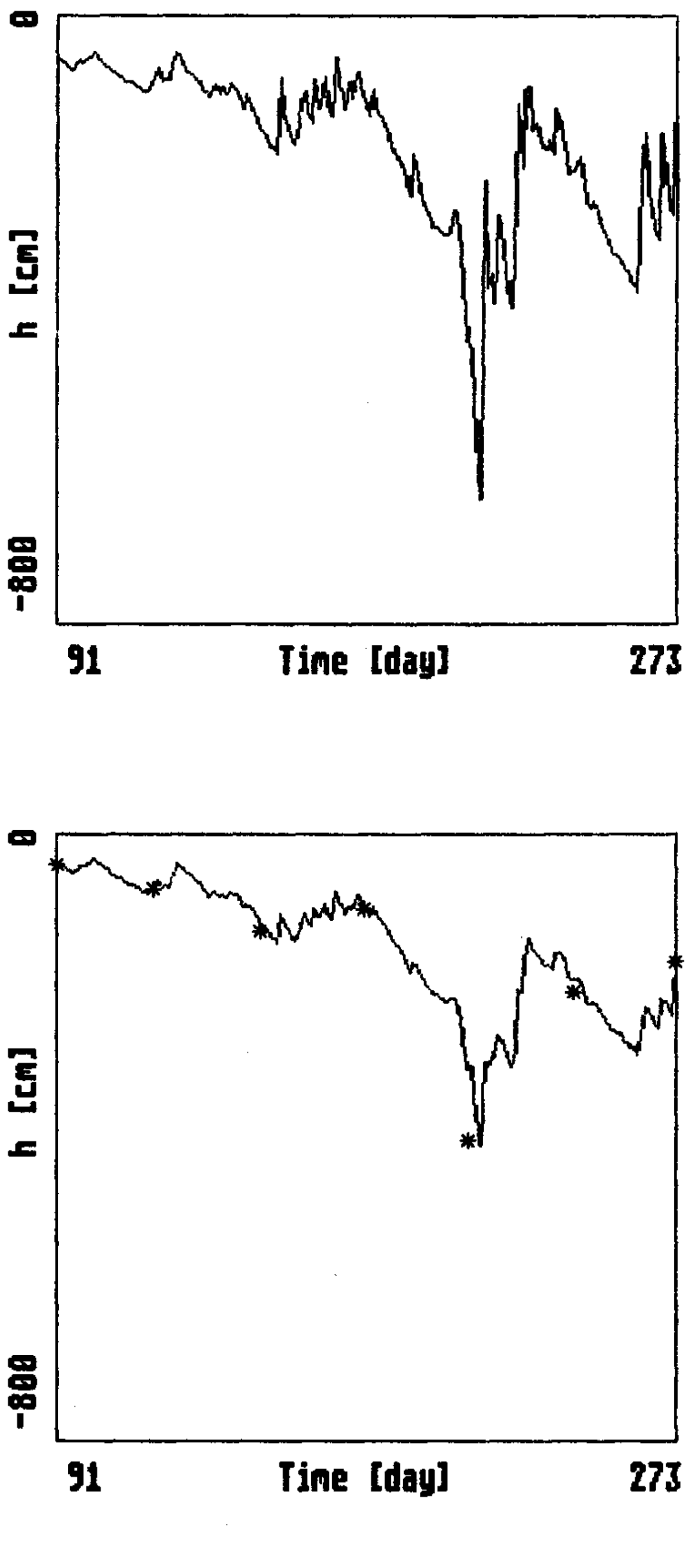


Figure 7.8. Pressure head at the soil surface and mean pressure head within the root zone for Example Problem #2 as simulated by SWM II '___' and SWATRE '*'.

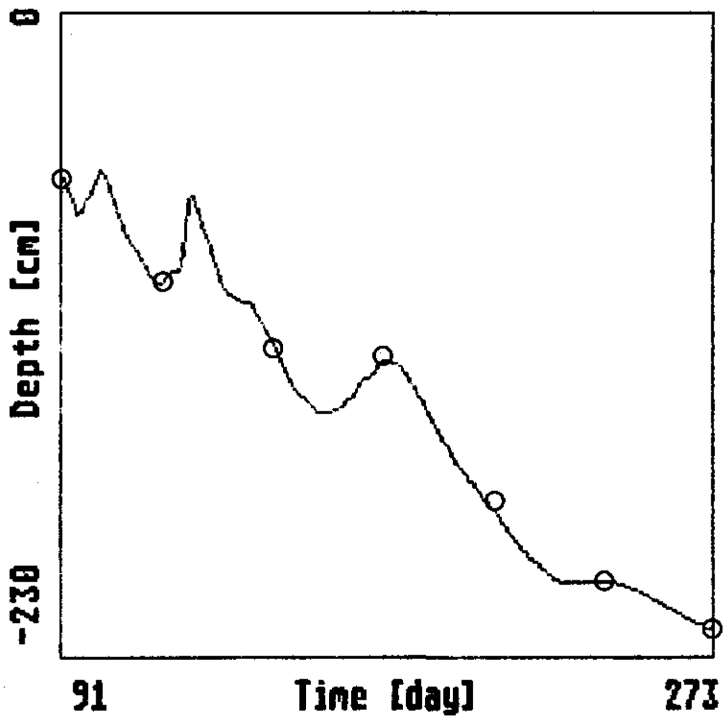


Figure 7.9. Variations of the calculated groundwater table with time for Example Problem #2 as simulated by SWM II '___' and SWATRE 'O'.

TABLE 7.11. Input Data for Example Problem #2, Input File 'SELECTOR.IN'.

```
*** BLOCK A: BASIC INFORMATION ******************************************
Heading
  ' Example Problem #2 - Grass Field Problem (Hupselse Beek 1982)'
LUnit  TUnit        (indicated units are obligatory for all input data)
  'cm'  'day'
Kat (0:horizontal plane, 1:axisymmetric vertical flow, 2:vertical plane)
  2
MaxIt  Tol          (maximum number of iterations and p. head tolerance)
  20    .10
CheckF  ShortF  FluxF  AtmInF  SeepF
  f       t       t      t       f
*** BLOCK B: MATERIAL INFORMATION ***************************************
NMat   NLay   hTab1   hTabN   NPar
  2      2     .001   1000.    9
thr    ths    tha     thm     Alfa    n       Ks      Kk      thk
.0001  .399   .0001   .399    .0174   1.3757  29.75   29.75   .399
.0001  .339   .0001   .339    .0139   1.6024  45.34   45.34   .339
*** BLOCK C: TIME INFORMATION *******************************************
dt     dtMin   dtMax   DMul    DMul2   MPL
 .02   1e-10   0.50    1.3     .3      7
TPrint(1),TPrint(2),...,TPrint(MPL)                    (print-time array)
  91 120 151 181 212 243 273
*** END OF INPUT FILE 'SELECTOR.IN' *************************************
```

TABLE 7.12. Input Data for Example Problem #2, Input File 'SINK.IN'.

```
*** BLOCK I: SINK INFORMATION *******************************************
P0     P2H     P2L     P3      r2H     r2L
 -10.  -200.   -800.   -8000.  0.5     0.1
POptm(1),POptm(2),...,POptm(NMat)
 -25.  -25.
*** END OF INPUT FILE 'SINK.IN' *****************************************
```

TABLE 7.13. Input Data for Example Problem #2, Input File 'ATMOSPH.IN'.

```
*** BLOCK H: ATMOSPHERIC INFORMATION  ***********************************
*** Hupselse Beek 1982                ***********************************
*************************************************************************
SinkF  Dummy  qGWLF
 t      f      t
GWL0L  Aqh    Bqh                          (if qGWLF=f then Aqh=Bqh=0)
 230   -.1687 -.02674
tInit  MaxAL               (MaxAL = number of atmospheric data-records)
 90.    183
hCritS                 (max. allowed pressure head at the soil surface)
 1e30
```

TABLE 7.13. Input Data for Example Problem #2, Input File 'ATMOSPH.IN'.
(continued)

tAtm	Prec	rSoil	rRoot	hCritA	rt	ht
91	0	0	0.16	1000000	0	0
92	0.07	0	0.18	1000000	0	0
93	0.02	0	0.13	1000000	0	0
94	0	0	0.2	1000000	0	0
95	0	0	0.28	1000000	0	0
96	0.07	0	0.18	1000000	0	0
97	0.29	0	0.08	1000000	0	0
98	0.44	0	0.14	1000000	0	0
99	0.2	0	0.11	1000000	0	0
100	0.29	0	0.11	1000000	0	0
101	0.32	0	0.11	1000000	0	0
102	0.49	0	0.11	1000000	0	0
103	0.01	0	0.16	1000000	0	0
104	0	0	0.17	1000000	0	0
105	0	0	0.22	1000000	0	0
106	0	0	0.21	1000000	0	0
107	0	0	0.23	1000000	0	0
108	0	0	0.23	1000000	0	0
109	0	0	0.24	1000000	0	0
110	0	0	0.18	1000000	0	0
111	0	0	0.15	1000000	0	0
112	0	0	0.19	1000000	0	0
113	0.01	0	0.15	1000000	0	0
114	0.01	0	0.22	1000000	0	0
115	0	0	0.23	1000000	0	0
116	0.02	0	0.2	1000000	0	0
117	0	0	0.17	1000000	0	0
118	0.02	0	0.14	1000000	0	0
119	0.26	0	0.13	1000000	0	0
120	0.24	0	0.11	1000000	0	0
121	0.61	0	0.08	1000000	0	0
122	0	0	0.21	1000000	0	0
123	0.19	0	0.14	1000000	0	0
124	0.17	0	0.21	1000000	0	0
125	0.78	0	0.07	1000000	0	0
126	1.18	0	0.1	1000000	0	0
127	0.68	0	0.1	1000000	0	0
128	0	0	0.16	1000000	0	0
129	0	0	0.26	1000000	0	0
130	0	0	0.26	1000000	0	0
131	0	0	0.25	1000000	0	0
132	0	0	0.31	1000000	0	0
133	0	0	0.35	1000000	0	0
134	0	0	0.41	1000000	0	0
135	0	0	0.39	1000000	0	0
136	0	0	0.4	1000000	0	0
137	0.19	0	0.17	1000000	0	0
138	0.26	0	0.25	1000000	0	0
139	0.01	0	0.29	1000000	0	0
140	0.2	0	0.15	1000000	0	0
141	0	0	0.16	1000000	0	0
142	0.25	0	0.12	1000000	0	0
143	0.2	0	0.22	1000000	0	0
144	0.07	0	0.19	1000000	0	0
145	0	0	0.27	1000000	0	0

TABLE 7.13. Input Data for Example Problem #2, Input File 'ATMOSPH.IN'.
(continued)

146	0	0	0.44	1000000	0	0
147	0.29	0	0.39	1000000	0	0
148	0.01	0	0.23	1000000	0	0
149	0	0	0.31	1000000	0	0
150	0	0	0.42	1000000	0	0
151	0	0	0.43	1000000	0	0
152	0	0	0.4	1000000	0	0
153	0	0	0.46	1000000	0	0
154	0	0	0.42	1000000	0	0
155	0	0	0.28	1000000	0	0
156	0	0	0.44	1000000	0	0
157	1.07	0	0.29	1000000	0	0
158	0.01	0	0.36	1000000	0	0
159	0	0	0.36	1000000	0	0
160	0	0	0.3	1000000	0	0
161	0	0	0.34	1000000	0	0
162	0.22	0	0.18	1000000	0	0
163	0.5	0	0.26	1000000	0	0
164	0.46	0	0.12	1000000	0	0
165	0.01	0	0.25	1000000	0	0
166	0	0	0.2	1000000	0	0
167	0.8	0	0.08	1000000	0	0
168	0	0	0.23	1000000	0	0
169	0.19	0	0.15	1000000	0	0
170	0.78	0	0.1	1000000	0	0
171	0.01	0	0.26	1000000	0	0
172	0	0	0.2	1000000	0	0
173	1.56	0	0.2	1000000	0	0
174	0.18	0	0.31	1000000	0	0
175	0.23	0	0.16	1000000	0	0
176	0.02	0	0.3	1000000	0	0
177	0.65	0	0.37	1000000	0	0
178	0.12	0	0.21	1000000	0	0
179	0.82	0	0.2	1000000	0	0
180	0.62	0	0.17	1000000	0	0
181	0.01	0	0.29	1000000	0	0
182	0.04	0	0.24	1000000	0	0
183	0.01	0	0.37	1000000	0	0
184	0.44	0	0.11	1000000	0	0
185	0.02	0	0.26	1000000	0	0
186	0.05	0	0.32	1000000	0	0
187	0.01	0	0.21	1000000	0	0
188	0	0	0.34	1000000	0	0
189	0	0	0.41	1000000	0	0
190	0	0	0.51	1000000	0	0
191	0	0	0.3	1000000	0	0
192	0	0	0.37	1000000	0	0
193	0.01	0	0.46	1000000	0	0
194	0	0	0.47	1000000	0	0
195	0	0	0.48	1000000	0	0
196	0.24	0	0.36	1000000	0	0
197	0.01	0	0.32	1000000	0	0
198	0	0	0.38	1000000	0	0
199	0	0	0.43	1000000	0	0
200	0	0	0.3	1000000	0	0
201	0	0	0.35	1000000	0	0

TABLE 7.13. Input Data for Example Problem #2, Input File 'ATMOSPH.IN'.
(continued)

202	0	0	0.34	1000000	0	0
203	0	0	0.26	1000000	0	0
204	0	0	0.29	1000000	0	0
205	0	0	0.24	1000000	0	0
206	0	0	0.25	1000000	0	0
207	0.01	0	0.21	1000000	0	0
208	0.07	0	0.24	1000000	0	0
209	0.05	0	0.33	1000000	0	0
210	0	0	0.48	1000000	0	0
211	0	0	0.56	1000000	0	0
212	0	0	0.65	1000000	0	0
213	0	0	0.22	1000000	0	0
214	0	0	0.58	1000000	0	0
215	0	0	0.59	1000000	0	0
216	0.01	0	0.38	1000000	0	0
217	0.52	0	0.05	1000000	0	0
218	0	0	0.32	1000000	0	0
219	0.06	0	0.16	1000000	0	0
220	0	0	0.21	1000000	0	0
221	0.2	0	0.09	1000000	0	0
222	0.1	0	0.32	1000000	0	0
223	0	0	0.32	1000000	0	0
224	0	0	0.36	1000000	0	0
225	0.02	0	0.26	1000000	0	0
226	0.31	0	0.27	1000000	0	0
227	0.95	0	0.11	1000000	0	0
228	0	0	0.27	1000000	0	0
229	0.97	0	0.22	1000000	0	0
230	0.78	0	0.24	1000000	0	0
231	0	0	0.3	1000000	0	0
232	0.26	0	0.33	1000000	0	0
233	0.08	0	0.2	1000000	0	0
234	0.06	0	0.15	1000000	0	0
235	0.08	0	0.14	1000000	0	0
236	0.19	0	0.21	1000000	0	0
237	0.07	0	0.2	1000000	0	0
238	0.53	0	0.06	1000000	0	0
239	0.2	0	0.18	1000000	0	0
240	0.08	0	0.13	1000000	0	0
241	0	0	0.23	1000000	0	0
242	0	0	0.3	1000000	0	0
243	0.05	0	0.13	1000000	0	0
244	0.11	0	0.15	1000000	0	0
245	0.16	0	0.23	1000000	0	0
246	0	0	0.25	1000000	0	0
247	0	0	0.31	1000000	0	0
248	0.02	0	0.18	1000000	0	0
249	0.04	0	0.08	1000000	0	0
250	0	0	0.12	1000000	0	0
251	0	0	0.2	1000000	0	0
252	0	0	0.2	1000000	0	0
253	0	0	0.22	1000000	0	0
254	0	0	0.18	1000000	0	0
255	0	0	0.14	1000000	0	0
256	0	0	0.13	1000000	0	0
257	0	0	0.14	1000000	0	0

TABLE 7.13. Input Data for Example Problem #2, Input File 'ATMOSPH.IN'.
(continued)

258	0	0	0.2	1000000	0	0
259	0	0	0.14	1000000	0	0
260	0	0	0.19	1000000	0	0
261	0	0	0.14	1000000	0	0
262	0	0	0.2	1000000	0	0
263	0.35	0	0.23	1000000	0	0
264	0.52	0	0.16	1000000	0	0
265	0	0	0.21	1000000	0	0
266	0	0	0.19	1000000	0	0
267	0	0	0.18	1000000	0	0
268	0	0	0.18	1000000	0	0
269	0.53	0	0.09	1000000	0	0
270	0.07	0	0.23	1000000	0	0
271	0	0	0.17	1000000	0	0
272	0	0	0.22	1000000	0	0
273	1.04	0	0	1000000	0	0

*** END OF INPUT FILE 'ATMOSPH.IN ***

TABLE 7.14. Input Data for Example Problem #2, Input File 'GRID.IN'.

*** BLOCK E: NODAL INFORMATION **

NumNP	NumEl	IJ	NumBP
66	32	2	4

N	Kode	X	Z	P1	Q	M	Beta	Axz	Bxz	Dxz
1	-4	0.00	230.00	-55.00	0.00E+00	1	0.00	1.00	1.00	1.00
2	-4	1.00	230.00	-55.00	0.00E+00	1	0.00	1.00	1.00	1.00
3	0	0.00	229.00	-54.00	0.00E+00	1	0.00	1.00	1.00	1.00
4	0	1.00	229.00	-54.00	0.00E+00	1	0.00	1.00	1.00	1.00
5	0	0.00	228.00	-53.00	0.00E+00	1	1.00	1.00	1.00	1.00
6	0	1.00	228.00	-53.00	0.00E+00	1	1.00	1.00	1.00	1.00
7	0	0.00	226.00	-51.00	0.00E+00	1	1.00	1.00	1.00	1.00
8	0	1.00	226.00	-51.00	0.00E+00	1	1.00	1.00	1.00	1.00
9	0	0.00	224.00	-49.00	0.00E+00	1	1.00	1.00	1.00	1.00
10	0	1.00	224.00	-49.00	0.00E+00	1	1.00	1.00	1.00	1.00
11	0	0.00	220.00	-45.00	0.00E+00	1	1.00	1.00	1.00	1.00
12	0	1.00	220.00	-45.00	0.00E+00	1	1.00	1.00	1.00	1.00
13	0	0.00	215.00	-40.00	0.00E+00	1	1.00	1.00	1.00	1.00
14	0	1.00	215.00	-40.00	0.00E+00	1	1.00	1.00	1.00	1.00
15	0	0.00	210.00	-35.00	0.00E+00	1	1.00	1.00	1.00	1.00
16	0	1.00	210.00	-35.00	0.00E+00	1	1.00	1.00	1.00	1.00
17	0	0.00	205.00	-30.00	0.00E+00	1	1.00	1.00	1.00	1.00
18	0	1.00	205.00	-30.00	0.00E+00	1	1.00	1.00	1.00	1.00
19	0	0.00	200.00	-25.00	0.00E+00	1	1.00	1.00	1.00	1.00
20	0	1.00	200.00	-25.00	0.00E+00	1	1.00	1.00	1.00	1.00
21	0	0.00	190.00	-15.00	0.00E+00	1	0.00	1.00	1.00	1.00
22	0	1.00	190.00	-15.00	0.00E+00	1	0.00	1.00	1.00	1.00
23	0	0.00	180.00	-5.00	0.00E+00	2	0.00	1.00	1.00	1.00
24	0	1.00	180.00	-5.00	0.00E+00	2	0.00	1.00	1.00	1.00
25	0	0.00	170.00	5.00	0.00E+00	2	0.00	1.00	1.00	1.00
26	0	1.00	170.00	5.00	0.00E+00	2	0.00	1.00	1.00	1.00
27	0	0.00	160.00	15.00	0.00E+00	2	0.00	1.00	1.00	1.00
28	0	1.00	160.00	15.00	0.00E+00	2	0.00	1.00	1.00	1.00

TABLE 7.14. Input Data for Example Problem #2, Input File 'GRID.IN'.
(continued)

29	0	0.00	150.00	25.00	0.00E+00	2	0.00	1.00	1.00	1.00
30	0	1.00	150.00	25.00	0.00E+00	2	0.00	1.00	1.00	1.00
31	0	0.00	140.00	35.00	0.00E+00	2	0.00	1.00	1.00	1.00
32	0	1.00	140.00	35.00	0.00E+00	2	0.00	1.00	1.00	1.00
33	0	0.00	130.00	45.00	0.00E+00	2	0.00	1.00	1.00	1.00
34	0	1.00	130.00	45.00	0.00E+00	2	0.00	1.00	1.00	1.00
35	0	0.00	120.00	55.00	0.00E+00	2	0.00	1.00	1.00	1.00
36	0	1.00	120.00	55.00	0.00E+00	2	0.00	1.00	1.00	1.00
37	0	0.00	110.00	65.00	0.00E+00	2	0.00	1.00	1.00	1.00
38	0	1.00	110.00	65.00	0.00E+00	2	0.00	1.00	1.00	1.00
39	0	0.00	100.00	75.00	0.00E+00	2	0.00	1.00	1.00	1.00
40	0	1.00	100.00	75.00	0.00E+00	2	0.00	1.00	1.00	1.00
41	0	0.00	90.00	85.00	0.00E+00	2	0.00	1.00	1.00	1.00
42	0	1.00	90.00	85.00	0.00E+00	2	0.00	1.00	1.00	1.00
43	0	0.00	80.00	95.00	0.00E+00	2	0.00	1.00	1.00	1.00
44	0	1.00	80.00	95.00	0.00E+00	2	0.00	1.00	1.00	1.00
45	0	0.00	70.00	105.00	0.00E+00	2	0.00	1.00	1.00	1.00
46	0	1.00	70.00	105.00	0.00E+00	2	0.00	1.00	1.00	1.00
47	0	0.00	60.00	115.00	0.00E+00	2	0.00	1.00	1.00	1.00
48	0	1.00	60.00	115.00	0.00E+00	2	0.00	1.00	1.00	1.00
49	0	0.00	50.00	125.00	0.00E+00	2	0.00	1.00	1.00	1.00
50	0	1.00	50.00	125.00	0.00E+00	2	0.00	1.00	1.00	1.00
51	0	0.00	40.00	135.00	0.00E+00	2	0.00	1.00	1.00	1.00
52	0	1.00	40.00	135.00	0.00E+00	2	0.00	1.00	1.00	1.00
53	0	0.00	30.00	145.00	0.00E+00	2	0.00	1.00	1.00	1.00
54	0	1.00	30.00	145.00	0.00E+00	2	0.00	1.00	1.00	1.00
55	0	0.00	20.00	155.00	0.00E+00	2	0.00	1.00	1.00	1.00
56	0	1.00	20.00	155.00	0.00E+00	2	0.00	1.00	1.00	1.00
57	0	0.00	10.00	165.00	0.00E+00	2	0.00	1.00	1.00	1.00
58	0	1.00	10.00	165.00	0.00E+00	2	0.00	1.00	1.00	1.00
59	0	0.00	5.00	170.00	0.00E+00	2	0.00	1.00	1.00	1.00
60	0	1.00	5.00	170.00	0.00E+00	2	0.00	1.00	1.00	1.00
61	0	0.00	2.00	173.00	0.00E+00	2	0.00	1.00	1.00	1.00
62	0	1.00	2.00	173.00	0.00E+00	2	0.00	1.00	1.00	1.00
63	0	0.00	1.00	174.00	0.00E+00	2	0.00	1.00	1.00	1.00
64	0	1.00	1.00	174.00	0.00E+00	2	0.00	1.00	1.00	1.00
65	-3	0.00	0.00	175.00	0.00E+00	2	0.00	1.00	1.00	1.00
66	-3	1.00	0.00	175.00	0.00E+00	2	0.00	1.00	1.00	1.00

*** BLOCK F: ELEMENT INFORMATION ***

E	i	j	k	l	Angle	ConA11	ConA33	LayNum
1	1	3	4	2	0.00	1.00	1.00	1
2	3	5	6	4	0.00	1.00	1.00	1
3	5	7	8	6	0.00	1.00	1.00	1
4	7	9	10	8	0.00	1.00	1.00	1
5	9	11	12	10	0.00	1.00	1.00	1
6	11	13	14	12	0.00	1.00	1.00	1
7	13	15	16	14	0.00	1.00	1.00	1
8	15	17	18	16	0.00	1.00	1.00	1
9	17	19	20	18	0.00	1.00	1.00	1
10	19	21	22	20	0.00	1.00	1.00	2
11	21	23	24	22	0.00	1.00	1.00	2
12	23	25	26	24	0.00	1.00	1.00	2
13	25	27	28	26	0.00	1.00	1.00	2
14	27	29	30	28	0.00	1.00	1.00	2
15	29	31	32	30	0.00	1.00	1.00	2
16	31	33	34	32	0.00	1.00	1.00	2

TABLE 7.14. Input Data for Example Problem #2, Input File 'GRID.IN'.
(continued)

```
17  33  35  36  34   0.00   1.00   1.00   2
18  35  37  38  36   0.00   1.00   1.00   2
19  37  39  40  38   0.00   1.00   1.00   2
20  39  41  42  40   0.00   1.00   1.00   2
21  41  43  44  42   0.00   1.00   1.00   2
22  43  45  46  44   0.00   1.00   1.00   2
23  45  47  48  46   0.00   1.00   1.00   2
24  47  49  50  48   0.00   1.00   1.00   2
25  49  51  52  50   0.00   1.00   1.00   2
26  51  53  54  52   0.00   1.00   1.00   2
27  53  55  56  54   0.00   1.00   1.00   2
28  55  57  58  56   0.00   1.00   1.00   2
29  57  59  60  58   0.00   1.00   1.00   2
30  59  61  62  60   0.00   1.00   1.00   2
31  61  63  64  62   0.00   1.00   1.00   2
32  63  65  66  64   0.00   1.00   1.00   2
### BLOCK G: BOUNDARY GEOMETRIC INFORMATION ##############################
KXB(1), KXB(2),...,KXB(NumBP)                (boundary node number array)
  1     2       65      66
Width(1), Width(2),...,Width(NumBP)                         (width array)
  0.50  0.50    0.50    0.50
rLen                               (Length associated with transpiration)
  1.00
### END OF INPUT FILE 'GRID.IN' ##########################################
```

TABLE 7.15. Output Data for Example Problem #2, Output File 'A_LEVEL.OUT'.

Example Problem #2 - Grass Field Problem (Hupselse Beek 1982)

Program SWM II
Date: 2. 5.1988 Time: 16:49
Time dependent boundary conditions
Vertical plane flow, V = L*L
Units: L = cm , T = day

Time [T]	CumQAP [L]	CumQRP [L]	CumQA [L]	CumQR [L]	CumQ3 [L]	hAtm [L]	hRoot [L]	hKode3 [L]	A-level
91.0000	0.000E+00	0.160E+00	0.000E+00	0.160E+00	0.388E-01	-58.3	-37.7	171.8	1
92.0000	-0.700E-01	0.340E+00	-0.700E-01	0.340E+00	0.743E-01	-59.5	-39.9	169.3	2
93.0000	-0.900E-01	0.470E+00	-0.900E-01	0.470E+00	0.108E+00	-62.4	-42.1	167.3	3
94.0000	-0.900E-01	0.670E+00	-0.900E-01	0.670E+00	0.139E+00	-66.4	-45.6	164.3	4
95.0000	-0.900E-01	0.950E+00	-0.900E-01	0.950E+00	0.168E+00	-71.3	-50.1	160.3	5
96.0000	-0.160E+00	0.113E+01	-0.160E+00	0.113E+01	0.194E+00	-71.2	-51.7	157.9	6
97.0000	-0.450E+00	0.121E+01	-0.450E+00	0.121E+01	0.219E+00	-63.6	-48.2	158.9	7
98.0000	-0.890E+00	0.135E+01	-0.890E+00	0.135E+01	0.244E+00	-57.8	-44.1	162.1	8
99.0000	-0.109E+01	0.146E+01	-0.109E+01	0.146E+01	0.272E+00	-60.9	-43.8	164.0	9
100.0000	-0.138E+01	0.157E+01	-0.138E+01	0.157E+01	0.301E+00	-57.8	-41.6	166.0	10
101.0000	-0.170E+01	0.168E+01	-0.170E+01	0.168E+01	0.331E+00	-55.1	-38.9	168.6	11
102.0000	-0.219E+01	0.179E+01	-0.219E+01	0.179E+01	0.364E+00	-48.8	-33.5	173.7	12
103.0000	-0.220E+01	0.195E+01	-0.220E+01	0.195E+01	0.401E+00	-57.1	-36.9	172.4	13
104.0000	-0.220E+01	0.212E+01	-0.220E+01	0.212E+01	0.437E+00	-60.8	-40.2	169.4	14
105.0000	-0.220E+01	0.234E+01	-0.220E+01	0.234E+01	0.471E+00	-64.8	-44.0	165.8	15
106.0000	-0.220E+01	0.255E+01	-0.220E+01	0.255E+01	0.501E+00	-68.4	-47.5	162.5	16
107.0000	-0.220E+01	0.278E+01	-0.220E+01	0.278E+01	0.529E+00	-72.3	-51.2	159.1	17
108.0000	-0.220E+01	0.301E+01	-0.220E+01	0.301E+01	0.554E+00	-76.0	-54.7	155.9	18
109.0000	-0.220E+01	0.325E+01	-0.220E+01	0.325E+01	0.577E+00	-79.7	-58.2	152.7	19
110.0000	-0.220E+01	0.343E+01	-0.220E+01	0.343E+01	0.599E+00	-82.1	-60.7	150.1	20
111.0000	-0.220E+01	0.358E+01	-0.220E+01	0.358E+01	0.618E+00	-84.1	-62.7	148.0	21
112.0000	-0.220E+01	0.377E+01	-0.220E+01	0.377E+01	0.637E+00	-87.0	-65.4	145.7	22
113.0000	-0.221E+01	0.392E+01	-0.221E+01	0.392E+01	0.655E+00	-88.4	-67.2	143.8	23
114.0000	-0.222E+01	0.414E+01	-0.222E+01	0.414E+01	0.672E+00	-91.7	-70.1	141.5	24
115.0000	-0.222E+01	0.437E+01	-0.222E+01	0.437E+01	0.688E+00	-95.6	-73.3	139.1	25
116.0000	-0.224E+01	0.457E+01	-0.224E+01	0.457E+01	0.703E+00	-96.9	-75.3	136.9	26
117.0000	-0.224E+01	0.474E+01	-0.224E+01	0.474E+01	0.717E+00	-99.6	-77.4	135.0	27
118.0000	-0.226E+01	0.488E+01	-0.226E+01	0.488E+01	0.730E+00	-99.7	-78.4	133.4	28
119.0000	-0.252E+01	0.501E+01	-0.252E+01	0.501E+01	0.743E+00	-86.8	-74.3	132.8	29
120.0000	-0.276E+01	0.512E+01	-0.276E+01	0.512E+01	0.755E+00	-84.2	-72.0	133.3	30
121.0000	-0.337E+01	0.520E+01	-0.337E+01	0.520E+01	0.768E+00	-68.2	-63.3	136.1	31
122.0000	-0.337E+01	0.541E+01	-0.337E+01	0.541E+01	0.781E+00	-89.1	-70.4	137.3	32
123.0000	-0.356E+01	0.555E+01	-0.356E+01	0.555E+01	0.796E+00	-84.3	-69.7	137.5	33
124.0000	-0.373E+01	0.576E+01	-0.373E+01	0.576E+01	0.810E+00	-85.8	-70.7	137.2	34
125.0000	-0.451E+01	0.583E+01	-0.451E+01	0.583E+01	0.824E+00	-62.0	-58.2	140.6	35
126.0000	-0.569E+01	0.593E+01	-0.569E+01	0.593E+01	0.839E+00	-48.2	-45.3	151.4	36
127.0000	-0.637E+01	0.603E+01	-0.637E+01	0.603E+01	0.860E+00	-51.2	-40.7	162.9	37
128.0000	-0.637E+01	0.619E+01	-0.637E+01	0.619E+01	0.888E+00	-65.1	-45.1	163.9	38
129.0000	-0.637E+01	0.645E+01	-0.637E+01	0.645E+01	0.917E+00	-70.6	-49.6	160.6	39
130.0000	-0.637E+01	0.671E+01	-0.637E+01	0.671E+01	0.943E+00	-74.9	-53.6	157.0	40
131.0000	-0.637E+01	0.696E+01	-0.637E+01	0.696E+01	0.967E+00	-78.8	-57.3	153.6	41
132.0000	-0.637E+01	0.727E+01	-0.637E+01	0.727E+01	0.989E+00	-83.8	-61.8	149.8	42
133.0000	-0.637E+01	0.762E+01	-0.637E+01	0.762E+01	0.101E+01	-89.4	-66.9	145.9	43
134.0000	-0.637E+01	0.803E+01	-0.637E+01	0.803E+01	0.103E+01	-96.1	-72.7	141.6	44

TABLE 7.15. Output Data for Example Problem #2, Output File 'A_LEVEL.OUT'.
(continued)

135.0000	-0.637E+01	0.842E+01	-0.637E+01	0.842E+01	0.104E+01	-102.0	-77.9	137.5	45
136.0000	-0.637E+01	0.882E+01	-0.637E+01	0.882E+01	0.106E+01	-108.1	-83.2	133.6	46
137.0000	-0.656E+01	0.899E+01	-0.656E+01	0.899E+01	0.107E+01	-94.9	-79.2	131.4	47
138.0000	-0.682E+01	0.924E+01	-0.682E+01	0.924E+01	0.108E+01	-89.7	-77.5	130.4	48
139.0000	-0.683E+01	0.953E+01	-0.683E+01	0.953E+01	0.109E+01	-104.1	-83.2	129.0	49
140.0000	-0.703E+01	0.968E+01	-0.703E+01	0.968E+01	0.110E+01	-94.2	-80.4	128.0	50
141.0000	-0.703E+01	0.984E+01	-0.703E+01	0.984E+01	0.112E+01	-104.5	-83.7	127.2	51
142.0000	-0.728E+01	0.996E+01	-0.728E+01	0.996E+01	0.113E+01	-91.4	-79.7	126.8	52
143.0000	-0.748E+01	0.102E+02	-0.748E+01	0.102E+02	0.114E+01	-92.8	-80.2	126.7	53
144.0000	-0.755E+01	0.104E+02	-0.755E+01	0.104E+02	0.115E+01	-100.4	-83.0	126.2	54
145.0000	-0.755E+01	0.106E+02	-0.755E+01	0.106E+02	0.116E+01	-110.5	-88.5	124.9	55
146.0000	-0.755E+01	0.111E+02	-0.755E+01	0.111E+02	0.117E+01	-121.8	-96.6	122.3	56
147.0000	-0.784E+01	0.115E+02	-0.784E+01	0.115E+02	0.118E+01	-104.9	-93.9	119.7	57
148.0000	-0.785E+01	0.117E+02	-0.785E+01	0.117E+02	0.119E+01	-118.9	-97.0	117.9	58
149.0000	-0.785E+01	0.120E+02	-0.785E+01	0.120E+02	0.120E+01	-127.2	-102.3	115.9	59
150.0000	-0.785E+01	0.124E+02	-0.785E+01	0.124E+02	0.120E+01	-137.6	-109.9	113.4	60
151.0000	-0.785E+01	0.129E+02	-0.785E+01	0.129E+02	0.121E+01	-147.4	-117.0	110.5	61
152.0000	-0.785E+01	0.133E+02	-0.785E+01	0.133E+02	0.122E+01	-155.1	-122.7	107.6	62
153.0000	-0.785E+01	0.137E+02	-0.785E+01	0.137E+02	0.122E+01	-165.7	-130.5	104.6	63
154.0000	-0.785E+01	0.141E+02	-0.785E+01	0.141E+02	0.123E+01	-174.1	-136.6	101.7	64
155.0000	-0.785E+01	0.144E+02	-0.785E+01	0.144E+02	0.124E+01	-174.9	-137.8	98.9	65
156.0000	-0.785E+01	0.149E+02	-0.785E+01	0.149E+02	0.124E+01	-185.3	-145.8	96.3	66
157.0000	-0.892E+01	0.151E+02	-0.892E+01	0.152E+02	0.125E+01	-82.9	-106.4	94.1	67
158.0000	-0.893E+01	0.155E+02	-0.893E+01	0.155E+02	0.125E+01	-131.2	-116.4	93.2	68
159.0000	-0.893E+01	0.159E+02	-0.893E+01	0.159E+02	0.125E+01	-149.7	-126.9	92.3	69
160.0000	-0.893E+01	0.162E+02	-0.893E+01	0.162E+02	0.126E+01	-161.0	-133.9	91.1	70
161.0000	-0.893E+01	0.165E+02	-0.893E+01	0.165E+02	0.126E+01	-172.3	-141.6	89.6	71
162.0000	-0.915E+01	0.167E+02	-0.915E+01	0.167E+02	0.127E+01	-141.3	-133.1	88.1	72
163.0000	-0.965E+01	0.169E+02	-0.965E+01	0.170E+02	0.127E+01	-109.6	-119.4	87.0	73
164.0000	-0.101E+02	0.171E+02	-0.101E+02	0.171E+02	0.127E+01	-99.6	-107.5	86.6	74
165.0000	-0.101E+02	0.173E+02	-0.101E+02	0.173E+02	0.128E+01	-130.6	-117.1	86.8	75
166.0000	-0.101E+02	0.175E+02	-0.101E+02	0.175E+02	0.128E+01	-143.8	-124.3	86.8	76
167.0000	-0.109E+02	0.176E+02	-0.109E+02	0.176E+02	0.128E+01	-85.4	-100.6	86.9	77
168.0000	-0.109E+02	0.178E+02	-0.109E+02	0.178E+02	0.129E+01	-122.8	-111.0	87.6	78
169.0000	-0.111E+02	0.180E+02	-0.111E+02	0.180E+02	0.129E+01	-117.0	-111.6	88.2	79
170.0000	-0.119E+02	0.181E+02	-0.119E+02	0.181E+02	0.130E+01	-80.9	-93.8	89.1	80
171.0000	-0.119E+02	0.183E+02	-0.119E+02	0.183E+02	0.130E+01	-117.2	-106.0	90.5	81
172.0000	-0.119E+02	0.185E+02	-0.119E+02	0.185E+02	0.130E+01	-132.1	-114.7	91.2	82
173.0000	-0.135E+02	0.187E+02	-0.135E+02	0.187E+02	0.131E+01	-57.3	-78.0	92.2	83
174.0000	-0.136E+02	0.190E+02	-0.136E+02	0.190E+02	0.131E+01	-94.9	-90.9	95.4	84
175.0000	-0.139E+02	0.192E+02	-0.139E+02	0.192E+02	0.132E+01	-99.3	-95.0	97.9	85
176.0000	-0.139E+02	0.195E+02	-0.139E+02	0.195E+02	0.132E+01	-121.7	-106.2	98.9	86
177.0000	-0.145E+02	0.199E+02	-0.145E+02	0.199E+02	0.133E+01	-89.4	-97.6	99.1	87
178.0000	-0.147E+02	0.201E+02	-0.147E+02	0.201E+02	0.133E+01	-110.8	-102.0	99.5	88
179.0000	-0.155E+02	0.203E+02	-0.155E+02	0.203E+02	0.134E+01	-76.9	-87.0	100.6	89
180.0000	-0.161E+02	0.205E+02	-0.161E+02	0.205E+02	0.134E+01	-76.6	-81.7	103.1	90
181.0000	-0.161E+02	0.207E+02	-0.161E+02	0.207E+02	0.135E+01	-108.2	-94.7	105.3	91
182.0000	-0.161E+02	0.210E+02	-0.161E+02	0.210E+02	0.135E+01	-118.7	-102.1	105.8	92
183.0000	-0.162E+02	0.214E+02	-0.162E+02	0.214E+02	0.136E+01	-133.7	-111.6	105.0	93
184.0000	-0.166E+02	0.215E+02	-0.166E+02	0.215E+02	0.137E+01	-99.1	-98.8	104.4	94
185.0000	-0.166E+02	0.217E+02	-0.166E+02	0.217E+02	0.137E+01	-123.1	-105.9	104.0	95
186.0000	-0.167E+02	0.220E+02	-0.167E+02	0.220E+02	0.138E+01	-131.6	-112.4	103.1	96
187.0000	-0.167E+02	0.223E+02	-0.167E+02	0.223E+02	0.138E+01	-139.1	-115.8	102.0	97
188.0000	-0.167E+02	0.226E+02	-0.167E+02	0.226E+02	0.139E+01	-150.2	-123.3	100.4	98
189.0000	-0.167E+02	0.230E+02	-0.167E+02	0.230E+02	0.139E+01	-162.9	-132.0	98.4	99
190.0000	-0.167E+02	0.235E+02	-0.167E+02	0.235E+02	0.140E+01	-179.9	-143.6	96.1	100

TABLE 7.15. Output Data for Example Problem #2, Output File 'A_LEVEL.OUT'.
(continued)

191.0000	-0.167E+02	0.238E+02	-0.167E+02	0.238E+02	0.140E+01	-184.4	-146.3	93.6	101
192.0000	-0.167E+02	0.242E+02	-0.167E+02	0.242E+02	0.141E+01	-192.5	-152.3	91.3	102
193.0000	-0.167E+02	0.246E+02	-0.167E+02	0.246E+02	0.141E+01	-204.2	-161.7	88.8	103
194.0000	-0.167E+02	0.251E+02	-0.167E+02	0.251E+02	0.142E+01	-221.2	-172.0	86.3	104
195.0000	-0.167E+02	0.256E+02	-0.167E+02	0.256E+02	0.142E+01	-238.7	-182.9	83.8	105
196.0000	-0.169E+02	0.260E+02	-0.169E+02	0.260E+02	0.142E+01	-183.2	-172.6	81.3	106
197.0000	-0.169E+02	0.263E+02	-0.169E+02	0.263E+02	0.143E+01	-214.3	-175.8	79.0	107
198.0000	-0.169E+02	0.267E+02	-0.169E+02	0.267E+02	0.143E+01	-231.7	-184.0	76.9	108
199.0000	-0.169E+02	0.271E+02	-0.169E+02	0.271E+02	0.143E+01	-250.2	-195.1	74.8	109
200.0000	-0.169E+02	0.274E+02	-0.169E+02	0.274E+02	0.144E+01	-257.6	-198.5	72.8	110
201.0000	-0.169E+02	0.277E+02	-0.169E+02	0.277E+02	0.144E+01	-268.2	-205.5	70.8	111
202.0000	-0.169E+02	0.281E+02	-0.169E+02	0.281E+02	0.144E+01	-278.6	-212.1	68.8	112
203.0000	-0.169E+02	0.283E+02	-0.169E+02	0.283E+02	0.144E+01	-281.5	-213.7	66.9	113
204.0000	-0.169E+02	0.286E+02	-0.169E+02	0.286E+02	0.144E+01	-287.0	-217.9	65.1	114
205.0000	-0.169E+02	0.289E+02	-0.169E+02	0.289E+02	0.145E+01	-288.3	-219.1	63.4	115
206.0000	-0.169E+02	0.291E+02	-0.169E+02	0.291E+02	0.145E+01	-290.6	-221.5	61.7	116
207.0000	-0.169E+02	0.293E+02	-0.169E+02	0.293E+02	0.145E+01	-284.6	-220.6	60.1	117
208.0000	-0.170E+02	0.296E+02	-0.170E+02	0.296E+02	0.145E+01	-258.6	-217.7	58.5	118
209.0000	-0.171E+02	0.299E+02	-0.171E+02	0.299E+02	0.145E+01	-264.4	-222.9	57.0	119
210.0000	-0.171E+02	0.304E+02	-0.171E+02	0.304E+02	0.146E+01	-307.8	-243.2	55.5	120
211.0000	-0.171E+02	0.309E+02	-0.171E+02	0.309E+02	0.146E+01	-356.5	-271.5	53.9	121
212.0000	-0.171E+02	0.316E+02	-0.171E+02	0.316E+02	0.146E+01	-426.6	-311.9	52.2	122
213.0000	-0.171E+02	0.318E+02	-0.171E+02	0.318E+02	0.146E+01	-443.8	-308.7	50.4	123
214.0000	-0.171E+02	0.324E+02	-0.171E+02	0.324E+02	0.146E+01	-509.9	-347.6	48.6	124
215.0000	-0.171E+02	0.330E+02	-0.171E+02	0.329E+02	0.146E+01	-599.8	-393.5	46.8	125
216.0000	-0.171E+02	0.334E+02	-0.171E+02	0.333E+02	0.146E+01	-639.2	-411.3	45.0	126
217.0000	-0.176E+02	0.334E+02	-0.176E+02	0.334E+02	0.147E+01	-219.9	-303.2	43.2	127
218.0000	-0.176E+02	0.337E+02	-0.176E+02	0.337E+02	0.147E+01	-358.8	-307.5	41.5	128
219.0000	-0.177E+02	0.339E+02	-0.177E+02	0.338E+02	0.147E+01	-340.3	-295.2	39.9	129
220.0000	-0.177E+02	0.341E+02	-0.177E+02	0.341E+02	0.147E+01	-379.8	-296.8	38.4	130
221.0000	-0.179E+02	0.342E+02	-0.179E+02	0.341E+02	0.147E+01	-263.5	-266.9	37.0	131
222.0000	-0.180E+02	0.345E+02	-0.180E+02	0.345E+02	0.147E+01	-279.6	-269.7	35.7	132
223.0000	-0.180E+02	0.348E+02	-0.180E+02	0.348E+02	0.147E+01	-337.6	-283.1	34.5	133
224.0000	-0.180E+02	0.352E+02	-0.180E+02	0.351E+02	0.147E+01	-378.5	-302.0	33.3	134
225.0000	-0.180E+02	0.354E+02	-0.180E+02	0.354E+02	0.147E+01	-386.1	-308.8	32.1	135
226.0000	-0.183E+02	0.357E+02	-0.183E+02	0.357E+02	0.147E+01	-241.9	-284.3	30.9	136
227.0000	-0.192E+02	0.358E+02	-0.192E+02	0.358E+02	0.147E+01	-116.8	-206.2	29.7	137
228.0000	-0.192E+02	0.361E+02	-0.192E+02	0.361E+02	0.148E+01	-203.6	-209.7	28.6	138
229.0000	-0.202E+02	0.363E+02	-0.202E+02	0.363E+02	0.148E+01	-99.0	-164.8	27.6	139
230.0000	-0.210E+02	0.366E+02	-0.210E+02	0.365E+02	0.148E+01	-93.3	-137.8	26.7	140
231.0000	-0.210E+02	0.369E+02	-0.210E+02	0.368E+02	0.148E+01	-151.2	-152.6	26.1	141
232.0000	-0.212E+02	0.372E+02	-0.212E+02	0.371E+02	0.148E+01	-141.2	-158.8	25.8	142
233.0000	-0.213E+02	0.374E+02	-0.213E+02	0.373E+02	0.148E+01	-163.2	-165.9	25.8	143
234.0000	-0.214E+02	0.375E+02	-0.214E+02	0.375E+02	0.148E+01	-174.7	-171.5	25.9	144
235.0000	-0.215E+02	0.377E+02	-0.215E+02	0.376E+02	0.148E+01	-177.6	-175.1	25.9	145
236.0000	-0.217E+02	0.379E+02	-0.217E+02	0.378E+02	0.148E+01	-165.1	-175.7	26.0	146
237.0000	-0.217E+02	0.381E+02	-0.217E+02	0.380E+02	0.148E+01	-183.2	-181.7	26.0	147
238.0000	-0.223E+02	0.381E+02	-0.223E+02	0.381E+02	0.148E+01	-123.3	-157.7	26.0	148
239.0000	-0.225E+02	0.383E+02	-0.225E+02	0.383E+02	0.148E+01	-141.2	-157.4	26.0	149
240.0000	-0.225E+02	0.385E+02	-0.225E+02	0.384E+02	0.148E+01	-158.7	-161.5	26.0	150
241.0000	-0.225E+02	0.387E+02	-0.225E+02	0.386E+02	0.149E+01	-185.7	-174.4	26.1	151
242.0000	-0.225E+02	0.390E+02	-0.225E+02	0.389E+02	0.149E+01	-209.1	-190.5	26.2	152
243.0000	-0.226E+02	0.391E+02	-0.226E+02	0.391E+02	0.149E+01	-206.2	-193.1	26.3	153
244.0000	-0.227E+02	0.393E+02	-0.227E+02	0.392E+02	0.149E+01	-194.8	-192.9	26.2	154
245.0000	-0.229E+02	0.395E+02	-0.229E+02	0.395E+02	0.149E+01	-185.6	-193.9	26.0	155
246.0000	-0.229E+02	0.397E+02	-0.229E+02	0.397E+02	0.149E+01	-223.0	-205.5	25.8	156

TABLE 7.15. Output Data for Example Problem #2, Output File 'A_LEVEL.OUT'. (continued)

247.0000	-0.229E+02	0.401E+02	-0.229E+02	0.400E+02	0.149E+01	-249.1	-221.3	25.5	157
248.0000	-0.229E+02	0.402E+02	-0.229E+02	0.402E+02	0.149E+01	-254.9	-226.5	25.1	158
249.0000	-0.229E+02	0.403E+02	-0.229E+02	0.403E+02	0.149E+01	-246.3	-223.2	24.7	159
250.0000	-0.229E+02	0.404E+02	-0.229E+02	0.404E+02	0.149E+01	-259.5	-226.0	24.2	160
251.0000	-0.229E+02	0.406E+02	-0.229E+02	0.406E+02	0.149E+01	-272.6	-234.5	23.7	161
252.0000	-0.229E+02	0.408E+02	-0.229E+02	0.408E+02	0.149E+01	-285.5	-242.7	23.2	162
253.0000	-0.229E+02	0.411E+02	-0.229E+02	0.410E+02	0.149E+01	-300.0	-252.2	22.6	163
254.0000	-0.229E+02	0.412E+02	-0.229E+02	0.412E+02	0.149E+01	-310.5	-258.2	22.0	164
255.0000	-0.229E+02	0.414E+02	-0.229E+02	0.413E+02	0.149E+01	-315.9	-260.8	21.4	165
256.0000	-0.229E+02	0.415E+02	-0.229E+02	0.415E+02	0.150E+01	-319.3	-262.6	20.7	166
257.0000	-0.229E+02	0.416E+02	-0.229E+02	0.416E+02	0.150E+01	-323.4	-265.3	20.1	167
258.0000	-0.229E+02	0.418E+02	-0.229E+02	0.418E+02	0.150E+01	-334.3	-273.3	19.4	168
259.0000	-0.229E+02	0.420E+02	-0.229E+02	0.419E+02	0.150E+01	-339.1	-275.6	18.7	169
260.0000	-0.229E+02	0.422E+02	-0.229E+02	0.421E+02	0.150E+01	-349.2	-282.6	18.0	170
261.0000	-0.229E+02	0.423E+02	-0.229E+02	0.423E+02	0.150E+01	-353.7	-284.7	17.3	171
262.0000	-0.229E+02	0.425E+02	-0.229E+02	0.425E+02	0.150E+01	-365.1	-292.8	16.6	172
263.0000	-0.233E+02	0.427E+02	-0.233E+02	0.427E+02	0.150E+01	-214.8	-268.6	15.9	173
264.0000	-0.238E+02	0.429E+02	-0.238E+02	0.429E+02	0.150E+01	-154.8	-229.8	15.2	174
265.0000	-0.238E+02	0.431E+02	-0.238E+02	0.431E+02	0.150E+01	-231.4	-234.1	14.5	175
266.0000	-0.238E+02	0.433E+02	-0.238E+02	0.433E+02	0.150E+01	-259.2	-242.1	13.8	176
267.0000	-0.238E+02	0.435E+02	-0.238E+02	0.434E+02	0.150E+01	-278.7	-250.1	13.1	177
268.0000	-0.238E+02	0.437E+02	-0.238E+02	0.436E+02	0.150E+01	-295.0	-258.2	12.4	178
269.0000	-0.243E+02	0.438E+02	-0.243E+02	0.437E+02	0.150E+01	-155.6	-218.9	11.8	179
270.0000	-0.244E+02	0.440E+02	-0.244E+02	0.439E+02	0.150E+01	-208.1	-222.5	11.2	180
271.0000	-0.244E+02	0.442E+02	-0.244E+02	0.441E+02	0.150E+01	-241.1	-229.8	10.7	181
272.0000	-0.244E+02	0.444E+02	-0.244E+02	0.443E+02	0.150E+01	-264.7	-241.5	10.1	182
273.0000	-0.254E+02	0.444E+02	-0.254E+02	0.443E+02	0.150E+01	-100.7	-174.4	9.6	183

TABLE 7.16. Output Data for Example Problem #2, Output File 'BALANCE.OUT'.

```
Time [T]          Total     Sub-region number ...

     91.0000                     1           2
Area   [V]     0.230E+03  0.300E+02  0.200E+03
Volume [V]     0.787E+02  0.104E+02  0.683E+02
InFlow [V/T]  -0.199E+00 -0.115E+00 -0.841E-01
hMean  [L]                    -42.7        72.4

    120.0000                     1           2
Area   [V]     0.230E+03  0.300E+02  0.200E+03
Volume [V]     0.758E+02  0.935E+01  0.664E+02
InFlow [V/T]   0.117E+00  0.591E-01  0.582E-01
hMean  [L]                    -77.2        33.0

    151.0000                     1           2
Area   [V]     0.230E+03  0.300E+02  0.200E+03
Volume [V]     0.727E+02  0.824E+01  0.645E+02
InFlow [V/T]  -0.437E+00 -0.144E+00 -0.293E+00
hMean  [L]                   -122.5        11.5

    181.0000                     1           2
Area   [V]     0.230E+03  0.300E+02  0.200E+03
Volume [V]     0.729E+02  0.880E+01  0.641E+02
InFlow [V/T]  -0.284E+00 -0.356E+00  0.717E-01
hMean  [L]                    -90.4         5.1

    212.0000                     1           2
Area   [V]     0.230E+03  0.300E+02  0.200E+03
Volume [V]     0.630E+02  0.597E+01  0.570E+02
InFlow [V/T]  -0.636E+00 -0.304E+00 -0.332E+00
hMean  [L]                   -327.4       -52.2

    243.0000                     1           2
Area   [V]     0.230E+03  0.300E+02  0.200E+03
Volume [V]     0.610E+02  0.717E+01  0.538E+02
InFlow [V/T]  -0.802E-01 -0.291E-01 -0.511E-01
hMean  [L]                   -198.2       -74.2

    273.0000                     1           2
Area   [V]     0.230E+03  0.300E+02  0.200E+03
Volume [V]     0.585E+02  0.770E+01  0.508E+02
InFlow [V/T]   0.103E+01  0.988E+00  0.458E-01
hMean  [L]                   -174.4       -94.1
```

TABLE 7.17. Output Data for Example Problem #2, Output File 'H.OUT'.

Time ### 91.0000 ###

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	230.0	-58.3	-58.3
3	0.0	229.0	-57.3	-57.3
5	0.0	228.0	-56.3	-56.3
7	0.0	226.0	-54.3	-54.3
9	0.0	224.0	-52.3	-52.3
11	0.0	220.0	-48.3	-48.3
13	0.0	215.0	-43.2	-43.2
15	0.0	210.0	-38.2	-38.2
17	0.0	205.0	-33.1	-33.1
19	0.0	200.0	-28.1	-28.1
21	0.0	190.0	-18.1	-18.1
23	0.0	180.0	-8.1	-8.1
25	0.0	170.0	1.9	1.9
27	0.0	160.0	11.9	11.9
29	0.0	150.0	21.9	21.9
31	0.0	140.0	31.9	31.9
33	0.0	130.0	41.9	41.9
35	0.0	120.0	51.9	51.9
37	0.0	110.0	61.9	61.9
39	0.0	100.0	71.9	71.9
41	0.0	90.0	81.9	81.9
43	0.0	80.0	91.8	91.8
45	0.0	70.0	101.8	101.8
47	0.0	60.0	111.8	111.8
49	0.0	50.0	121.8	121.8
51	0.0	40.0	131.8	131.8
53	0.0	30.0	141.8	141.8
55	0.0	20.0	151.8	151.8
57	0.0	10.0	161.8	161.8
59	0.0	5.0	166.8	166.8
61	0.0	2.0	169.8	169.8
63	0.0	1.0	170.8	170.8
65	0.0	0.0	171.8	171.8

Time ### 120.0000 ###

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	230.0	-84.2	-84.2
3	0.0	229.0	-83.9	-83.9
5	0.0	228.0	-83.5	-83.5
7	0.0	226.0	-82.7	-82.7
9	0.0	224.0	-81.8	-81.8
11	0.0	220.0	-79.8	-79.8
13	0.0	215.0	-76.8	-76.8
15	0.0	210.0	-73.4	-73.4
17	0.0	205.0	-69.6	-69.6
19	0.0	200.0	-65.4	-65.4
21	0.0	190.0	-56.4	-56.4
23	0.0	180.0	-46.6	-46.6
25	0.0	170.0	-36.6	-36.6
27	0.0	160.0	-26.7	-26.7

TABLE 7.17. Output Data for Example Problem #2, Output File 'H.OUT'.
(continued)

29	0.0	150.0	-16.7	-16.7
31	0.0	140.0	-6.7	-6.7
33	0.0	130.0	3.3	3.3
35	0.0	120.0	13.3	13.3
37	0.0	110.0	23.3	23.3
39	0.0	100.0	33.3	33.3
41	0.0	90.0	43.3	43.3
43	0.0	80.0	53.3	53.3
45	0.0	70.0	63.3	63.3
47	0.0	60.0	73.3	73.3
49	0.0	50.0	83.3	83.3
51	0.0	40.0	93.3	93.3
53	0.0	30.0	103.3	103.3
55	0.0	20.0	113.3	113.3
57	0.0	10.0	123.3	123.3
59	0.0	5.0	128.3	128.3
61	0.0	2.0	131.3	131.3
63	0.0	1.0	132.3	132.3
65	0.0	0.0	133.3	133.3

Time *** 151.0000 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	230.0	-147.4	-147.4
3	0.0	229.0	-146.4	-146.4
5	0.0	228.0	-145.5	-145.4
7	0.0	226.0	-143.4	-143.4
9	0.0	224.0	-141.0	-141.0
11	0.0	220.0	-135.6	-135.6
13	0.0	215.0	-127.7	-127.7
15	0.0	210.0	-119.0	-119.0
17	0.0	205.0	-109.9	-109.9
19	0.0	200.0	-100.5	-100.5
21	0.0	190.0	-82.8	-82.8
23	0.0	180.0	-71.0	-71.0
25	0.0	170.0	-60.3	-60.3
27	0.0	160.0	-49.8	-49.8
29	0.0	150.0	-39.6	-39.6
31	0.0	140.0	-29.5	-29.5
33	0.0	130.0	-19.4	-19.4
35	0.0	120.0	-9.4	-9.4
37	0.0	110.0	0.6	0.6
39	0.0	100.0	10.6	10.6
41	0.0	90.0	20.6	20.6
43	0.0	80.0	30.6	30.6
45	0.0	70.0	40.6	40.6
47	0.0	60.0	50.6	50.6
49	0.0	50.0	60.5	60.5
51	0.0	40.0	70.5	70.5
53	0.0	30.0	80.5	80.5
55	0.0	20.0	90.5	90.5
57	0.0	10.0	100.5	100.5
59	0.0	5.0	105.5	105.5

TABLE 7.17. Output Data for Example Problem #2, Output File 'H.OUT'.
(continued)

61	0.0	2.0	108.5	108.5
63	0.0	1.0	109.5	109.5
65	0.0	0.0	110.5	110.5

Time *** 181.0000 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	230.0	-108.2	-108.2
3	0.0	229.0	-107.2	-107.2
5	0.0	228.0	-106.4	-106.4
7	0.0	226.0	-104.8	-104.8
9	0.0	224.0	-103.3	-103.3
11	0.0	220.0	-100.7	-100.7
13	0.0	215.0	-97.7	-97.7
15	0.0	210.0	-95.0	-95.0
17	0.0	205.0	-92.2	-92.2
19	0.0	200.0	-89.2	-89.2
21	0.0	190.0	-82.5	-82.5
23	0.0	180.0	-73.5	-73.5
25	0.0	170.0	-64.0	-64.0
27	0.0	160.0	-54.3	-54.3
29	0.0	150.0	-44.5	-44.5
31	0.0	140.0	-34.6	-34.6
33	0.0	130.0	-24.7	-24.7
35	0.0	120.0	-14.7	-14.7
37	0.0	110.0	-4.7	-4.7
39	0.0	100.0	5.3	5.3
41	0.0	90.0	15.3	15.3
43	0.0	80.0	25.3	25.3
45	0.0	70.0	35.3	35.3
47	0.0	60.0	45.3	45.3
49	0.0	50.0	55.3	55.3
51	0.0	40.0	65.3	65.3
53	0.0	30.0	75.3	75.3
55	0.0	20.0	85.3	85.3
57	0.0	10.0	95.3	95.3
59	0.0	5.0	100.3	100.3
61	0.0	2.0	103.3	103.3
63	0.0	1.0	104.3	104.3
65	0.0	0.0	105.3	105.3

Time *** 212.0000 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	230.0	-426.5	-426.7
3	0.0	229.0	-426.7	-426.6
5	0.0	228.0	-428.4	-428.2
7	0.0	226.0	-429.3	-429.3
9	0.0	224.0	-425.8	-425.7
11	0.0	220.0	-406.9	-406.9
13	0.0	215.0	-367.6	-367.6
15	0.0	210.0	-319.7	-319.7

TABLE 7.17. Output Data for Example Problem #2, Output File 'H.OUT'.
(continued)

17	0.0	205.0	-271.8	-271.8
19	0.0	200.0	-228.3	-228.4
21	0.0	190.0	-166.7	-166.8
23	0.0	180.0	-144.1	-144.1
25	0.0	170.0	-128.4	-128.4
27	0.0	160.0	-114.7	-114.7
29	0.0	150.0	-102.2	-102.2
31	0.0	140.0	-90.5	-90.5
33	0.0	130.0	-79.4	-79.4
35	0.0	120.0	-68.7	-68.7
37	0.0	110.0	-58.3	-58.3
39	0.0	100.0	-48.1	-48.1
41	0.0	90.0	-37.9	-37.9
43	0.0	80.0	-27.9	-27.9
45	0.0	70.0	-17.9	-17.9
47	0.0	60.0	-7.8	-7.8
49	0.0	50.0	2.2	2.2
51	0.0	40.0	12.2	12.2
53	0.0	30.0	22.2	22.2
55	0.0	20.0	32.2	32.2
57	0.0	10.0	42.2	42.2
59	0.0	5.0	47.2	47.2
61	0.0	2.0	50.2	50.2
63	0.0	1.0	51.2	51.2
65	0.0	0.0	52.2	52.2

Time *** 243.0000 ***

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	230.0	-206.2	-206.2
3	0.0	229.0	-206.4	-206.4
5	0.0	228.0	-206.5	-206.5
7	0.0	226.0	-206.7	-206.7
9	0.0	224.0	-206.5	-206.5
11	0.0	220.0	-205.1	-205.1
13	0.0	215.0	-201.8	-201.8
15	0.0	210.0	-197.0	-197.0
17	0.0	205.0	-190.9	-190.9
19	0.0	200.0	-183.7	-183.7
21	0.0	190.0	-167.7	-167.7
23	0.0	180.0	-155.7	-155.7
25	0.0	170.0	-144.8	-144.8
27	0.0	160.0	-134.3	-134.3
29	0.0	150.0	-124.0	-124.0
31	0.0	140.0	-113.9	-113.9
33	0.0	130.0	-103.8	-103.8
35	0.0	120.0	-93.8	-93.8
37	0.0	110.0	-83.7	-83.7
39	0.0	100.0	-73.7	-73.7
41	0.0	90.0	-63.7	-63.7
43	0.0	80.0	-53.7	-53.7
45	0.0	70.0	-43.7	-43.7
47	0.0	60.0	-33.7	-33.7
49	0.0	50.0	-23.7	-23.7

TABLE 7.17. Output Data for Example Problem #2, Output File 'H.OUT'.
(continued)

51	0.0	40.0	-13.7	-13.7
53	0.0	30.0	-3.7	-3.7
55	0.0	20.0	6.3	6.3
57	0.0	10.0	16.3	16.3
59	0.0	5.0	21.3	21.3
61	0.0	2.0	24.3	24.3
63	0.0	1.0	25.3	25.3
65	0.0	0.0	26.3	26.3

Time ### 273.0000 ###

n	x(n)	z(n)	h(n)	h(n+1) ...
1	0.0	230.0	-100.7	-100.7
3	0.0	229.0	-103.9	-103.9
5	0.0	228.0	-107.4	-107.4
7	0.0	226.0	-114.9	-114.9
9	0.0	224.0	-123.2	-123.2
11	0.0	220.0	-142.1	-142.1
13	0.0	215.0	-167.7	-167.7
15	0.0	210.0	-190.2	-190.1
17	0.0	205.0	-204.0	-204.0
19	0.0	200.0	-208.1	-208.1
21	0.0	190.0	-196.0	-196.0
23	0.0	180.0	-182.6	-182.6
25	0.0	170.0	-170.0	-170.0
27	0.0	160.0	-157.7	-157.7
29	0.0	150.0	-145.8	-145.8
31	0.0	140.0	-134.3	-134.3
33	0.0	130.0	-123.1	-123.1
35	0.0	120.0	-112.3	-112.3
37	0.0	110.0	-101.6	-101.6
39	0.0	100.0	-91.2	-91.2
41	0.0	90.0	-80.9	-80.9
43	0.0	80.0	-70.7	-70.7
45	0.0	70.0	-60.5	-60.5
47	0.0	60.0	-50.5	-50.5
49	0.0	50.0	-40.4	-40.4
51	0.0	40.0	-30.4	-30.4
53	0.0	30.0	-20.4	-20.4
55	0.0	20.0	-10.4	-10.4
57	0.0	10.0	-0.4	-0.4
59	0.0	5.0	4.6	4.6
61	0.0	2.0	7.6	7.6
63	0.0	1.0	8.6	8.6
65	0.0	0.0	9.6	9.6

TABLE 7.18. Output Data for Example Problem #2, Output File 'RUN_INF.OUT'.

TLevel	Time	dt	Iter	ItCum
11	0.910E+02	0.173E+00	2	21
70	0.120E+03	0.500E+00	3	154
132	0.151E+03	0.500E+00	3	329
202	0.181E+03	0.500E+00	3	579
264	0.212E+03	0.500E+00	4	773
356	0.243E+03	0.500E+00	2	1122
433	0.273E+03	0.167E+00	5	1381

Real time [min] 52.5666666603088374

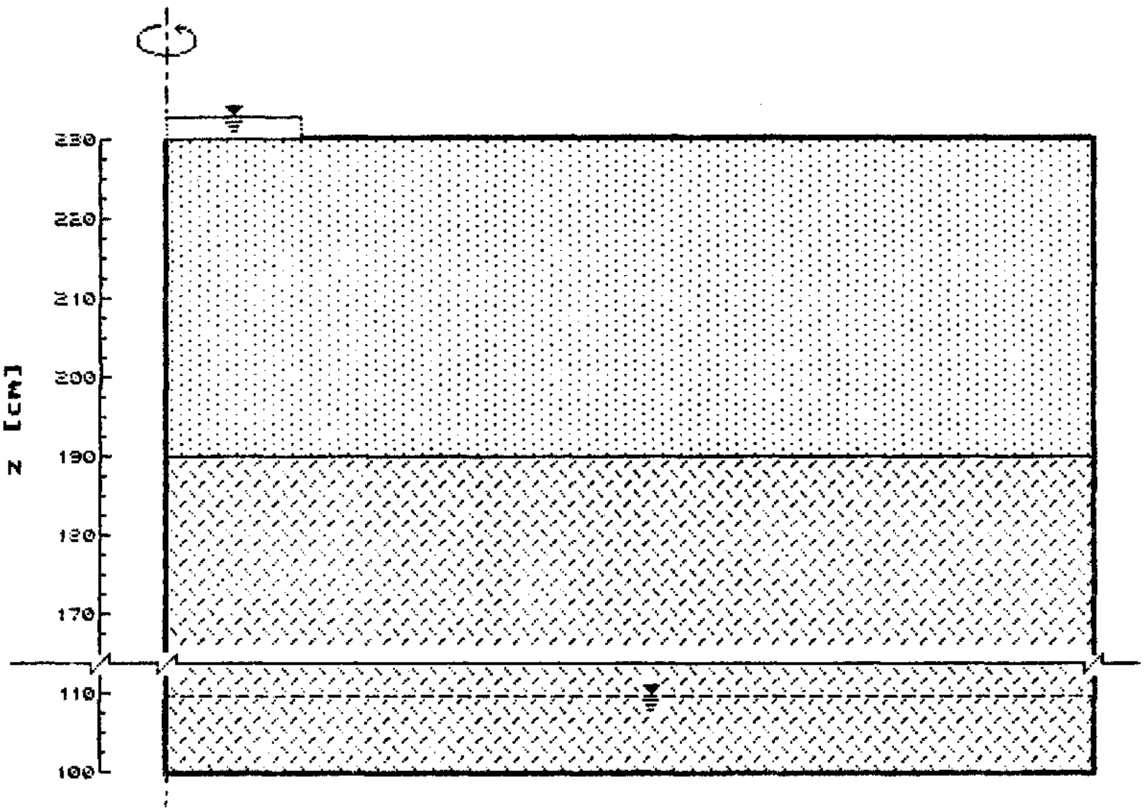
7.3. EXAMPLE PROBLEM #3 - INFILTRATION TEST

Upper part of the soil profile from Example Problem #2 was used for numerical simulation of single ring ponded infiltration experiment. The flow system and finite element mesh are shown in figure 7.10. .

The axisymmetric quasi-three-dimensional program option was employed for analysis of unsaturated water flow during the infiltration. Calculations were performed over the period of 6 hours. As an initial condition, the pressure head profile from the beginning of June 82 obtained in example problem #2 was used. All sides of the flow region were set impervious except of small portion of the surface boundary on which constant zero pressure head boundary condition was imposed indicating the ponded surface inside the infiltration ring. The groundwater table was fixed during the simulation at approximately constant level of 120 cm under the surface. Influence of the root zone extraction was neglected.

The advancement of wetting front during the infiltration is shown in Figure 7.11. . Pressure heads and moisture contents at the surface boundary at the end of simulation are presented in Figure 7.12. . Dependence of the boundary Darcian velocity inside the infiltration ring on the distance from the symmetry axis is presented in Figure 7.13. . Infiltration rate and cumulative infiltration rate as functions of time are shown in Figure 7.14..

Complete print of input data is provided in Tables 7.19. and 7.20. . Partial print of output data is presented in Tables 7.21. trough 7.30. .



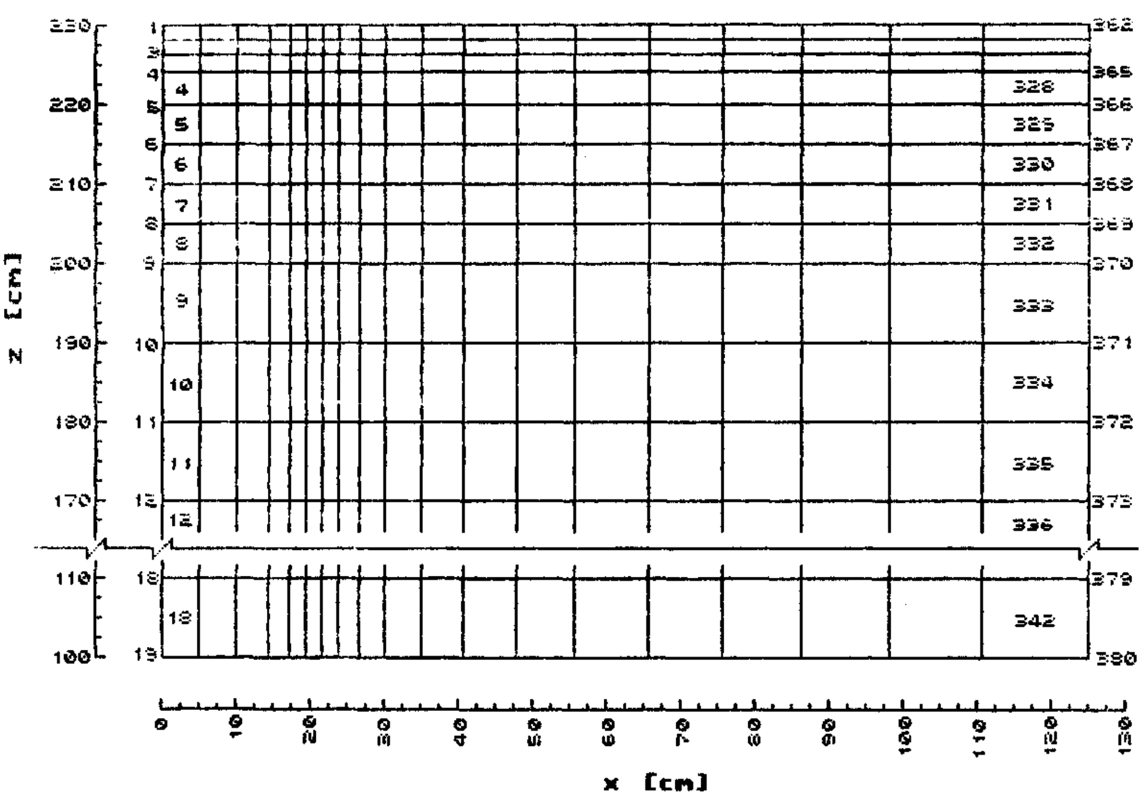


Figure 7.10. Flow system and finite element mesh for Example Problem #3.

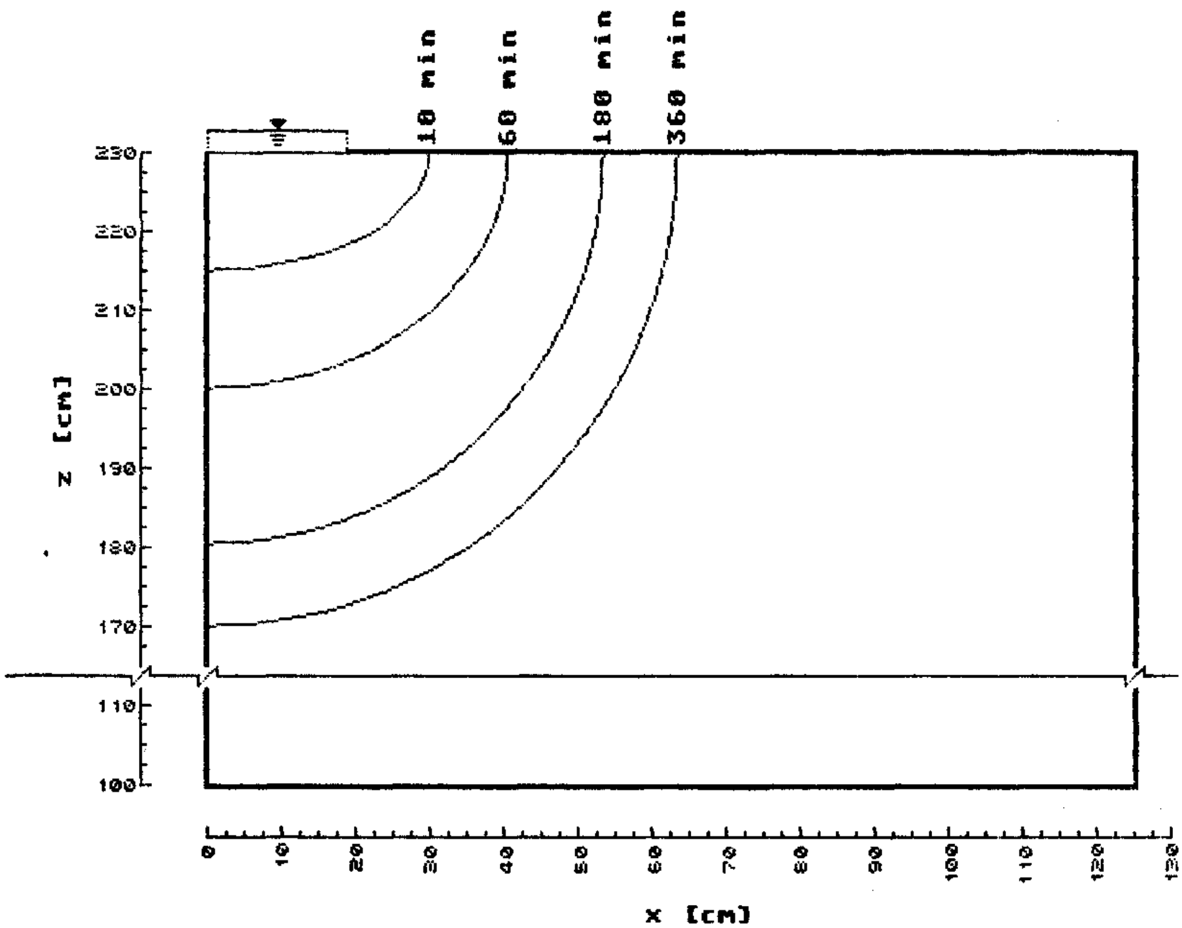


Figure 7.11. Advancement of moisture front during the infiltration experiment, Example Problem #3.

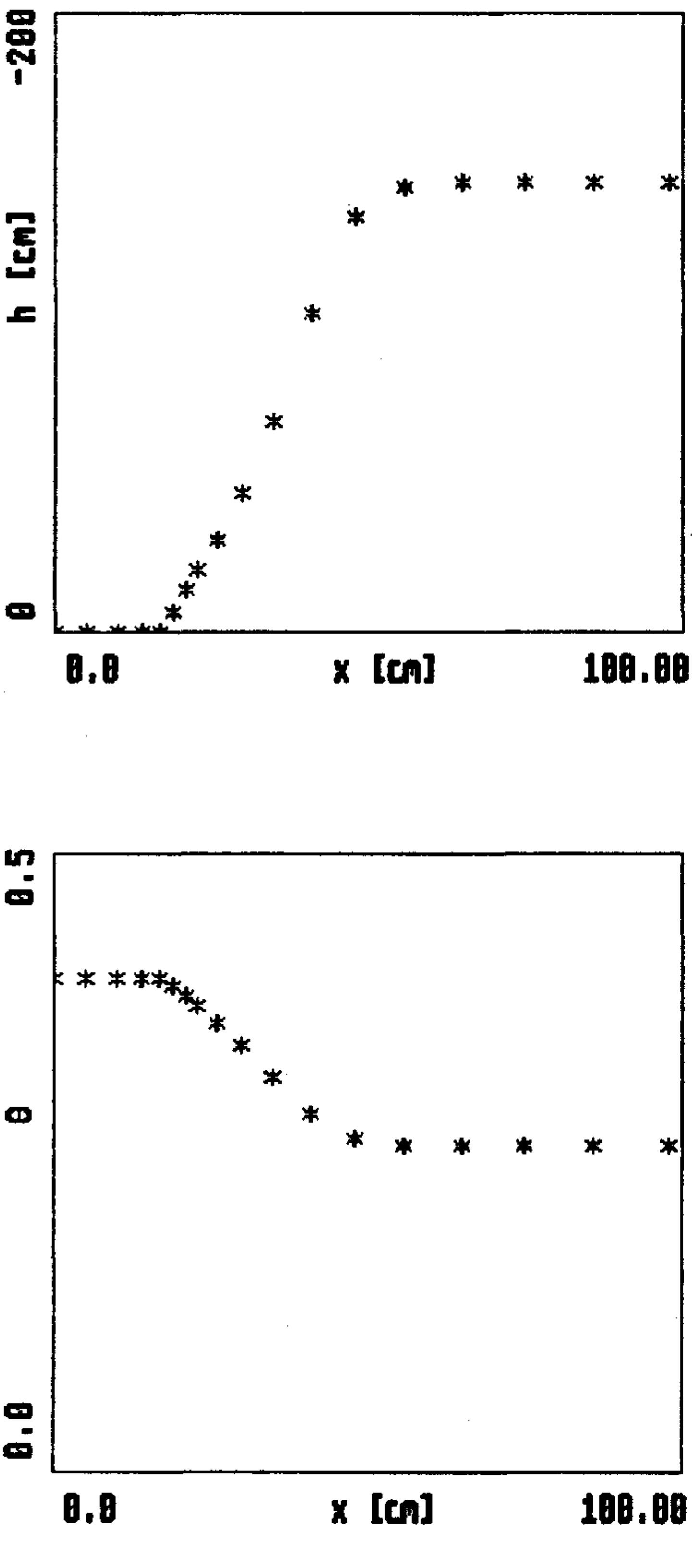


Figure 7.12. Pressure heads, h, and moisture contents, θ, along the surface boundary after 360 minutes from the start of infiltration, Example Problem #3.

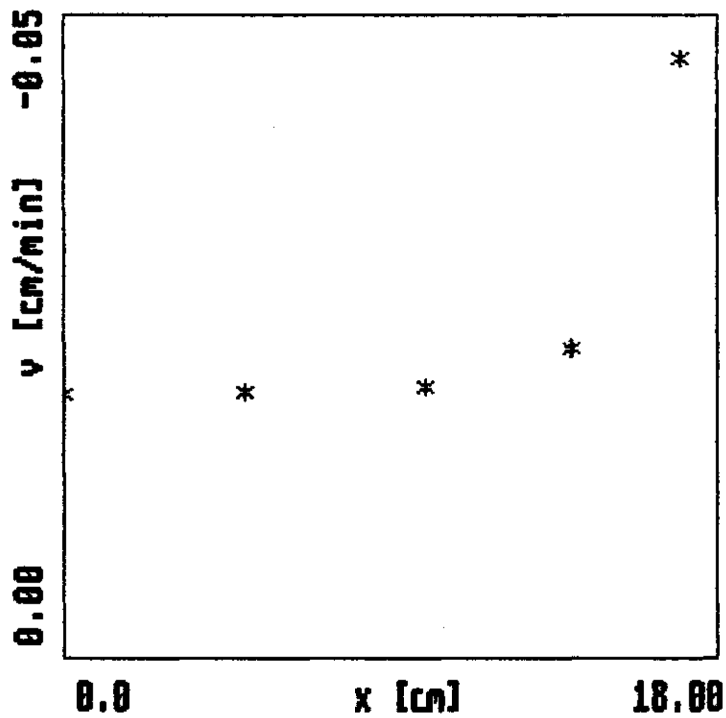
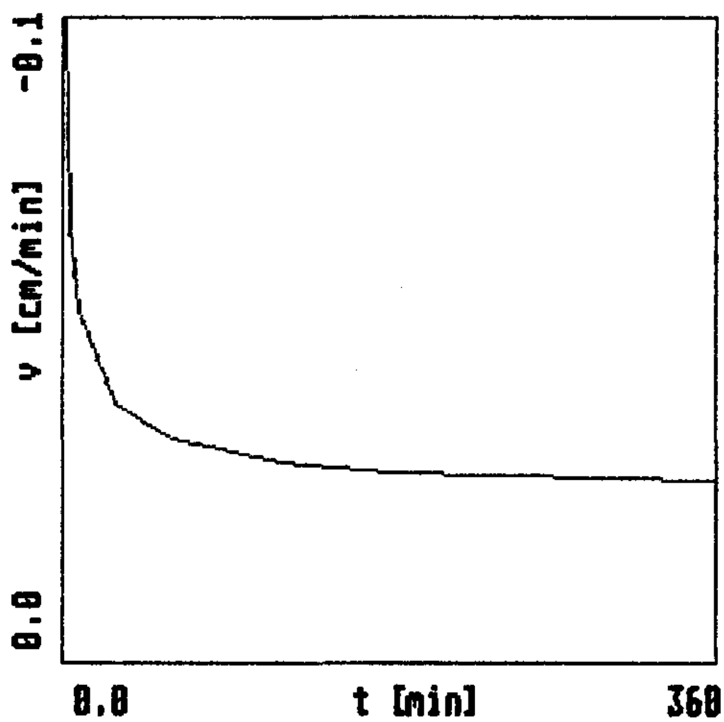


Figure 7.13. The course of boundary darcian velocity inside the infiltration ring (of radius equal to 18 cm) as function of distance from the symmetry axis, Example Problem #3.



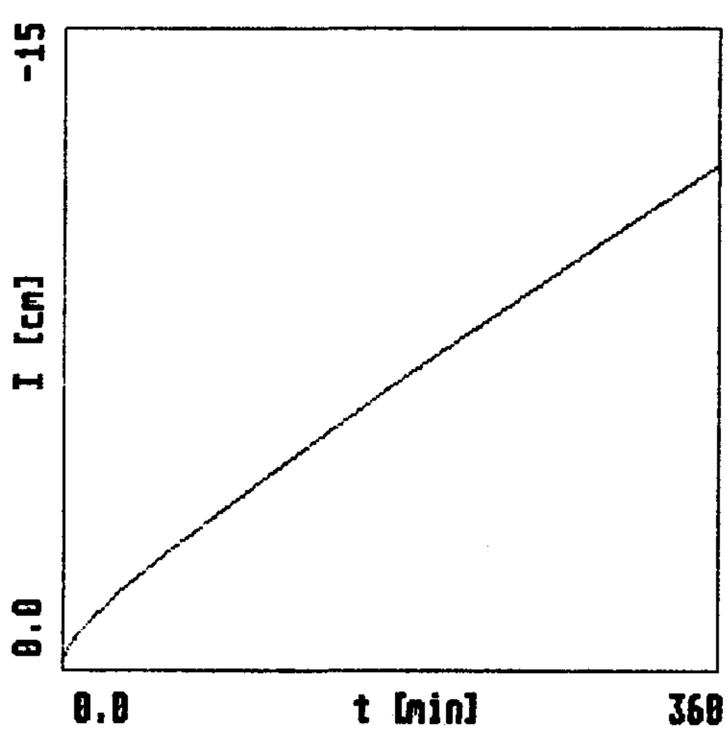


Figure 7.14. Mean infiltration rate, v, and cumulative infiltration rate, I, averaged over the infiltration ring area, Example Problem #3.

TABLE 7.19. Input Data for Example Problem #3, Input File 'SELECTOR.IN'.

```
*** BLOCK A: BASIC INFORMATION *****************************************
Heading
  ' Example problem #3 - Infiltration Test '
LUnit  TUnit        (indicated units are obligatory for all input data)
  'cm'  'min'
Kat (0:horizontal plane, 1:axisymmetric vertical flow, 2:vertical plane)
  1
MaxIt  Tol          (maximum number of iterations and p. head tolerance)
  20   .10
CheckF ShortF FluxF  AtmInF SeepF
  f      t     t       f     f
*** BLOCK B: MATERIAL INFORMATION **************************************
NMat   NLay   hTab1   hTabN  NPar
  2     2     .001    200.   9
thr    ths    tha     thm    Alfa    n       Ks     Kk     thk
.0001  .399   .0001   .399   .0174   1.3757  .0207  .0207  .399
.0001  .339   .0001   .339   .0139   1.6024  .0315  .0315  .339
*** BLOCK C: TIME INFORMATION ******************************************
dt     dtMin  dtMax   DMul   DMul2   MPL
.01    .001   30.     1.33   .33     10
TPrint(1),TPrint(2),...,TPrint(MPL)                  (print-time array)
1 5 10 30 60 120 180 240 300 360
*** END OF INPUT FILE 'SELECTOR.IN' ************************************
```

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.

```
*** BLOCK E: NODAL INFORMATION *****************************************
NumNP  NumEl  IJ     NumBP
 380    342   19      40
  N Kode    X      Z         P1        Q      M  Beta  Axz   Bxz   Dxz
  1  1    0.00  230.00     0.00  0.00E+00  1  0.00  1.00  1.00  1.00
  2  0    0.00  228.00  -145.50  0.00E+00  1  0.00  1.00  1.00  1.00
  3  0    0.00  226.00  -143.40  0.00E+00  1  0.00  1.00  1.00  1.00
  4  0    0.00  224.00  -141.00  0.00E+00  1  0.00  1.00  1.00  1.00
  5  0    0.00  220.00  -135.60  0.00E+00  1  0.00  1.00  1.00  1.00
  6  0    0.00  215.00  -127.70  0.00E+00  1  0.00  1.00  1.00  1.00
  7  0    0.00  210.00  -119.00  0.00E+00  1  0.00  1.00  1.00  1.00
  8  0    0.00  205.00  -109.90  0.00E+00  1  0.00  1.00  1.00  1.00
  9  0    0.00  200.00  -100.50  0.00E+00  1  0.00  1.00  1.00  1.00
 10  0    0.00  190.00   -82.80  0.00E+00  1  0.00  1.00  1.00  1.00
 11  0    0.00  180.00   -71.00  0.00E+00  2  0.00  1.00  1.00  1.00
 12  0    0.00  170.00   -60.30  0.00E+00  2  0.00  1.00  1.00  1.00
 13  0    0.00  160.00   -49.80  0.00E+00  2  0.00  1.00  1.00  1.00
 14  0    0.00  150.00   -39.60  0.00E+00  2  0.00  1.00  1.00  1.00
 15  0    0.00  140.00   -29.50  0.00E+00  2  0.00  1.00  1.00  1.00
 16  0    0.00  130.00   -19.40  0.00E+00  2  0.00  1.00  1.00  1.00
 17  0    0.00  120.00    -9.40  0.00E+00  2  0.00  1.00  1.00  1.00
 18  0    0.00  110.00     0.60  0.00E+00  2  0.00  1.00  1.00  1.00
 19  0    0.00  100.00    10.60  0.00E+00  2  0.00  1.00  1.00  1.00
 20  1    5.00  230.00     0.00  0.00E+00  1  0.00  1.00  1.00  1.00
```

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

21	0	5.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00		
22	0	5.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00		
23	0	5.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00		
24	0	5.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00		
25	0	5.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00		
26	0	5.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00		
27	0	5.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00		
28	0	5.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00		
29	0	5.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00		
30	0	5.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00		
31	0	5.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00		
32	0	5.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00		
33	0	5.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00		
34	0	5.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00		
35	0	5.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00		
36	0	5.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00		
37	0	5.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00		
38	0	5.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00		
39	1	10.00	230.00	0.00	0.00E+00	1	0.00	1.00	1.00	1.00		
40	0	10.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00		
41	0	10.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00		
42	0	10.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00		
43	0	10.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00		
44	0	10.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00		
45	0	10.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00		
46	0	10.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00		
47	0	10.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00		
48	0	10.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00		
49	0	10.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00		
50	0	10.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00		
51	0	10.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00		
52	0	10.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00		
53	0	10.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00		
54	0	10.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00		
55	0	10.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00		
56	0	10.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00		
57	0	10.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00		
58	1	14.00	230.00	0.00	0.00E+00	1	0.00	1.00	1.00	1.00		
59	0	14.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00		
60	0	14.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00		
61	0	14.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00		
62	0	14.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00		
63	0	14.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00		
64	0	14.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00		
65	0	14.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00		
66	0	14.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00		
67	0	14.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00		
68	0	14.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00		
69	0	14.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00		
70	0	14.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00		
71	0	14.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00		
72	0	14.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00		
73	0	14.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00		
74	0	14.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00		
75	0	14.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00		
76	0	14.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00		

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

77	1	17.00	230.00	0.00	0.00E+00	1	0.00	1.00	1.00	1.00
78	0	17.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
79	0	17.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
80	0	17.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
81	0	17.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
82	0	17.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
83	0	17.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
84	0	17.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
85	0	17.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
86	0	17.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
87	0	17.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
88	0	17.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
89	0	17.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
90	0	17.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
91	0	17.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
92	0	17.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
93	0	17.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
94	0	17.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
95	0	17.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
96	0	19.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
97	0	19.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
98	0	19.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
99	0	19.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
100	0	19.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
101	0	19.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
102	0	19.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
103	0	19.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
104	0	19.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
105	0	19.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
106	0	19.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
107	0	19.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
108	0	19.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
109	0	19.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
110	0	19.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
111	0	19.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
112	0	19.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
113	0	19.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
114	0	19.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
115	0	21.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
116	0	21.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
117	0	21.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
118	0	21.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
119	0	21.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
120	0	21.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
121	0	21.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
122	0	21.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
123	0	21.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
124	0	21.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
125	0	21.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
126	0	21.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
127	0	21.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
128	0	21.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
129	0	21.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
130	0	21.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
131	0	21.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
132	0	21.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

133	0	21.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
134	0	23.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
135	0	23.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
136	0	23.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
137	0	23.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
138	0	23.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
139	0	23.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
140	0	23.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
141	0	23.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
142	0	23.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
143	0	23.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
144	0	23.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
145	0	23.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
146	0	23.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
147	0	23.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
148	0	23.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
149	0	23.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
150	0	23.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
151	0	23.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
152	0	23.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
153	0	26.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
154	0	26.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
155	0	26.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
156	0	26.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
157	0	26.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
158	0	26.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
159	0	26.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
160	0	26.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
161	0	26.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
162	0	26.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
163	0	26.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
164	0	26.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
165	0	26.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
166	0	26.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
167	0	26.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
168	0	26.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
169	0	26.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
170	0	26.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
171	0	26.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
172	0	30.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
173	0	30.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
174	0	30.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
175	0	30.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
176	0	30.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
177	0	30.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
178	0	30.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
179	0	30.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
180	0	30.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
181	0	30.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
182	0	30.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
183	0	30.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
184	0	30.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
185	0	30.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
186	0	30.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
187	0	30.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
188	0	30.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

189	0	30.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
190	0	30.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
191	0	35.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
192	0	35.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
193	0	35.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
194	0	35.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
195	0	35.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
196	0	35.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
197	0	35.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
198	0	35.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
199	0	35.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
200	0	35.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
201	0	35.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
202	0	35.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
203	0	35.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
204	0	35.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
205	0	35.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
206	0	35.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
207	0	35.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
208	0	35.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
209	0	35.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
210	0	41.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
211	0	41.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
212	0	41.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
213	0	41.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
214	0	41.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
215	0	41.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
216	0	41.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
217	0	41.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
218	0	41.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
219	0	41.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
220	0	41.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
221	0	41.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
222	0	41.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
223	0	41.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
224	0	41.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
225	0	41.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
226	0	41.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
227	0	41.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
228	0	41.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
229	0	48.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
230	0	48.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
231	0	48.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
232	0	48.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
233	0	48.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
234	0	48.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
235	0	48.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
236	0	48.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
237	0	48.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
238	0	48.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
239	0	48.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
240	0	48.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
241	0	48.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
242	0	48.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
243	0	48.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
244	0	48.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

245	0	48.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
246	0	48.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
247	0	48.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
248	0	56.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
249	0	56.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
250	0	56.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
251	0	56.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
252	0	56.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
253	0	56.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
254	0	56.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
255	0	56.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
256	0	56.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
257	0	56.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
258	0	56.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
259	0	56.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
260	0	56.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
261	0	56.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
262	0	56.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
263	0	56.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
264	0	56.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
265	0	56.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
266	0	56.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
267	0	65.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
268	0	65.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
269	0	65.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
270	0	65.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
271	0	65.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
272	0	65.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
273	0	65.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
274	0	65.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
275	0	65.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
276	0	65.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
277	0	65.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
278	0	65.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
279	0	65.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
280	0	65.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
281	0	65.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
282	0	65.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
283	0	65.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
284	0	65.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
285	0	65.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
286	0	75.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
287	0	75.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
288	0	75.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
289	0	75.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
290	0	75.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
291	0	75.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
292	0	75.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
293	0	75.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
294	0	75.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
295	0	75.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
296	0	75.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
297	0	75.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
298	0	75.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
299	0	75.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
300	0	75.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

301	0	75.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
302	0	75.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
303	0	75.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
304	0	75.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
305	0	86.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
306	0	86.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
307	0	86.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
308	0	86.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
309	0	86.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
310	0	86.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
311	0	86.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
312	0	86.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
313	0	86.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
314	0	86.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
315	0	86.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
316	0	86.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
317	0	86.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
318	0	86.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
319	0	86.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
320	0	86.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
321	0	86.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
322	0	86.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
323	0	86.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
324	0	98.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
325	0	98.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
326	0	98.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
327	0	98.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
328	0	98.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
329	0	98.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
330	0	98.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
331	0	98.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
332	0	98.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
333	0	98.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
334	0	98.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
335	0	98.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
336	0	98.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
337	0	98.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
338	0	98.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
339	0	98.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
340	0	98.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
341	0	98.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
342	0	98.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
343	0	111.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
344	0	111.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
345	0	111.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
346	0	111.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
347	0	111.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
348	0	111.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
349	0	111.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
350	0	111.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
351	0	111.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
352	0	111.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
353	0	111.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
354	0	111.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
355	0	111.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
356	0	111.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

357	0	111.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
358	0	111.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
359	0	111.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
360	0	111.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
361	0	111.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00
362	0	125.00	230.00	-147.40	0.00E+00	1	0.00	1.00	1.00	1.00
363	0	125.00	228.00	-145.50	0.00E+00	1	0.00	1.00	1.00	1.00
364	0	125.00	226.00	-143.40	0.00E+00	1	0.00	1.00	1.00	1.00
365	0	125.00	224.00	-141.00	0.00E+00	1	0.00	1.00	1.00	1.00
366	0	125.00	220.00	-135.60	0.00E+00	1	0.00	1.00	1.00	1.00
367	0	125.00	215.00	-127.70	0.00E+00	1	0.00	1.00	1.00	1.00
368	0	125.00	210.00	-119.00	0.00E+00	1	0.00	1.00	1.00	1.00
369	0	125.00	205.00	-109.90	0.00E+00	1	0.00	1.00	1.00	1.00
370	0	125.00	200.00	-100.50	0.00E+00	1	0.00	1.00	1.00	1.00
371	0	125.00	190.00	-82.80	0.00E+00	1	0.00	1.00	1.00	1.00
372	0	125.00	180.00	-71.00	0.00E+00	2	0.00	1.00	1.00	1.00
373	0	125.00	170.00	-60.30	0.00E+00	2	0.00	1.00	1.00	1.00
374	0	125.00	160.00	-49.80	0.00E+00	2	0.00	1.00	1.00	1.00
375	0	125.00	150.00	-39.60	0.00E+00	2	0.00	1.00	1.00	1.00
376	0	125.00	140.00	-29.50	0.00E+00	2	0.00	1.00	1.00	1.00
377	0	125.00	130.00	-19.40	0.00E+00	2	0.00	1.00	1.00	1.00
378	0	125.00	120.00	-9.40	0.00E+00	2	0.00	1.00	1.00	1.00
379	0	125.00	110.00	0.60	0.00E+00	2	0.00	1.00	1.00	1.00
380	0	125.00	100.00	10.60	0.00E+00	2	0.00	1.00	1.00	1.00

*** BLOCK F: ELEMENT INFORMATION ***

E	i	j	k	l	Angle	ConA11	ConA33	LayNum
1	2	21	20	1	0.00	1.00	1.00	1
2	3	22	21	2	0.00	1.00	1.00	1
3	4	23	22	3	0.00	1.00	1.00	1
4	5	24	23	4	0.00	1.00	1.00	1
5	6	25	24	5	0.00	1.00	1.00	1
6	7	26	25	6	0.00	1.00	1.00	1
7	8	27	26	7	0.00	1.00	1.00	1
8	9	28	27	8	0.00	1.00	1.00	1
9	10	29	28	9	0.00	1.00	1.00	1
10	11	30	29	10	0.00	1.00	1.00	2
11	12	31	30	11	0.00	1.00	1.00	2
12	13	32	31	12	0.00	1.00	1.00	2
13	14	33	32	13	0.00	1.00	1.00	2
14	15	34	33	14	0.00	1.00	1.00	2
15	16	35	34	15	0.00	1.00	1.00	2
16	17	36	35	16	0.00	1.00	1.00	2
17	18	37	36	17	0.00	1.00	1.00	2
18	19	38	37	18	0.00	1.00	1.00	2
19	21	40	39	20	0.00	1.00	1.00	1
20	22	41	40	21	0.00	1.00	1.00	1
21	23	42	41	22	0.00	1.00	1.00	1
22	24	43	42	23	0.00	1.00	1.00	1
23	25	44	43	24	0.00	1.00	1.00	1
24	26	45	44	25	0.00	1.00	1.00	1
25	27	46	45	26	0.00	1.00	1.00	1
26	28	47	46	27	0.00	1.00	1.00	1
27	29	48	47	28	0.00	1.00	1.00	1
28	30	49	48	29	0.00	1.00	1.00	2
29	31	50	49	30	0.00	1.00	1.00	2
30	32	51	50	31	0.00	1.00	1.00	2

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'. (continued)

31	33	52	51	32	0.00	1.00	1.00	2
32	34	53	52	33	0.00	1.00	1.00	2
33	35	54	53	34	0.00	1.00	1.00	2
34	36	55	54	35	0.00	1.00	1.00	2
35	37	56	55	36	0.00	1.00	1.00	2
36	38	57	56	37	0.00	1.00	1.00	2
37	40	59	58	39	0.00	1.00	1.00	1
38	41	60	59	40	0.00	1.00	1.00	1
39	42	61	60	41	0.00	1.00	1.00	1
40	43	62	61	42	0.00	1.00	1.00	1
41	44	63	62	43	0.00	1.00	1.00	1
42	45	64	63	44	0.00	1.00	1.00	1
43	46	65	64	45	0.00	1.00	1.00	1
44	47	66	65	46	0.00	1.00	1.00	1
45	48	67	66	47	0.00	1.00	1.00	1
46	49	68	67	48	0.00	1.00	1.00	2
47	50	69	68	49	0.00	1.00	1.00	2
48	51	70	69	50	0.00	1.00	1.00	2
49	52	71	70	51	0.00	1.00	1.00	2
50	53	72	71	52	0.00	1.00	1.00	2
51	54	73	72	53	0.00	1.00	1.00	2
52	55	74	73	54	0.00	1.00	1.00	2
53	56	75	74	55	0.00	1.00	1.00	2
54	57	76	75	56	0.00	1.00	1.00	2
55	59	78	77	58	0.00	1.00	1.00	1
56	60	79	78	59	0.00	1.00	1.00	1
57	61	80	79	60	0.00	1.00	1.00	1
58	62	81	80	61	0.00	1.00	1.00	1
59	63	82	81	62	0.00	1.00	1.00	1
60	64	83	82	63	0.00	1.00	1.00	1
61	65	84	83	64	0.00	1.00	1.00	1
62	66	85	84	65	0.00	1.00	1.00	1
63	67	86	85	66	0.00	1.00	1.00	1
64	68	87	86	67	0.00	1.00	1.00	2
65	69	88	87	68	0.00	1.00	1.00	2
66	70	89	88	69	0.00	1.00	1.00	2
67	71	90	89	70	0.00	1.00	1.00	2
68	72	91	90	71	0.00	1.00	1.00	2
69	73	92	91	72	0.00	1.00	1.00	2
70	74	93	92	73	0.00	1.00	1.00	2
71	75	94	93	74	0.00	1.00	1.00	2
72	76	95	94	75	0.00	1.00	1.00	2
73	78	97	96	77	0.00	1.00	1.00	1
74	79	98	97	78	0.00	1.00	1.00	1
75	80	99	98	79	0.00	1.00	1.00	1
76	81	100	99	80	0.00	1.00	1.00	1
77	82	101	100	81	0.00	1.00	1.00	1
78	83	102	101	82	0.00	1.00	1.00	1
79	84	103	102	83	0.00	1.00	1.00	1
80	85	104	103	84	0.00	1.00	1.00	1
81	86	105	104	85	0.00	1.00	1.00	1
82	87	106	105	86	0.00	1.00	1.00	2
83	88	107	106	87	0.00	1.00	1.00	2
84	89	108	107	88	0.00	1.00	1.00	2
85	90	109	108	89	0.00	1.00	1.00	2
86	91	110	109	90	0.00	1.00	1.00	2

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

87	92	111	110	91	0.00	1.00	1.00	2
88	93	112	111	92	0.00	1.00	1.00	2
89	94	113	112	93	0.00	1.00	1.00	2
90	95	114	113	94	0.00	1.00	1.00	2
91	97	116	115	96	0.00	1.00	1.00	1
92	98	117	116	97	0.00	1.00	1.00	1
93	99	118	117	98	0.00	1.00	1.00	1
94	100	119	118	99	0.00	1.00	1.00	1
95	101	120	119	100	0.00	1.00	1.00	1
96	102	121	120	101	0.00	1.00	1.00	1
97	103	122	121	102	0.00	1.00	1.00	1
98	104	123	122	103	0.00	1.00	1.00	1
99	105	124	123	104	0.00	1.00	1.00	1
100	106	125	124	105	0.00	1.00	1.00	2
101	107	126	125	106	0.00	1.00	1.00	2
102	108	127	126	107	0.00	1.00	1.00	2
103	109	128	127	108	0.00	1.00	1.00	2
104	110	129	128	109	0.00	1.00	1.00	2
105	111	130	129	110	0.00	1.00	1.00	2
106	112	131	130	111	0.00	1.00	1.00	2
107	113	132	131	112	0.00	1.00	1.00	2
108	114	133	132	113	0.00	1.00	1.00	2
109	116	135	134	115	0.00	1.00	1.00	1
110	117	136	135	116	0.00	1.00	1.00	1
111	118	137	136	117	0.00	1.00	1.00	1
112	119	138	137	118	0.00	1.00	1.00	1
113	120	139	138	119	0.00	1.00	1.00	1
114	121	140	139	120	0.00	1.00	1.00	1
115	122	141	140	121	0.00	1.00	1.00	1
116	123	142	141	122	0.00	1.00	1.00	1
117	124	143	142	123	0.00	1.00	1.00	1
118	125	144	143	124	0.00	1.00	1.00	2
119	126	145	144	125	0.00	1.00	1.00	2
120	127	146	145	126	0.00	1.00	1.00	2
121	128	147	146	127	0.00	1.00	1.00	2
122	129	148	147	128	0.00	1.00	1.00	2
123	130	149	148	129	0.00	1.00	1.00	2
124	131	150	149	130	0.00	1.00	1.00	2
125	132	151	150	131	0.00	1.00	1.00	2
126	133	152	151	132	0.00	1.00	1.00	2
127	135	154	153	134	0.00	1.00	1.00	1
128	136	155	154	135	0.00	1.00	1.00	1
129	137	156	155	136	0.00	1.00	1.00	1
130	138	157	156	137	0.00	1.00	1.00	1
131	139	158	157	138	0.00	1.00	1.00	1
132	140	159	158	139	0.00	1.00	1.00	1
133	141	160	159	140	0.00	1.00	1.00	1
134	142	161	160	141	0.00	1.00	1.00	1
135	143	162	161	142	0.00	1.00	1.00	1
136	144	163	162	143	0.00	1.00	1.00	2
137	145	164	163	144	0.00	1.00	1.00	2
138	146	165	164	145	0.00	1.00	1.00	2
139	147	166	165	146	0.00	1.00	1.00	2
140	148	167	166	147	0.00	1.00	1.00	2
141	149	168	167	148	0.00	1.00	1.00	2
142	150	169	168	149	0.00	1.00	1.00	2

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

143	151	170	169	150	0.00	1.00	1.00	2
144	152	171	170	151	0.00	1.00	1.00	2
145	154	173	172	153	0.00	1.00	1.00	1
146	155	174	173	154	0.00	1.00	1.00	1
147	156	175	174	155	0.00	1.00	1.00	1
148	157	176	175	156	0.00	1.00	1.00	1
149	158	177	176	157	0.00	1.00	1.00	1
150	159	178	177	158	0.00	1.00	1.00	1
151	160	179	178	159	0.00	1.00	1.00	1
152	161	180	179	160	0.00	1.00	1.00	1
153	162	181	180	161	0.00	1.00	1.00	1
154	163	182	181	162	0.00	1.00	1.00	2
155	164	183	182	163	0.00	1.00	1.00	2
156	165	184	183	164	0.00	1.00	1.00	2
157	166	185	184	165	0.00	1.00	1.00	2
158	167	186	185	166	0.00	1.00	1.00	2
159	168	187	186	167	0.00	1.00	1.00	2
160	169	188	187	168	0.00	1.00	1.00	2
161	170	189	188	169	0.00	1.00	1.00	2
162	171	190	189	170	0.00	1.00	1.00	2
163	173	192	191	172	0.00	1.00	1.00	1
164	174	193	192	173	0.00	1.00	1.00	1
165	175	194	193	174	0.00	1.00	1.00	1
166	176	195	194	175	0.00	1.00	1.00	1
167	177	196	195	176	0.00	1.00	1.00	1
168	178	197	196	177	0.00	1.00	1.00	1
169	179	198	197	178	0.00	1.00	1.00	1
170	180	199	198	179	0.00	1.00	1.00	1
171	181	200	199	180	0.00	1.00	1.00	1
172	182	201	200	181	0.00	1.00	1.00	2
173	183	202	201	182	0.00	1.00	1.00	2
174	184	203	202	183	0.00	1.00	1.00	2
175	185	204	203	184	0.00	1.00	1.00	2
176	186	205	204	185	0.00	1.00	1.00	2
177	187	206	205	186	0.00	1.00	1.00	2
178	188	207	206	187	0.00	1.00	1.00	2
179	189	208	207	188	0.00	1.00	1.00	2
180	190	209	208	189	0.00	1.00	1.00	2
181	192	211	210	191	0.00	1.00	1.00	1
182	193	212	211	192	0.00	1.00	1.00	1
183	194	213	212	193	0.00	1.00	1.00	1
184	195	214	213	194	0.00	1.00	1.00	1
185	196	215	214	195	0.00	1.00	1.00	1
186	197	216	215	196	0.00	1.00	1.00	1
187	198	217	216	197	0.00	1.00	1.00	1
188	199	218	217	198	0.00	1.00	1.00	1
189	200	219	218	199	0.00	1.00	1.00	1
190	201	220	219	200	0.00	1.00	1.00	2
191	202	221	220	201	0.00	1.00	1.00	2
192	203	222	221	202	0.00	1.00	1.00	2
193	204	223	222	203	0.00	1.00	1.00	2
194	205	224	223	204	0.00	1.00	1.00	2
195	206	225	224	205	0.00	1.00	1.00	2
196	207	226	225	206	0.00	1.00	1.00	2
197	208	227	226	207	0.00	1.00	1.00	2
198	209	228	227	208	0.00	1.00	1.00	2

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'. (continued)

199	211	230	229	210	0.00	1.00	1.00	1
200	212	231	230	211	0.00	1.00	1.00	1
201	213	232	231	212	0.00	1.00	1.00	1
202	214	233	232	213	0.00	1.00	1.00	1
203	215	234	233	214	0.00	1.00	1.00	1
204	216	235	234	215	0.00	1.00	1.00	1
205	217	236	235	216	0.00	1.00	1.00	1
206	218	237	236	217	0.00	1.00	1.00	1
207	219	238	237	218	0.00	1.00	1.00	1
208	220	239	238	219	0.00	1.00	1.00	2
209	221	240	239	220	0.00	1.00	1.00	2
210	222	241	240	221	0.00	1.00	1.00	2
211	223	242	241	222	0.00	1.00	1.00	2
212	224	243	242	223	0.00	1.00	1.00	2
213	225	244	243	224	0.00	1.00	1.00	2
214	226	245	244	225	0.00	1.00	1.00	2
215	227	246	245	226	0.00	1.00	1.00	2
216	228	247	246	227	0.00	1.00	1.00	2
217	230	249	248	229	0.00	1.00	1.00	1
218	231	250	249	230	0.00	1.00	1.00	1
219	232	251	250	231	0.00	1.00	1.00	1
220	233	252	251	232	0.00	1.00	1.00	1
221	234	253	252	233	0.00	1.00	1.00	1
222	235	254	253	234	0.00	1.00	1.00	1
223	236	255	254	235	0.00	1.00	1.00	1
224	237	256	255	236	0.00	1.00	1.00	1
225	238	257	256	237	0.00	1.00	1.00	1
226	239	258	257	238	0.00	1.00	1.00	2
227	240	259	258	239	0.00	1.00	1.00	2
228	241	260	259	240	0.00	1.00	1.00	2
229	242	261	260	241	0.00	1.00	1.00	2
230	243	262	261	242	0.00	1.00	1.00	2
231	244	263	262	243	0.00	1.00	1.00	2
232	245	264	263	244	0.00	1.00	1.00	2
233	246	265	264	245	0.00	1.00	1.00	2
234	247	266	265	246	0.00	1.00	1.00	2
235	249	268	267	248	0.00	1.00	1.00	1
236	250	269	268	249	0.00	1.00	1.00	1
237	251	270	269	250	0.00	1.00	1.00	1
238	252	271	270	251	0.00	1.00	1.00	1
239	253	272	271	252	0.00	1.00	1.00	1
240	254	273	272	253	0.00	1.00	1.00	1
241	255	274	273	254	0.00	1.00	1.00	1
242	256	275	274	255	0.00	1.00	1.00	1
243	257	276	275	256	0.00	1.00	1.00	1
244	258	277	276	257	0.00	1.00	1.00	2
245	259	278	277	258	0.00	1.00	1.00	2
246	260	279	278	259	0.00	1.00	1.00	2
247	261	280	279	260	0.00	1.00	1.00	2
248	262	281	280	261	0.00	1.00	1.00	2
249	263	282	281	262	0.00	1.00	1.00	2
250	264	283	282	263	0.00	1.00	1.00	2
251	265	284	283	264	0.00	1.00	1.00	2
252	266	285	284	265	0.00	1.00	1.00	2
253	268	287	286	267	0.00	1.00	1.00	1
254	269	288	287	268	0.00	1.00	1.00	1

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

255	270	289	288	269	0.00	1.00	1.00	1
256	271	290	289	270	0.00	1.00	1.00	1
257	272	291	290	271	0.00	1.00	1.00	1
258	273	292	291	272	0.00	1.00	1.00	1
259	274	293	292	273	0.00	1.00	1.00	1
260	275	294	293	274	0.00	1.00	1.00	1
261	276	295	294	275	0.00	1.00	1.00	1
262	277	296	295	276	0.00	1.00	1.00	2
263	278	297	296	277	0.00	1.00	1.00	2
264	279	298	297	278	0.00	1.00	1.00	2
265	280	299	298	279	0.00	1.00	1.00	2
266	281	300	299	280	0.00	1.00	1.00	2
267	282	301	300	281	0.00	1.00	1.00	2
268	283	302	301	282	0.00	1.00	1.00	2
269	284	303	302	283	0.00	1.00	1.00	2
270	285	304	303	284	0.00	1.00	1.00	2
271	287	306	305	286	0.00	1.00	1.00	1
272	288	307	306	287	0.00	1.00	1.00	1
273	289	308	307	288	0.00	1.00	1.00	1
274	290	309	308	289	0.00	1.00	1.00	1
275	291	310	309	290	0.00	1.00	1.00	1
276	292	311	310	291	0.00	1.00	1.00	1
277	293	312	311	292	0.00	1.00	1.00	1
278	294	313	312	293	0.00	1.00	1.00	1
279	295	314	313	294	0.00	1.00	1.00	1
280	296	315	314	295	0.00	1.00	1.00	2
281	297	316	315	296	0.00	1.00	1.00	2
282	298	317	316	297	0.00	1.00	1.00	2
283	299	318	317	298	0.00	1.00	1.00	2
284	300	319	318	299	0.00	1.00	1.00	2
285	301	320	319	300	0.00	1.00	1.00	2
286	302	321	320	301	0.00	1.00	1.00	2
287	303	322	321	302	0.00	1.00	1.00	2
288	304	323	322	303	0.00	1.00	1.00	2
289	306	325	324	305	0.00	1.00	1.00	1
290	307	326	325	306	0.00	1.00	1.00	1
291	308	327	326	307	0.00	1.00	1.00	1
292	309	328	327	308	0.00	1.00	1.00	1
293	310	329	328	309	0.00	1.00	1.00	1
294	311	330	329	310	0.00	1.00	1.00	1
295	312	331	330	311	0.00	1.00	1.00	1
296	313	332	331	312	0.00	1.00	1.00	1
297	314	333	332	313	0.00	1.00	1.00	1
298	315	334	333	314	0.00	1.00	1.00	2
299	316	335	334	315	0.00	1.00	1.00	2
300	317	336	335	316	0.00	1.00	1.00	2
301	318	337	336	317	0.00	1.00	1.00	2
302	319	338	337	318	0.00	1.00	1.00	2
303	320	339	338	319	0.00	1.00	1.00	2
304	321	340	339	320	0.00	1.00	1.00	2
305	322	341	340	321	0.00	1.00	1.00	2
306	323	342	341	322	0.00	1.00	1.00	2
307	325	344	343	324	0.00	1.00	1.00	1
308	326	345	344	325	0.00	1.00	1.00	1
309	327	346	345	326	0.00	1.00	1.00	1
310	328	347	346	327	0.00	1.00	1.00	1

TABLE 7.20. Input Data for Example Problem #3, Input File 'GRID.IN'.
(continued)

```
311 329 348 347 328   0.00   1.00   1.00   1
312 330 349 348 329   0.00   1.00   1.00   1
313 331 350 349 330   0.00   1.00   1.00   1
314 332 351 350 331   0.00   1.00   1.00   1
315 333 352 351 332   0.00   1.00   1.00   1
316 334 353 352 333   0.00   1.00   1.00   2
317 335 354 353 334   0.00   1.00   1.00   2
318 336 355 354 335   0.00   1.00   1.00   2
319 337 356 355 336   0.00   1.00   1.00   2
320 338 357 356 337   0.00   1.00   1.00   2
321 339 358 357 338   0.00   1.00   1.00   2
322 340 359 358 339   0.00   1.00   1.00   2
323 341 360 359 340   0.00   1.00   1.00   2
324 342 361 360 341   0.00   1.00   1.00   2
325 344 363 362 343   0.00   1.00   1.00   1
326 345 364 363 344   0.00   1.00   1.00   1
327 346 365 364 345   0.00   1.00   1.00   1
328 347 366 365 346   0.00   1.00   1.00   1
329 348 367 366 347   0.00   1.00   1.00   1
330 349 368 367 348   0.00   1.00   1.00   1
331 350 369 368 349   0.00   1.00   1.00   1
332 351 370 369 350   0.00   1.00   1.00   1
333 352 371 370 351   0.00   1.00   1.00   1
334 353 372 371 352   0.00   1.00   1.00   2
335 354 373 372 353   0.00   1.00   1.00   2
336 355 374 373 354   0.00   1.00   1.00   2
337 356 375 374 355   0.00   1.00   1.00   2
338 357 376 375 356   0.00   1.00   1.00   2
339 358 377 376 357   0.00   1.00   1.00   2
340 359 378 377 358   0.00   1.00   1.00   2
341 360 379 378 359   0.00   1.00   1.00   2
342 361 380 379 360   0.00   1.00   1.00   2
*** BLOCK G: BOUNDARY GEOMETRIC INFORMATION ****************************
KXB(1), KXB(2),...,KXB(NumBP)                  (boundary node number array)
         1          20          39          58          77          96         115
       134         153         172         191         210         229         248
       267         286         305         324         343         362          19
        38          57          76          95         114         133         152
       171         190         209         228         247         266         285
       304         323         342         361         380
Width(1), Width(2),...,Width(NumBP)                            (width array)
     26.18      157.08      273.32      300.55      261.80      238.76      263.89
    366.52      579.10      857.66     1221.04     1688.09     2277.66     3008.61
   3899.77     4970.01     6238.17     7723.10     9443.65     5292.55       26.18
    157.08      273.32      300.55      261.80      238.76      263.89      366.52
    579.10      857.66     1221.04     1688.09     2277.66     3008.61     3899.77
   4970.01     6238.17     7723.10     9443.65     5292.55
rLen                                                          (surface area)
  49087.50
*** END OF INPUT FILE 'GRID.IN' ****************************************
```

TABLE 7.21. Output Data for Example Problem #3, Output File 'V_MEAN.OUT'.

```
Example problem #3 - Infiltration Test

Program SWM II
Date:  3. 5.1988   Time: 11:56
Time independent boundary conditions
Axisymmetric flow, V = L*L*L
Units: L = cm   , T = min
```

Time [T]	rAtm [L/T]	rRoot [L/T]	vAtm [L/T]	vRoot [L/T]	vKode3 [L/T]	vKode1 [L/T]	vSeep [L/T]	t-level
1.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.129E+00	0.000E+00	14
5.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.675E-01	0.000E+00	30
10.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.543E-01	0.000E+00	38
30.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.399E-01	0.000E+00	52
60.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.346E-01	0.000E+00	62
120.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.310E-01	0.000E+00	72
180.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.296E-01	0.000E+00	78
240.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.289E-01	0.000E+00	82
300.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.285E-01	0.000E+00	85
360.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.283E-01	0.000E+00	90

TABLE 7.22. Output Data for Example Problem #3, Output File 'CUM_Q.OUT'.

Time [T]	CumQAP [L]	CumQRP [L]	CumQA [L]	CumQR [L]	CumQ3 [L]	CumQ1 [L]	CumQS [L]	t-level
1.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.248E+00	0.000E+00	14
5.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.588E+00	0.000E+00	30
10.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.883E+00	0.000E+00	38
30.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.177E+01	0.000E+00	52
60.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.287E+01	0.000E+00	62
120.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.480E+01	0.000E+00	72
180.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.661E+01	0.000E+00	78
240.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.836E+01	0.000E+00	82
300.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.101E+02	0.000E+00	85
360.0000	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	-0.118E+02	0.000E+00	90

TABLE 7.23. Output at the End of Simulation for Example Problem #3, Output File 'BOUNDARY.OUT'.

Time: 360.0000

i	n	x	z	Code	Q [V/T]	v [L/T]	h	th
1	1	0.0	230.0	1	0.532E+00	-0.203E-01	0.0	0.399
2	20	5.0	230.0	1	0.321E+01	-0.205E-01	0.0	0.399
3	39	10.0	230.0	1	0.570E+01	-0.209E-01	0.0	0.399
4	58	14.0	230.0	1	0.717E+01	-0.239E-01	0.0	0.399
5	77	17.0	230.0	1	0.122E+02	-0.466E-01	0.0	0.399
6	96	19.0	230.0	0	0.000E+00	0.000E+00	-6.3	0.394
7	115	21.0	230.0	0	0.000E+00	0.000E+00	-13.1	0.386
8	134	23.0	230.0	0	0.000E+00	0.000E+00	-19.6	0.377
9	153	26.0	230.0	0	0.000E+00	0.000E+00	-29.5	0.364
10	172	30.0	230.0	0	0.000E+00	0.000E+00	-44.2	0.345
11	191	35.0	230.0	0	0.000E+00	0.000E+00	-67.4	0.320
12	210	41.0	230.0	0	0.000E+00	0.000E+00	-102.8	0.290
13	229	48.0	230.0	0	0.000E+00	0.000E+00	-134.3	0.269
14	248	56.0	230.0	0	0.000E+00	0.000E+00	-143.7	0.264
15	267	65.0	230.0	0	0.000E+00	0.000E+00	-145.0	0.263
16	286	75.0	230.0	0	0.000E+00	0.000E+00	-145.1	0.263
17	305	86.0	230.0	0	0.000E+00	0.000E+00	-145.1	0.263
18	324	98.0	230.0	0	0.000E+00	0.000E+00	-145.1	0.263
19	343	111.0	230.0	0	0.000E+00	0.000E+00	-145.1	0.263
20	362	125.0	230.0	0	0.000E+00	0.000E+00	-145.2	0.263
21	19	0.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
22	38	5.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
23	57	10.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
24	76	14.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
25	95	17.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
26	114	19.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
27	133	21.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
28	152	23.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
29	171	26.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
30	190	30.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
31	209	35.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
32	228	41.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
33	247	48.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
34	266	56.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
35	285	65.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
36	304	75.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
37	323	86.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
38	342	98.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
39	361	111.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339
40	380	125.0	100.0	0	0.000E+00	0.000E+00	9.9	0.339

TABLE 7.24. Output Data for Example Problem #3, Output File 'BALANCE.OUT'.

```
Time [T]          Total     Sub-region number ...

     1.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.190E+07  0.551E+06  0.135E+07
InFlow [V/T]    0.131E+03  0.140E+03 -0.827E+01
hMean  [L]                    -117.4      -34.9

     5.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.552E+06  0.135E+07
InFlow [V/T]    0.683E+02  0.764E+02 -0.815E+01
hMean  [L]                    -117.2      -34.9

    10.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.552E+06  0.135E+07
InFlow [V/T]    0.562E+02  0.642E+02 -0.809E+01
hMean  [L]                    -117.0      -34.9

    30.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.553E+06  0.135E+07
InFlow [V/T]    0.406E+02  0.485E+02 -0.796E+01
hMean  [L]                    -116.4      -35.0

    60.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.554E+06  0.135E+07
InFlow [V/T]    0.349E+02  0.427E+02 -0.780E+01
hMean  [L]                    -115.5      -35.0

   120.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.557E+06  0.135E+07
InFlow [V/T]    0.313E+02  0.388E+02 -0.746E+01
hMean  [L]                    -114.1      -35.1

   180.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.559E+06  0.135E+07
InFlow [V/T]    0.300E+02  0.369E+02 -0.686E+01
hMean  [L]                    -112.9      -35.2

   240.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.561E+06  0.135E+07
InFlow [V/T]    0.293E+02  0.351E+02 -0.581E+01
hMean  [L]                    -111.7      -35.3

   300.0000                      1          2
Area   [V]      0.638E+07  0.196E+07  0.442E+07
Volume [V]      0.191E+07  0.563E+06  0.135E+07
InFlow [V/T]    0.290E+02  0.334E+02 -0.435E+01
hMean  [L]                    -110.5      -35.4
```

TABLE 7.24. Output Data for Example Problem #3, Output File 'BALANCE.OUT'.
(continued)

360.0000		1	2
Area [V]	0.638E+07	0.196E+07	0.442E+07
Volume [V]	0.192E+07	0.565E+06	0.135E+07
InFlow [V/T]	0.286E+02	0.313E+02	-0.267E+01
hMean [L]		-108.9	-35.5

TABLE 7.25. Output at the Beginning of Simulation for Example Problem #3, Output File 'H.OUT'.

Time *** 1.0000 ***

n	x(n)	z(n)	h(n)	h(n+1) ...								
1	0.0	230.0	0.0	-17.0	-112.3	-140.6	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8
11	0.0	180.0	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6	0.0
21	5.0	228.0	-15.6	-107.2	-140.5	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0
31	5.0	170.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6	0.0	-16.2
41	10.0	226.0	-110.3	-140.6	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3
51	10.0	160.0	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6	0.0	-17.0	-115.1
61	14.0	224.0	-140.7	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8
71	14.0	150.0	-39.6	-29.5	-19.4	-9.4	0.6	10.6	0.0	-25.1	-127.3	-140.8
81	17.0	220.0	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6
91	17.0	140.0	-29.5	-19.4	-9.4	0.6	10.6	-35.5	-115.9	-142.7	-141.0	-135.6
101	19.0	215.0	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5
111	19.0	130.0	-19.4	-9.4	0.6	10.6	-136.8	-144.9	-143.4	-141.0	-135.6	-127.7
121	21.0	210.0	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4
131	21.0	120.0	-9.4	0.6	10.6	-147.3	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0
141	23.0	205.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4
151	23.0	110.0	0.6	10.6	-147.4	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0	-109.9
161	26.0	200.0	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6
171	26.0	100.0	10.6	-147.4	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0	-109.9	-100.5
181	30.0	190.0	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6
191	35.0	230.0	-147.4	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8
201	35.0	180.0	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6	-147.4
211	41.0	228.0	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0
221	41.0	170.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6	-147.4	-145.5
231	48.0	226.0	-143.4	-141.0	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3
241	48.0	160.0	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6	-147.4	-145.5	-143.4
251	56.0	224.0	-141.0	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8
261	56.0	150.0	-39.6	-29.5	-19.4	-9.4	0.6	10.6	-147.4	-145.5	-143.4	-141.0
271	65.0	220.0	-135.6	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6
281	65.0	140.0	-29.5	-19.4	-9.4	0.6	10.6	-147.4	-145.5	-143.4	-141.0	-135.6
291	75.0	215.0	-127.7	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5
301	75.0	130.0	-19.4	-9.4	0.6	10.6	-147.4	-145.5	-143.4	-141.0	-135.6	-127.7
311	86.0	210.0	-119.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4
321	86.0	120.0	-9.4	0.6	10.6	-147.4	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0
331	98.0	205.0	-109.9	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4
341	98.0	110.0	0.6	10.6	-147.4	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0	-109.9
351	111.0	200.0	-100.5	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6
361	111.0	100.0	10.6	-147.4	-145.5	-143.4	-141.0	-135.6	-127.7	-119.0	-109.9	-100.5
371	125.0	190.0	-82.8	-71.0	-60.3	-49.8	-39.6	-29.5	-19.4	-9.4	0.6	10.6

TABLE 7.26. Output at the End of Simulation for Example Problem #3, Output File 'H.OUT'.

Time ### 360.0000 ###

n	x(n)	z(n)	h(n)	h(n+1) ...									
1	0.0	230.0	0.0	0.0	-0.1	-0.6	-2.7	-7.0	-13.0	-21.2	-32.3	-61.2
11	0.0	180.0	-66.3	-59.2	-49.9	-40.1	-30.1	-20.1	-10.1	-0.1	9.9	0.0
21	5.0	228.0	0.0	-0.2	-0.9	-3.3	-7.8	-14.1	-22.4	-33.6	-62.5	-66.4
31	5.0	170.0	-59.2	-49.9	-40.1	-30.1	-20.1	-10.1	-0.1	9.9	0.0	-0.2
41	10.0	226.0	-0.9	-1.9	-5.0	-10.0	-16.7	-25.5	-37.1	-65.1	-66.9	-59.4
51	10.0	160.0	-50.0	-40.1	-30.1	-20.1	-10.1	-0.1	9.9	0.0	-0.7	-2.2
61	14.0	224.0	-3.9	-7.6	-13.2	-20.3	-29.6	-41.8	-67.8	-67.6	-59.5	-50.0
71	14.0	150.0	-40.1	-30.1	-20.1	-10.1	-0.1	9.9	0.0	-2.3	-4.7	-6.8
81	17.0	220.0	-10.9	-16.8	-24.2	-33.9	-46.4	-70.1	-68.1	-59.7	-50.1	-40.1
91	17.0	140.0	-30.1	-20.1	-10.1	-0.1	9.9	-6.3	-6.8	-8.3	-10.0	-13.9
101	19.0	215.0	-19.8	-27.4	-37.4	-50.2	-71.6	-68.4	-59.8	-50.1	-40.1	-30.1
111	19.0	130.0	-20.1	-10.1	-0.1	9.9	-13.1	-12.4	-13.0	-14.2	-17.6	-23.4
121	21.0	210.0	-31.2	-41.5	-54.3	-73.1	-68.8	-59.9	-50.1	-40.1	-30.1	-20.1
131	21.0	120.0	-10.1	-0.1	9.9	-19.6	-18.4	-18.4	-19.1	-22.0	-27.7	-35.7
141	23.0	205.0	-46.2	-58.9	-74.6	-69.2	-60.0	-50.2	-40.1	-30.1	-20.1	-10.1
151	23.0	110.0	-0.1	9.9	-29.5	-28.0	-27.5	-27.7	-30.1	-35.6	-43.7	-54.5
161	26.0	200.0	-66.6	-76.6	-69.7	-60.2	-50.2	-40.2	-30.1	-20.1	-10.1	-0.1
171	26.0	100.0	9.9	-44.2	-42.6	-41.8	-41.8	-43.6	-49.0	-57.3	-67.5	-77.0
181	30.0	190.0	-78.9	-70.2	-60.3	-50.3	-40.2	-30.1	-20.1	-10.1	-0.1	9.9
191	35.0	230.0	-67.4	-65.8	-64.8	-64.5	-66.1	-71.1	-78.2	-84.7	-87.9	-80.9
201	35.0	180.0	-70.7	-60.5	-50.3	-40.2	-30.1	-20.1	-10.1	-0.1	9.9	-102.8
211	41.0	228.0	-101.2	-100.0	-99.2	-99.2	-100.5	-101.1	-99.4	-95.2	-82.2	-71.1
221	41.0	170.0	-60.7	-50.4	-40.2	-30.1	-20.1	-10.1	-0.1	9.9	-134.3	-132.4
231	48.0	226.0	-130.4	-128.4	-124.5	-119.2	-113.0	-105.8	-98.2	-82.8	-71.4	-60.8
241	48.0	160.0	-50.4	-40.2	-30.1	-20.1	-10.1	-0.1	9.9	-143.7	-141.6	-139.3
251	56.0	224.0	-136.9	-131.5	-124.0	-115.8	-107.4	-98.9	-93.0	-71.5	-60.8	-50.5
261	56.0	150.0	-40.3	-30.1	-20.1	-10.1	-0.1	9.9	-145.0	-142.9	-140.6	-138.0
271	65.0	220.0	-132.4	-124.6	-116.2	-107.6	-99.0	-83.0	-71.5	-60.9	-50.5	-40.3
281	65.0	140.0	-30.2	-20.1	-10.1	-0.1	9.9	-145.1	-143.0	-140.7	-138.1	-132.5
291	75.0	215.0	-124.7	-116.3	-107.6	-99.1	-83.1	-71.5	-60.9	-50.5	-40.3	-30.2
301	75.0	130.0	-20.1	-10.1	-0.1	9.9	-145.1	-143.0	-140.7	-138.1	-132.5	-124.7
311	86.0	210.0	-116.3	-107.6	-99.1	-83.1	-71.5	-60.9	-50.5	-40.3	-30.2	-20.1
321	86.0	120.0	-10.1	-0.1	9.9	-145.1	-143.0	-140.7	-138.1	-132.5	-124.7	-116.3
331	98.0	205.0	-107.6	-99.1	-83.1	-71.6	-60.9	-50.5	-40.3	-30.2	-20.1	-10.1
341	98.0	110.0	-0.1	9.9	-145.1	-143.0	-140.7	-138.1	-132.5	-124.7	-116.3	-107.6
351	111.0	200.0	-99.0	-83.0	-71.6	-60.9	-50.5	-40.3	-30.2	-20.1	-10.1	-0.1
361	111.0	100.0	9.9	-145.2	-143.1	-140.7	-138.2	-132.5	-124.7	-116.2	-107.5	-98.9
371	125.0	190.0	-82.9	-71.6	-60.9	-50.5	-40.3	-30.2	-20.1	-10.1	-0.1	9.9

TABLE 7.27. Output at the End of Simulation for Example Problem #3, Output File 'TH.OUT'.

Time *** 360.0000 ***

n	x(n)	z(n)	th(n)	th(n+1) ...									
1	0.0	230.0	0.399	0.399	0.399	0.399	0.397	0.393	0.386	0.375	0.360	0.326	
11	0.0	180.0	0.268	0.276	0.287	0.299	0.312	0.324	0.334	0.339	0.339	0.399	
21	5.0	228.0	0.399	0.399	0.399	0.397	0.392	0.385	0.373	0.359	0.325	0.267	
31	5.0	170.0	0.276	0.287	0.299	0.312	0.324	0.334	0.339	0.339	0.399	0.399	
41	10.0	226.0	0.399	0.398	0.395	0.390	0.381	0.369	0.354	0.322	0.267	0.276	
51	10.0	160.0	0.287	0.299	0.312	0.324	0.334	0.339	0.339	0.399	0.399	0.398	
61	14.0	224.0	0.396	0.392	0.386	0.376	0.364	0.348	0.320	0.266	0.275	0.287	
71	14.0	150.0	0.299	0.312	0.324	0.334	0.339	0.339	0.399	0.398	0.396	0.393	
81	17.0	220.0	0.389	0.381	0.371	0.358	0.343	0.317	0.266	0.275	0.287	0.299	
91	17.0	140.0	0.312	0.324	0.334	0.339	0.339	0.394	0.393	0.392	0.390	0.385	
101	19.0	215.0	0.377	0.367	0.354	0.338	0.316	0.265	0.275	0.287	0.299	0.312	
111	19.0	130.0	0.324	0.334	0.339	0.339	0.386	0.387	0.386	0.384	0.380	0.372	
121	21.0	210.0	0.362	0.349	0.334	0.314	0.265	0.275	0.287	0.299	0.312	0.324	
131	21.0	120.0	0.334	0.339	0.339	0.377	0.379	0.379	0.378	0.374	0.366	0.356	
141	23.0	205.0	0.343	0.329	0.313	0.264	0.275	0.287	0.299	0.312	0.324	0.334	
151	23.0	110.0	0.339	0.339	0.364	0.366	0.367	0.366	0.363	0.356	0.346	0.333	
161	26.0	200.0	0.321	0.311	0.264	0.275	0.287	0.299	0.312	0.324	0.334	0.339	
171	76.0	100.0	0.339	0.345	0.347	0.348	0.348	0.346	0.340	0.330	0.320	0.311	
181	30.0	190.0	0.309	0.263	0.274	0.287	0.299	0.312	0.324	0.334	0.339	0.339	
191	35.0	230.0	0.320	0.322	0.323	0.323	0.321	0.316	0.310	0.304	0.301	0.307	
201	35.0	180.0	0.263	0.274	0.287	0.299	0.312	0.324	0.334	0.339	0.339	0.290	
211	41.0	228.0	0.291	0.292	0.292	0.293	0.292	0.291	0.292	0.296	0.306	0.262	
221	41.0	170.0	0.274	0.286	0.299	0.312	0.324	0.334	0.339	0.339	0.269	0.271	
231	48.0	226.0	0.272	0.273	0.275	0.279	0.283	0.288	0.293	0.306	0.262	0.274	
241	48.0	160.0	0.286	0.299	0.312	0.324	0.334	0.339	0.339	0.264	0.265	0.267	
251	56.0	224.0	0.268	0.271	0.276	0.281	0.287	0.293	0.306	0.262	0.274	0.286	
261	56.0	150.0	0.299	0.312	0.324	0.334	0.339	0.339	0.263	0.265	0.266	0.267	
271	65.0	220.0	0.270	0.275	0.280	0.286	0.293	0.305	0.262	0.274	0.286	0.299	
281	65.0	140.0	0.312	0.324	0.334	0.339	0.339	0.263	0.265	0.266	0.267	0.270	
291	75.0	215.0	0.275	0.280	0.286	0.293	0.305	0.262	0.274	0.286	0.299	0.312	
301	75.0	130.0	0.324	0.334	0.339	0.339	0.263	0.265	0.266	0.267	0.270	0.275	
311	86.0	210.0	0.280	0.286	0.293	0.305	0.262	0.274	0.286	0.299	0.312	0.324	
321	86.0	120.0	0.334	0.339	0.339	0.263	0.265	0.266	0.267	0.270	0.275	0.280	
331	98.0	205.0	0.286	0.293	0.305	0.262	0.274	0.286	0.299	0.312	0.324	0.334	
341	98.0	110.0	0.339	0.339	0.263	0.265	0.266	0.267	0.270	0.275	0.280	0.286	
351	111.0	200.0	0.293	0.306	0.262	0.274	0.286	0.299	0.312	0.324	0.334	0.339	
361	111.0	100.0	0.339	0.263	0.265	0.266	0.267	0.270	0.275	0.281	0.286	0.293	
371	125.0	190.0	0.306	0.262	0.274	0.286	0.299	0.312	0.324	0.334	0.339	0.339	

TABLE 7.28. Output at the End of Simulation for Example Problem #3, Output File 'VX.OUT'.

Time *** 360.0000 ***

e	x(e)	z(e)	vx(e)	vx(e+1) ...							
1	2.5	229.0	-0.25E-04	0.18E-03	0.60E-03	0.10E-02	0.11E-02	0.95E-03	0.71E-03	0.49E-03	0.25E-03	0.13E-03
11	2.5	175.0	0.32E-04	0.15E-04	0.71E-05	0.41E-05	0.29E-05	0.23E-05	0.20E-05	0.14E-05	0.34E-03	0.12E-02
21	7.5	225.0	0.21E-02	0.26E-02	0.26E-02	0.21E-02	0.16E-02	0.11E-02	0.54E-03	0.30E-03	0.13E-03	0.50E-04
31	7.5	155.0	0.21E-04	0.97E-05	0.53E-05	0.35E-05	0.33E-05	0.30E-05	0.12E-02	0.30E-02	0.42E-02	0.46E-02
41	12.0	217.5	0.40E-02	0.31E-02	0.23E-02	0.16E-02	0.72E-03	0.39E-03	0.19E-03	0.72E-04	0.30E-04	0.14E-04
51	12.0	135.0	0.74E-05	0.47E-05	0.45E-05	0.45E-05	0.39E-02	0.70E-02	0.74E-02	0.65E-02	0.50E-02	0.37E-02
61	15.5	207.5	0.26E-02	0.18E-02	0.78E-03	0.42E-03	0.21E-03	0.82E-04	0.35E-04	0.16E-04	0.87E-05	0.55E-05
71	15.5	115.0	0.53E-05	0.55E-05	0.29E-01	0.15E-01	0.11E-01	0.77E-02	0.54E-02	0.38E-02	0.27E-02	0.18E-02
81	18.0	195.0	0.78E-03	0.41E-03	0.22E-03	0.86E-04	0.37E-04	0.17E-04	0.93E-05	0.59E-05	0.58E-05	0.62E-05
91	20.0	229.0	0.16E-01	0.13E-01	0.10E-01	0.76E-02	0.54E-02	0.38E-02	0.27E-02	0.18E-02	0.77E-03	0.40E-03
101	20.0	175.0	0.22E-03	0.87E-04	0.38E-04	0.18E-04	0.98E-05	0.63E-05	0.62E-05	0.67E-05	0.11E-01	0.10E-01
111	22.0	225.0	0.88E-02	0.70E-02	0.52E-02	0.37E-02	0.26E-02	0.17E-02	0.73E-03	0.39E-03	0.21E-03	0.85E-04
121	22.0	155.0	0.38E-04	0.18E-04	0.10E-04	0.66E-05	0.66E-05	0.71E-05	0.74E-02	0.74E-02	0.69E-02	0.60E-02
131	24.5	217.5	0.47E-02	0.35E-02	0.25E-02	0.16E-02	0.67E-03	0.35E-03	0.19E-03	0.82E-04	0.37E-04	0.19E-04
141	24.5	135.0	0.10E-04	0.69E-05	0.71E-05	0.77E-05	0.49E-02	0.51E-02	0.50E-02	0.46E-02	0.38E-02	0.29E-02
151	28.0	207.5	0.21E-02	0.13E-02	0.55E-03	0.29E-03	0.17E-03	0.74E-04	0.35E-04	0.18E-04	0.10E-04	0.72E-05
161	28.0	115.0	0.76E-05	0.82E-05	0.31E-02	0.33E-02	0.33E-02	0.31E-02	0.27E-02	0.21E-02	0.14E-02	0.85E-03
171	32.5	195.0	0.36E-03	0.20E-03	0.13E-03	0.60E-04	0.30E-04	0.16E-04	0.10E-04	0.73E-05	0.80E-05	0.88E-05
181	38.0	229.0	0.18E-02	0.18E-02	0.18E-02	0.17E-02	0.15E-02	0.11E-02	0.71E-03	0.40E-03	0.17E-03	0.11E-03
191	38.0	175.0	0.81E-04	0.43E-04	0.24E-04	0.14E-04	0.90E-05	0.70E-05	0.81E-05	0.91E-05	0.58E-03	0.59E-03
201	44.5	225.0	0.59E-03	0.55E-03	0.45E-03	0.33E-03	0.21E-03	0.12E-03	0.56E-04	0.47E-04	0.43E-04	0.26E-04
211	44.5	155.0	0.16E-04	0.10E-04	0.75E-05	0.63E-05	0.78E-05	0.91E-05	0.96E-04	0.97E-04	0.96E-04	0.91E-04
221	52.0	217.5	0.77E-04	0.57E-04	0.38E-04	0.24E-04	0.13E-04	0.16E-04	0.19E-04	0.14E-04	0.97E-05	0.71E-05
231	52.0	135.0	0.57E-05	0.54E-05	0.72E-05	0.87E-05	0.11E-04	0.11E-04	0.11E-04	0.10E-04	0.90E-05	0.70E-05
241	60.5	207.5	0.50E-05	0.34E-05	0.22E-05	0.46E-05	0.70E-05	0.62E-05	0.51E-05	0.44E-05	0.40E-05	0.43E-05
251	60.5	115.0	0.64E-05	0.80E-05	0.85E-06	0.86E-06	0.87E-06	0.85E-06	0.77E-06	0.63E-06	0.48E-06	0.35E-06
261	70.0	195.0	0.32E-06	0.12E-05	0.24E-05	0.26E-05	0.26E-05	0.26E-05	0.28E-05	0.34E-05	0.54E-05	0.70E-05
271	80.5	229.0	0.57E-07	0.57E-07	0.57E-07	0.55E-07	0.47E-07	0.35E-07	0.22E-07	0.13E-07	0.29E-07	0.36E-06
281	80.5	175.0	0.96E-06	0.12E-05	0.14E-05	0.16E-05	0.20E-05	0.27E-05	0.45E-05	0.59E-05	0.10E-07	0.87E-08
291	92.0	225.0	0.70E-08	0.24E-08	-0.81E-08	-0.23E-07	-0.39E-07	-0.55E-07	-0.54E-07	0.13E-06	0.63E-06	0.88E-06
301	92.0	155.0	0.11E-05	0.13E-05	0.16E-05	0.21E-05	0.36E-05	0.47E-05	0.44E-07	0.38E-07	0.28E-07	0.38E-08
311	104.5	217.5	-0.47E-07	-0.11E-06	-0.19E-06	-0.27E-06	-0.34E-06	-0.17E-06	0.86E-06	0.10E-05	0.11E-05	0.12E-05
321	104.5	135.0	0.14E-05	0.18E-05	0.28E-05	0.35E-05	0.48E-06	0.35E-06	0.28E-06	0.13E-06	-0.20E-06	-0.51E-06
331	118.0	207.5	-0.78E-06	-0.11E-05	-0.17E-05	-0.15E-05	0.27E-05	0.15E-05	0.12E-05	0.12E-05	0.13E-05	0.15E-05
341	118.0	115.0	0.20E-05	0.20E-05								

TABLE 7.29. Output at the End of Simulation for Example Problem #3, Output File 'VX.OUT'.

Time *** 360.0000 ***

e	x(e)	z(e)	vz(e)	vz(e+1) ...								
1	2.5	229.0	-0.20E-01	-0.20E-01	-0.20E-01	-0.18E-01	-0.14E-01	-0.11E-01	-0.84E-02	-0.62E-02	-0.37E-02	-0.16E-02
11	2.5	175.0	-0.55E-03	-0.19E-03	-0.52E-04	-0.84E-05	-0.42E-06	-0.47E-05	-0.89E-05	-0.39E-05	-0.21E-01	-0.20E-01
21	7.5	225.0	-0.19E-01	-0.17E-01	-0.13E-01	-0.10E-01	-0.77E-02	-0.57E-02	-0.33E-02	-0.14E-02	-0.51E-03	-0.17E-03
31	7.5	155.0	-0.44E-04	-0.51E-05	0.13E-05	-0.36E-05	-0.81E-05	-0.35E-05	-0.22E-01	-0.20E-01	-0.18E-01	-0.15E-01
41	12.0	217.5	-0.12E-01	-0.87E-02	-0.66E-02	-0.49E-02	-0.28E-02	-0.11E-02	-0.42E-03	-0.14E-03	-0.32E-04	0.13E-06
51	12.0	135.0	0.38E-05	-0.23E-05	-0.72E-05	-0.32E-05	-0.28E-01	-0.20E-01	-0.16E-01	-0.12E-01	-0.95E-02	-0.73E-02
61	15.5	207.5	-0.56E-02	-0.42E-02	-0.22E-02	-0.89E-03	-0.34E-03	-0.11E-03	-0.20E-04	0.54E-05	0.64E-05	-0.94E-06
71	15.5	115.0	-0.65E-05	-0.30E-05	-0.18E-01	-0.15E-01	-0.12E-01	-0.98E-02	-0.78E-02	-0.62E-02	-0.48E-02	-0.36E-02
81	18.0	195.0	-0.19E-02	-0.74E-03	-0.28E-03	-0.83E-04	-0.11E-04	0.96E-05	0.84E-05	0.12E-06	-0.59E-05	-0.28E-05
91	20.0	229.0	-0.52E-02	-0.77E-02	-0.80E-02	-0.74E-02	-0.64E-02	-0.53E-02	-0.42E-02	-0.31E-02	-0.15E-02	-0.61E-03
101	20.0	175.0	-0.23E-03	-0.64E-04	-0.27E-05	0.13E-04	0.10E-04	0.10E-05	-0.55E-05	-0.27E-05	-0.19E-02	-0.40E-02
111	22.0	225.0	-0.50E-02	-0.54E-02	-0.51E-02	-0.44E-02	-0.36E-02	-0.26E-02	-0.12E-02	-0.49E-03	-0.18E-03	-0.45E-04
121	22.0	155.0	0.52E-05	0.17E-04	0.12E-04	0.20E-05	-0.50E-05	-0.25E-05	-0.79E-03	-0.21E-02	-0.29E-02	-0.36E-02
131	24.5	217.5	-0.37E-02	-0.34E-02	-0.28E-02	-0.21E-02	-0.91E-03	-0.35E-03	-0.13E-03	-0.22E-04	0.15E-04	0.21E-04
141	24.5	135.0	0.14E-04	0.33E-05	-0.43E-05	-0.23E-05	-0.33E-03	-0.95E-03	-0.15E-02	-0.20E-02	-0.24E-02	-0.23E-02
151	28.0	207.5	-0.19E-02	-0.14E-02	-0.54E-03	-0.18E-03	-0.58E-04	0.61E-05	0.28E-04	0.27E-04	0.17E-04	0.50E-05
161	28.0	115.0	-0.34E-05	-0.21E-05	-0.14E-03	-0.40E-03	-0.66E-03	-0.99E-03	-0.13E-02	-0.12E-02	-0.10E-02	-0.69E-03
171	32.5	195.0	-0.21E-03	-0.36E-04	0.86E-05	0.36E-04	0.42E-04	0.35E-04	0.21E-04	0.72E-05	-0.22E-05	-0.17E-05
181	38.0	229.0	-0.59E-04	-0.14E-03	-0.23E-03	-0.37E-03	-0.48E-03	-0.46E-03	-0.35E-03	-0.20E-03	0.15E-05	0.64E-04
191	38.0	175.0	0.62E-04	0.62E-04	0.55E-04	0.42E-04	0.25E-04	0.96E-05	-0.81E-06	-0.12E-05	-0.17E-04	-0.30E-04
201	44.5	225.0	-0.45E-04	-0.78E-04	-0.93E-04	-0.70E-04	-0.21E-04	0.33E-04	0.98E-04	0.11E-03	0.94E-04	0.81E-04
211	44.5	155.0	0.66E-04	0.48E-04	0.29E-04	0.12E-04	0.57E-06	-0.79E-06	-0.23E-06	0.53E-05	0.10E-04	0.15E-04
221	52.0	217.5	0.29E-04	0.53E-04	0.79E-04	0.10E-03	0.13E-03	0.13E-03	0.11E-03	0.92E-04	0.73E-04	0.52E-04
231	52.0	135.0	0.32E-04	0.14E-04	0.18E-05	-0.38E-06	0.36E-05	0.12E-04	0.20E-04	0.32E-04	0.52E-04	0.75E-04
241	60.5	207.5	0.97E-04	0.11E-03	0.13E-03	0.13E-03	0.12E-03	0.97E-04	0.77E-04	0.55E-04	0.34E-04	0.15E-04
251	60.5	115.0	0.27E-05	-0.57E-07	0.40E-05	0.12E-04	0.21E-04	0.34E-04	0.55E-04	0.78E-04	0.99E-04	0.12E-03
261	70.0	195.0	0.13E-03	0.13E-03	0.12E-03	0.99E-04	0.79E-04	0.57E-04	0.35E-04	0.16E-04	0.33E-05	0.16E-06
271	80.5	229.0	0.40E-05	0.12E-04	0.21E-04	0.34E-04	0.55E-04	0.78E-04	0.10E-03	0.12E-03	0.13E-03	0.13E-03
281	80.5	175.0	0.12E-03	0.10E-03	0.79E-04	0.57E-04	0.36E-04	0.17E-04	0.36E-05	0.30E-06	0.40E-05	0.12E-04
291	92.0	225.0	0.21E-04	0.34E-04	0.55E-04	0.78E-04	0.10E-03	0.12E-03	0.13E-03	0.13E-03	0.12E-03	0.10E-03
301	92.0	155.0	0.79E-04	0.58E-04	0.36E-04	0.17E-04	0.39E-05	0.40E-06	0.40E-05	0.12E-04	0.21E-04	0.34E-04
311	104.5	217.5	0.55E-04	0.78E-04	0.10E-03	0.12E-03	0.13E-03	0.13E-03	0.12E-03	0.10E-03	0.80E-04	0.58E-04
321	104.5	135.0	0.37E-04	0.17E-04	0.42E-05	0.56E-06	0.46E-05	0.13E-04	0.21E-04	0.35E-04	0.56E-04	0.79E-04
331	118.0	207.5	0.10E-03	0.12E-03	0.13E-03	0.12E-03	0.12E-03	0.10E-03	0.80E-04	0.59E-04	0.37E-04	0.18E-04
341	118.0	115.0	0.50E-05	0.10E-05								

TABLE 7.30. Output Data for Example Problem #3, Output File 'RUN_INF.OUT'.

TLevel	Time	dt	Iter	ItCum
14	0.100E+01	0.129E+00	4	47
30	0.500E+01	0.396E+00	3	106
38	0.100E+02	0.694E+00	3	136
52	0.300E+02	0.245E+01	4	189
62	0.600E+02	0.361E+01	3	226
72	0.120E+03	0.727E+01	4	265
78	0.180E+03	0.178E+02	5	288
82	0.240E+03	0.240E+02	5	304
85	0.300E+03	0.300E+02	8	319
90	0.360E+03	0.140E+02	3	333

Real time [min] 529.33333396911619

8. PROGRAM ORGANIZATION AND LISTING

8.1. DESCRIPTION OF PROGRAM UNITS

The program consists of a main program unit and 32 subprograms. The subprograms are organized into 6 source files which are stored and compiled separately and then linked toge ther with the main program to form executable program. The list of source files and subprograms and their short description follows:

SWMII.FOR	(Main program unit)
INPUT2.FOR	BasInf, MatIn, GenMat, TmIn, SeepIn, NodInf, ElemIn, GeomIn,AtmIn,SinkIn
EQUATIO2.FOR	Reset, Dirich, Solve, Shift, SetMat
TIME2.FOR	TmCont, SetAtm, Fqh
MATERIA2.FOR	FK, FC, FQ, FH
SINK2.FOR	SetSnk, FAlfa
OUTPUT2.FOR	TLInf, ALInf, hOut, thOut, QOut, FlxOut, SubReg, BouOut

Main program unit SWMII.FOR

This is the main execution unit of the program SWMII. It controls execution of the program and determines which optional subroutines are necessary for a particular application. It also checks for convergence and updates the pressure head array.

Source file INPUT2.FOR

Subroutines included in this file are designed to read data from different input blocks. Following table summarizes from which input file and input block (described in Chapter 5.) the particular subroutine reads.

TABLE 8.1. Input Subroutines

Subroutine	Input File	Input Block
BasInf	SELECTOR.IN	A. Basic Information
MatIn		B. Material Information
TmIn		C. Time Information
SeepIn		D. Seepage Information
NodInf	GRID.IN	E. Nodal Information
ElemIn		F. Element Information
GeomIn		G. Boundary Geometric Information
AtmIn	ATMOSPH.IN	H. Atmospheric Information
SinkIn	SINK.IN	I. Sink Information

Source file EQUATIO2.FOR

Subroutine **Reset** updates coefficients of governing matrix equation and constructs the right hand side vector and the effective matrix.

Subroutine **Dirich** modifies the effective matrix and right hand side vector to properly account for prescribed pressue head nodes.

Subroutine **Solve** solves the effective matrix by Gaussian elimination.

Subroutine **Shift** changes the type of atmospheric or seepage face boundary condition from Dirichlet type to Neumann type and vice versa.

Subroutine **SetMat** interpolates intermediate values in the material hydraulic characteristics tables in order to determine the nodal values of K(h) and C(h).

Source file TIME2.FOR

Subroutine **TmCont** adjusts the current value of time increment Δt.

Subroutine **SetAtm** updates time variable boundary conditions.

Function **Fqh** describes the groundwater level - discharge relationship q(h). The function is called only from subroutine SetAtm.

Source file MATERIA2.FOR

This file includes functions **FK**, **FC**, **FQ**, and **FH** which describe material hydraulic characteristics K(h), C(h), θ(h), and h(θ), respectively.

Source file SINK2.FOR

This file includes subroutine **SetSnk** and function **FAlfa.** The purpose of these subprograms is to calculate the actual intensity of water extraction in the root zone in response to water uptake by plants.

Source file OUTPUT2.FOR

Subroutines included in this file are designed to print data to different output files. Following table summarizes which output files are filled by the particular subroutine.

TABLE 8.2. Output Subroutines

Subroutine	Output File
TLInf	H_MEAN.OUT V_MEAN.OUT CUM_Q.OUT RUN_INF.OUT
hOut	H.OUT
thOut	TH.OUT
QOut	Q.OUT
FlxOut	VZ.OUT VX.OUT
BouOut	BOUNDARY.OUT
SubReg	BALANCE.OUT
ALInf	A_LEVEL.OUT

8.2. PROGRAM LISTING

```
*!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!*
*                                                                       *
*     SWM II - Numerical model of two-dimensional flow                  *
*              in a variably saturated porous medium                    *
*                                                                       *
*     Designed by T.Vogel    (1987)                                     *
*     Based on model UNSAT2 (Neuman et al.,1974)                        *
*                                                                       *
*!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!*

      program SWMII

      parameter (NumNPD=500,
     !           NumElD=500,
     !           MBandD=30,
     !           NumBPD=100,
     !           NSeepD=2,
     !           NumSPD=20,
     !           NMatD=5,
     !           NTabD=100)

      double precision A,B,Rtime
      logical SinkF,qGWLF,AtmInF,ShortF,SeepF,CheckF,FluxF
      integer PLevel,ALevel,TLevel

      dimension A(MBandD,NumNPD),B(NumNPD),Kode(NumNPD),Q(NumNPD),
     ! hNew(NumNPD),hTemp(NumNPD),hOld(NumNPD),ConSat(NMatD),
     ! hTab(NTabD),ConTab(NTabD,NMatD),CapTab(NTabD,NMatD),
     ! F(NumNPD),Con(NumNPD),Cap(NumNPD),X(NumNPD),Y(NumNPD),
     ! MatNum(NumNPD),LayNum(NumNPD),KX(NumElD,4),KXB(NumBPD),
     ! ConAxx(NumElD),ConAzz(NumElD),ConAxz(NumElD),TPrint(50),
     ! NP(NSeepD,NumSPD),NSP(NSeepD),KodeS(NSeepD),
     ! Width(NumBPD),SWidth(4),
     ! Par(20,NMatD),Axz(NumNPD),Bxz(NumNPD),Dxz(NumNPD),thr(NMatD),
     ! Sink(NumNPD),POptm(NMatD),Beta(NumNPD),DS(NumNPD),CumQ(20)

      open(30,file='SWMII.IN\Selector.IN',  status='OLD')
      open(31,file='SWMII.IN\Atmosph.IN',   status='OLD')
      open(32,file='SWMII.IN\Grid.IN',      status='OLD')
      open(33,file='SWMII.IN\Sink.IN',      status='OLD')
      open(50,file='SWMII.OUT\Check.OUT',   status='NEW')
      open(71,file='SWMII.OUT\v_Mean.OUT',  status='NEW')
      open(77,file='SWMII.OUT\h_Mean.OUT',  status='NEW')
      open(78,file='SWMII.OUT\Cum_Q.OUT',   status='NEW')
      open(72,file='SWMII.OUT\A_Level.OUT', status='NEW')
      open(75,file='SWMII.OUT\h.OUT',       status='NEW')
      open(76,file='SWMII.OUT\th.OUT',      status='NEW')
      open(79,file='SWMII.OUT\Boundary.OUT',status='NEW')
      open(80,file='SWMII.OUT\Balance.OUT', status='NEW')
      open(82,file='SWMII.OUT\vx.OUT',      status='NEW')
      open(81,file='SWMII.OUT\vz.OUT',      status='NEW')

      data SinkF  ,qGWLF  ,tInit,NTab,ItCum,Iter,TLevel,ALevel,PLevel
     !    /.false.,.false., 0.  ,100 , 0.  , 0. ,  1   , 1    ,  1  /
     !
     !     CumQ ,Sink
     !    /20*0.,NumNPD*0./
```

```
      call BasInf (Kat,MaxIt,Tol,AtmInF,ShortF,SeepF,CheckF,FluxF)
      call NodInf (NumNP,NumEl,IJ,NumBP,MBand,NumNPD,Kode,Q,
     !             hNew,hOld,hTemp,X,Y,MatNum,Beta,Axz,Bxz,Dxz,CheckF)
      call ElemIn (NumEl,NumElD,KX,LayNum,ConAxx,ConAzz,ConAxz,CheckF)
      call GeomIn (NumNP,NumBP,SWidth,Width,Kode,KXB,rLen)
      call MatIn  (NMatD,NMat,NLay,Par,hTab(1),hTab(NTab))
      call GenMat (NumNP,NTab,NTabD,NMat,NMatD,thr,
     !             hNew,MatNum,Par,hTab,ConTab,CapTab,ConSat)
      if (AtmInF) then
      call AtmIn  (GWLOL,SinkF,qGWLF,tInit,tMax,Aqh,Bqh,hCritS,MaxAL)
      call SetAtm (tAtm,rTop,rRoot,hCritA,Width,KXB,NumBP,
     !             Kode,hNew,Q,NumNP,Aqh,Bqh,GWLOL,qGWLF)
      endif
      call TmIn   (tInit,tMax,TAtm,tOld,dt,dtMax,DMul,DMul2,dtMin,
     !             TPrint,t,DtOpt,AtmInF)
      if (SinkF) then
      call SinkIn (NMat,NumEl,NumNP,NumElD,Kat,KX,x,y,
     !             P0,POptm,P2H,P2L,P3,r2H,r2L,Beta)
      call SetSnk (NumNP,NumNPD,NMatD,MatNum,hNew,rRoot,
     !             Sink,P0,POptm,P2H,P2L,P3,r2H,r2L,Beta,rLen)
      endif
      if (SeepF)
     !call SeepIn (NSeepD,NumSPD,NSeep,NSP,NP,KodeS)
      call date   (i,i,iday)
      call time   (ihours,mins,isecs,i)
      Rtime=iday*24.*60.+ihours*60.+mins+isecs/60.
      close(30)
      close(33)
      open(70,file='SWMII.OUT\Run_Inf.OUT', status='NEW')
      open(74,file='SWMII.OUT\Q.OUT',       status='NEW')

* --- Beginning of time and iteration loop ----------------------------

  250 continue

      call SetMat(NumNP,NTab,NTabD,NMat,NMatD,hTab,ConTab,CapTab,
     !            hNew,hOld,MatNum,Par,Con,Cap,ConSat,Axz,Bxz,Dxz)
      call Reset (KODE,A,B,Q,hNew,hOld,F,Con,Cap,X,Y,KX,NumNP,MBand,
     !            NumEl,NumNPD,MBandD,NumElD,KAT,dt,
     !            SinkF,Sink,DS,Beta,rLen,vMeanR,hMeanR,
     !            ConAxx,ConAzz,ConAxz)
      if (AtmInF.or.SeepF)
     !call Shift (NumNP,NumBP,NSeepD,NumSPD,NSeep,NSP,NP,hNew,Q,Kode,
     !            KodeS,rTop,Width,KXB,hCritA,hCritS,SeepF,AtmInF)
      call Dirich(A,B,KODE,hNew,NUMNP,MBAND,NumNPD,MBandD)
      call Solve (A,B,NUMNP,MBAND,NumNPD,MBandD)

      do 260 i=1,NumNP
        hNew(i)=B(i)
  260 continue
      ITER =ITER+1
      ItCum=ItCum+1
      if ( MaxIt.le.1 ) goto 460
      if ( dt.le.dtMin ) then
        write( *,*) ' dt is equal to dtMin'
        goto 460
      endif
```

```
*     Test for convergence
      i=0
  410 continue
        i=i+1
        Epslon = abs(hNew(i)-hTemp(i))
      if (Epslon.le.Tol .and. i.lt.NumNP) goto 410
      if (Epslon.gt.Tol .and. Iter.le.MaxIt) then
        do 430 i=1,NumNP
          hTemp(i)=hNew(i)
  430   continue
        goto 250
      endif
      if (Epslon.gt.TOL .and. Iter.gt.MaxIt) then
        do 455 i=1,NumNP
          hNew(i) =hOld(i)
          hTemp(i)=hOld(i)
  455   continue
        Iter=0
        dt=amax1(dt/10,dtMin)
        dtOpt=dt
        t=tOld+dt
        goto 250
      endif

* --- end of iteration loop -----------------------------------------------

  460 continue

*     T-Level information
      call TLInf   (NumNP,NumBP,Kode,Q,hNew,CumQ,Width,SWidth,KXB,
     !              t,dt,TLevel,ShortF,TPrint(PLevel),Iter,ItCum,rTop,
     !              rRoot,vMeanR,hMeanT,hMeanR,hMeanG,AtmInF,SinkF)

*     P-Level information
      if (abs(TPrint(PLevel)-t).lt.0.001*dt) then
        call HOut  (hNew,X,Y,NumNP,t,IJ)
        call thOut (NMatD,NumNP,t,IJ,Par,hNew,X,Y,MatNum,thr,Axz,Dxz)
        if (FluxF) then
          call QOut  (Q,X,Y,NumNP,t,IJ)
          call FlxOut(NumNP,NumEl,NumElD,ConAxx,ConAzz,ConAxz,
     !    hNew,X,Y,IJ,Con,KX,t)
        endif
        call SubReg(NumEl,NumElD,NumNP,NMatD,hNew,hOld,x,y,MatNum,
     !  LayNum,Par,KX,Kat,t,dt,NLay,PLevel,thr,Axz,Dxz)
        call BouOut(NumNP,NumBP,NMatD,t,hNew,Q,Width,Par,KXB,Kode,
     !  MatNum,X,Y,thr,Axz,Dxz)
        PLevel=PLevel+1
      endif

*     A-level information
      if (abs(t-tAtm).le.0.001*dt .and. AtmInF) then
        call ALInf (t,CumQ,hMeanT,hMeanR,hMeanG,ALevel)
        if (ALevel.lt.MaxAL) then
          call SetAtm (tAtm,rTop,rRoot,hCritA,Width,KXB,NumBP,
     !    Kode,hNew,Q,NumNP,Aqh,Bqh,GWL0L,qGWLF)
          ALevel=ALevel+1
        endif
      endif
```

```
*     Root extraction
      if (SinkF)
     !call SetSnk (NumNP,NumNPD,NMatD,MatNum,hNew,rRoot,
     !             Sink,P0,POptm,P2H,P2L,P3,r2H,r2L,Beta,rLen)

*     Pressure heads for new time level
      do 470 i=1,NumNP
        hOld(i) =hNew(i)
        hTemp(i)=hNew(i)
  470 continue

*     Time governing
      if (abs(t-tMax).le.0.001*dt) then
        call date(i,i,iday)
        call time(ihours,mins,isecs,i)
        Rtime=iday*24.*60.+ihours*60.+mins+isecs/60. - Rtime
        write(70,*)
        write(70,*) 'Real time [min]',Rtime
        write( *,*) 'Real time [min]',Rtime
        stop
      endif
      call TmCont (dt,DTMAX,DtOpt,DMUL,DMul2,dtMin,
     !ITER,TPrint(PLevel),tAtm,t,tMax)
      Iter=0
      TLevel=TLevel+1
      tOld=t
      t=t+dt
      goto 250

* --- end of time loop ------------------------------------------------

      end

*||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
```

```
* Source file INPUT2.FOR ::::::::::::::::::::::::::::::::::::::::::::::::

      subroutine BasInf (Kat,MaxIt,Tol,AtmInF,ShortF,SeepF,CheckF,FluxF)

      character*72 Hed
      character*5  LUnit,TUnit
      logical CheckF,AtmInF,ShortF,SeepF,FluxF

      dimension IU(11)
      data IU /50,71,72,75,76,77,78,79,80,81,82/

      read(30,*)
      read(30,*)
      read(30,*) Hed
      read(30,*)
      read(30,*) LUnit,TUnit
      read(30,*)
      read(30,*) Kat
      read(30,*)
      read(30,*) MaxIt,Tol
      read(30,*)
      read(30,*) CheckF,ShortF,FluxF,AtmInF,SeepF
      call time (ihours,mins,isecs,ihunds)
      call date (iyear,month,iday)
      Rtime=ihours*60.+mins+isecs/60.
      do 10 i=1,11
        write(IU(i),  *) Hed
        write(IU(i),  *)
        write(IU(i),  *)'Program SWM II'
        write(IU(i),150) iday,month,iyear,ihours,mins
        if (AtmInF) then
          write(IU(i),*)'Time dependent boundary conditions'
        else
          write(IU(i),*)'Time independent boundary conditions'
        endif
        if (KAT.eq.0) write(IU(i),100)
        if (KAT.eq.1) write(IU(i),110)
        if (KAT.eq.2) write(IU(i),120)
        write(IU(i),*) 'Units: L = ',LUnit,', T = ',TUnit
10    continue
      write(*,*) '*** ',Hed,' ***'
      if (KAT.eq.0) write(*,100)
      if (KAT.eq.1) write(*,110)
      if (KAT.eq.2) write(*,120)
      write(50,140) MAXIT,TOL
100   format (' Horizontal plane flow, V = L*L')
110   format (' Axisymmetric flow, V = L*L*L')
120   format (' Vertical plane flow, V = L*L')
140   format(/' Max. number of iterations  ',I4/
     !        ' Pressure head tolerance [L]',F8.4/)
150   format (' Date: ',i2,'.',i2,'.',i4,'   Time: ',i2,':',i2)
      return
      end

************************************************************************
```

```
      subroutine MatIn (NMatD,NMat,NLay,Par,hTab1,hTabN)

      real K
      dimension Par(20,NMatD),Qe(7)
      data Qe /1.,.99,.90,.85,.75,.65,.50/

      Imax=7
      read (30, *)
      read (30, *)
      read (30, *) NMat,NLay,hTab1,hTabN,NPar
      hTab1=-amin1( abs(hTab1) , abs(hTabN) )
      hTabN=-amax1( abs(hTab1) , abs(hTabN) )
      read (30, *)
      write(50,200)
      do 10 M=1,NMat
        read (30, *)    (Par(i,M),i=1,NPar)
        write(50,210) M,(Par(i,M),i=1,NPar)
   10 continue
      write(50,220)
      do 50 M=1,NMat
        write(50, *)
        do 40 i=1,Imax
          h= FH( Qe(i),Par(1,M) )
          K= FK( h,Par(1,M) )
          C= FC( h,Par(1,M) )
          Q= FQ( h,Par(1,M) )
          write(50,230) M,Qe(i),Q,h,C,K
   40   continue
   50 continue
  200 format(/' MatNum, Param. array:'/)
  210 format(I5,8X,4f7.3,16E12.3)
  220 format(//' MatNum       Qe     Q      h         C         K')
  230 format(I5,8X,2F7.3,F10.3,e10.2,E12.3)
      return
      end

*************************************************************************
```

```
      subroutine GenMat (NumNP,NTab,NTabD,NMat,NMatD,thr,
     !                   hNew,MatNum,Par,hTab,ConTab,CapTab,ConSat)

      dimension hTab(NTab),ConTab(NTabD,NMatD),CapTab(NTabD,NMatD),
     !          hNew(NumNP),MatNum(NumNP),Par(20,NMatD),ConSat(NMat),
     !          thr(NMat)

      hTab1=hTab(1)
      hTabN=hTab(NTab)
      if (hTabN.ne.hTab1) then
        dlh=( alog10(-hTabN)-alog10(-hTab1) )/(NTab-1)
        do 5 i=1,NTab
          alh=alog10(-hTab1)+(i-1)*dlh
          hTab(i)=-10**alh
5       continue
        do 20 M=1,NMat
          ConSat(M)=FK(0.0,Par(1,M))
          thr(M)=FQ(-1e25,Par(1,M))
          do 10 i=1,NTab
            ConTab(i,M)=FK(hTab(i),Par(1,M))
            CapTab(i,M)=FC(hTab(i),Par(1,M))
10        continue
20      continue
      endif
      return
      end

************************************************************************

      subroutine TmIn (tInit,tMax,tAtm,tOld,dt,dtMax,DMul,DMul2,dtMin,
     !                 TPrint,t,DtOpt,AtmInF)

      logical AtmInF
      dimension TPrint(50)

      read (30, *)
      read (30, *)
      read (30, *) dt,dtMin,dtMax,DMul,DMul2,MPL
      read (30, *)
      read (30, *) (TPrint(i),i=1,MPL)
      DtOpt=dt
      if (.not.AtmInF) then
        tMax=TPrint(MPL)
        tAtm=tMax
      endif
      TPrint(MPL+1)=tMax
      tOld=TInit
      t=tInit+dT
      return
      end

************************************************************************
```

```
      subroutine SeepIn (NSeepD,NumSPD,NSeep,NSP,NP,KodeS)

      dimension NSP(NSeepD),NP(NSeepD,NumSPD),KodeS(NSeepD)

      read(30, *)
      read(30, *)
      read(30, *) NSeep
      read(30, *)
      read(30, *) (NSP(i),i=1,NSeep)
      read(30, *)
      read(30, *) (KodeS(i),i=1,NSeep)
      read(30, *)
      do 20 I=1,NSeep
        read(30,*) (NP(I,J),J=1,NSP(I))
20    continue
      return
      end

************************************************************************

      subroutine NodInf (NumNP,NumEl,IJ,NumBP,MBand,NumNPD,Kode,Q,
     !                   P,P1,hTemp,X,Y,MatNum,Beta,Axz,Bxz,Dxz,CheckF)
      logical CheckF

      dimension KODE(NumNPD),Q(NumNPD),P1(NumNPD),X(NumNPD),Y(NumNPD),
     !          P(NumNPD),hTemp(NumNPD),MatNum(NumNPD),Beta(NumNPD),
     !          Axz(NumNPD),Bxz(NumNPD),Dxz(NumNPD)

      read(32,*)
      read(32,*)
      read(32,*) NumNP,NumEl,IJ,NumBP
      read(32,*)
      MBand=IJ+2
      NPR=0
      L=0

20    L=L+1
      read(32,*) N,KODE(N),X(N),Y(N),P1(N),Q(N),MatNum(N),Beta(N),
     !Axz(N),Bxz(N),Dxz(N)
      IF (N-L) 50,90,70
50    write(*,230) N
      stop
70    DENO=N-L+1
      DX=(X(N)-X(NPR))/DENO
      DY=(Y(N)-Y(NPR))/DENO
      DP=(P1(N)-P1(NPR))/DENO
      DBeta=(Beta(N)-Beta(NPR))/Deno
      DA=(Axz(N)-Axz(NPR))/Deno
      DB=(Bxz(N)-Bxz(NPR))/Deno
      DD=(Dxz(N)-Dxz(NPR))/Deno
80    X(L)=X(L-1)+DX
      Y(L)=Y(L-1)+DY
      P1(L)=P1(L-1)+DP
      Beta(L)=Beta(L-1)+DBeta
      Axz(L)=Axz(L-1)+DA
      Bxz(L)=Bxz(L-1)+DB
      Dxz(L)=Dxz(L-1)+DD
      MatNum(L)=MatNum(L-1)
      KODE(L)=KODE(L-1)
      Q(L)=Q(L-1)
```

```
      L=L+1
      IF (L .LT. N) GO TO 80
   90 NPR=N
      IF (L .LT. NUMNP) GO TO 20

      do 105 N=1,NUMNP
        P(N)=P1(N)
        hTemp(N)=P1(N)
  105 continue
      if (CheckF) then
        write(50,210)
        do 110 N=1,NUMNP
          write(50,200) N,KODE(N),X(N),Y(N),P1(N),Q(N),MatNum(N),
     !    Beta(N),Axz(N),Bxz(N),Dxz(N)
  110   continue
      endif
  200 format (2I10,4E15.6,i10,4f7.3)
  210 format (////24H NODAL POINT INFORMATION //10H  NODE NO.,6X,4HKODE,
     '7X,3HX,R,12X,3HY,Z,11X,5H.PSI.,12X,1HQ/)
  230 format (' ERROR IN NodInf AT N=',I5)
      return
      end

**************************************************************************

      subroutine ElemIn (NumEl,NumElD,KX,LayNum,ConAxx,ConAzz,ConAxz,
     !                   CheckF)
      logical CheckF

      dimension KX(NumElD,4),ConAxx(NumEl),ConAzz(NumEl),ConAxz(NumEl),
     !          LayNum(NumEl)
      NUM=0
      read(32,*)
      read(32,*)
      do 60 N=1,NUMEL
        IF (NUM-N) 20,60,40
   20   read(32,*)NUM,(KX(NUM,I),I=1,4),ConAxz(Num),ConAxx(Num),
     ! ConAzz(Num),LayNum(Num)
        IF (KX(NUM,4) .EQ. 0) KX(NUM,4)=KX(NUM,3)
        IF (NUM .EQ. N) GO TO 60
   40   do 50 I=1,4
          KX(N,I)=KX(N-1,I)+1
   50   continue
        ConAxx(N)=ConAxx(N-1)
        ConAzz(N)=ConAzz(N-1)
        ConAxz(N)=ConAxz(N-1)
        LayNum(N)=LayNum(N-1)
   60 continue
      AA=3.141592654/180.
      do 80 N=1,NumEl
        Ang=AA*ConAxz(N)
        CAxx=ConAxx(N)
        CAzz=ConAzz(N)
        ConAxx(N)=CAxx*cos(Ang)*cos(Ang) + CAzz*sin(Ang)*sin(Ang)
        ConAzz(N)=CAxx*sin(Ang)*sin(Ang) + CAzz*cos(Ang)*cos(Ang)
        ConAxz(N)=(CAxx-CAzz)*sin(Ang)*cos(Ang)
   80 continue
      if (CheckF) then
        write(50,100)
        do 90 N=1,NumEl
```

```
      write(50,130) N,(KX(N,I),I=1,4),ConAxz(N),ConAxx(N),ConAzz(N),
   !    LayNum(N)
 90   continue
      endif
100 format (////20H ELEMENT INFORMATION//' ELEMENT    C O R N E R    N
   ! O D E S   ConAxz    ConAxx    ConAzz  LayNum'/)
130 format (I6,I9,3I6,E14.3,2F8.3,i5)
      return
      end

*************************************************************************

      subroutine GeomIn (NumNP,NumBP,SWidth,Width,Kode,KXB,rLen)

      dimension KXB(NumBP),Width(NumBP),SWidth(4),Kode(NumNP)

      read(32,*)
      read(32,*)
      read(32,*) (KXB(i),i=1,NumBP)
      read(32,*)
      read(32,*) (Width(i),i=1,NumBP)
      read(32,*)
      read(32,*) rLen
      do 10 i=1,4
        SWidth(i)=0
 10   continue
      do 20 i=1,NumBP
        n=KXB(i)
        j=iabs(Kode(n))
        if (j.eq.0) goto 20
        SWidth(j)=SWidth(j)+Width(i)
 20   continue
      return
      end

*************************************************************************

      subroutine AtmIn (GWL0L,SinkF,qGWLF,tInit,tMax,Aqh,Bqh,hCritS,
   !                    MaxAL)
      logical SinkF,qGWLF,DummyF

      read (31, *)
      read (31, *)
      read (31, *)
      read (31, *)
      read (31, *) SinkF,DummyF,qGWLF
      read (31, *)
      read (31, *) GWL0L,Aqh,Bqh
      read (31, *)
      read (31, *) tInit,MaxAL
      read (31, *)
      read (31, *) hCritS
      read (31, *)
      do 10 i=1,MaxAL-1
        read (31, *)
 10   continue
      read (31, *) tMax
      rewind 31
      do 20 i=1,12
        read (31, *)
```

```
  20 continue
     return
     end

********************************************************************************

     subroutine SinkIn (NumMat,NumEl,NumNP,NumElD,Kat,KX,x,y,
    !                   P0,POptm,P2H,P2L,P3,r2H,r2L,Beta)

     dimension POptm(NumMat),Beta(NumNP),KX(NumElD,4),x(NumNP),y(NumNP)

     read(33, *)
     read(33, *)
     read(33, *) P0,P2H,P2L,P3,r2H,r2L
     read(33, *)
     read(33, *) (POptm(i),i=1,NumMat)
     P0 =-abs(P0)
     P2L=-abs(P2L)
     P2H=-abs(P2H)
     P3 =-abs(P3)
     xMul=1
     SBeta=0
     do 200 N=1,NumEl
       NUS=4
       IF (KX(N,3) .EQ. KX(N,4)) NUS=3
       do 120 K=1,NUS-2
         I=KX(N,1)
         J=KX(N,K+1)
         L=KX(N,K+2)
         CJ=X(I)-X(L)
         CK=X(J)-X(I)
         BJ=Y(L)-Y(I)
         BK=Y(I)-Y(J)
         AE=(CK*BJ-CJ*BK)/2
         if (KAT.eq.1) xMul=2*3.1416*(X(I)+X(J)+X(L))/3
         BetaE=(Beta(I)+Beta(J)+Beta(L))/3
         SBeta=SBeta+xMul*AE*BetaE
 120   continue
 200 continue
     do 210 i=1,NumNP
       Beta(i)=Beta(i)/SBeta
 210 continue
     return
     end

* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
* Source file EQUATIO2.FOR ||||||||||||||||||||||||||||||||||||||||||||||||

      subroutine Reset (Kode,A,B,Q,P,P1,F,Con,Cap,X,Y,KX,NumNP,MBand,
     !                  NumEl,NumNPD,MBandD,NumElD,KAT,dt,
     !                  SinkF,Sink,DS,Beta,rLen,vMeanR,hMeanR,
     !                  ConAxx,ConAzz,ConAxz)
      logical SinkF
      double precision A,B

      dimension A(MBandD,NumNPD),B(NumNP),Q(NumNP),P(NumNP),P1(NumNP),
     ! F(NumNP),Con(NumNP),Cap(NumNP),X(NumNP),Y(NumNP),
     ! KX(NumElD,5),ConAxx(NumEl),ConAzz(NumEl),ConAxz(NumEl),
     ! Kode(NumNP),E(3,3),iLoc(3),Sink(NumNP),DS(NumNP),Beta(NumNP)

*     Initialisation
      xMul=1
      vMeanR  =0
      hMeanR=0
      AreaR =0
      do 40 I=1,NumNP
        B(I) =0.
        F(I) =0.
        DS(I)=0.
   20   do 30 J=1,MBand
          A(J,I)=0.
   30   continue
   40 continue

*     Loop on elements
      do 200 N=1,NumEl
        CONDI=ConAxx(N)
        CONDJ=ConAzz(N)
        CONDK=ConAxz(N)
        NUS=4
        if (KX(N,3).eq.KX(N,4)) NUS=3

*       Loop on subelements
        do 120 K=1,NUS-2
          I=KX(N,1)
          J=KX(N,K+1)
          L=KX(N,K+2)
          iLoc(1)=1
          iLoc(2)=K+1
          iLoc(3)=K+2
          CI=X(L)-X(J)
          CJ=X(I)-X(L)
          CK=X(J)-X(I)
          BI=Y(J)-Y(L)
          BJ=Y(L)-Y(I)
          BK=Y(I)-Y(J)
          AE=(CK*BJ-CJ*BK)/2
          CapE=(Cap(I)+Cap(J)+Cap(L))/3
          ConE=(Con(I)+Con(J)+Con(L))/3
          if (KAT.eq.1) xMul=2*3.1416*(X(I)+X(J)+X(L))/3
          AMul=xMul*ConE/4/AE
          BMul=xMul*ConE/2
          FMul=xMul*AE/12
          BetaE=(Beta(I)+Beta(J)+Beta(L))/3
          if (SinkF.and.BetaE.gt.0) then
            SinkE=(Sink(I)+Sink(J)+Sink(L))/3
```

```
          DS(I)=DS(I)+FMul*(3*SinkE+Sink(I))
          DS(J)=DS(J)+FMul*(3*SinkE+Sink(J))
          DS(L)=DS(L)+FMul*(3*SinkE+Sink(L))
          PE=(P(I)+P(J)+P(L))/3
          vMeanR=vMeanR+xMul*AE*SinkE/rLen
          hMeanR=hMeanR+xMul*PE*AE
          AreaR=AreaR+xMul*AE
        endif
        F(I)=F(I)+FMul*(3*CapE+Cap(I))
        F(J)=F(J)+FMul*(3*CapE+Cap(J))
        F(L)=F(L)+FMul*(3*CapE+Cap(L))
        if (KAT.ge.1) then
          B(I)=B(I)+BMul*(CONDK*BI+CONDJ*CI)
          B(J)=B(J)+BMul*(CONDK*BJ+CONDJ*CJ)
          B(L)=B(L)+BMul*(CONDK*BK+CONDJ*CK)
        endif
        E(1,1)=CONDI*BI*BI+2.*CONDK*BI*CI+CONDJ*CI*CI
        E(1,2)=CONDI*BI*BJ+CONDK*(BI*CJ+CI*BJ)+CONDJ*CI*CJ
        E(1,3)=CONDI*BI*BK+CONDK*(BI*CK+CI*BK)+CONDJ*CI*CK
        E(2,1)=E(1,2)
        E(2,2)=CONDI*BJ*BJ+2.*CONDK*BJ*CJ+CONDJ*CJ*CJ
        E(2,3)=CONDI*BJ*BK+CONDK*(BJ*CK+BK*CJ)+CONDJ*CJ*CK
        E(3,1)=E(1,3)
        E(3,2)=E(2,3)
        E(3,3)=CONDI*BK*BK+2.*CONDK*BK*CK+CONDJ*CK*CK
        do 110 i=1,3
          iG=KX(N,iLoc(i))
          do 100 j=1,3
            jG=KX(N,iLoc(j))
            iB=iG-jG+1
            if (iB.ge.1) A(iB,jG)=A(iB,jG) + AMul*E(i,j)
100       continue
110     continue
120   continue

200 continue
    if (AreaR.gt.0.) hMeanR=hMeanR/AreaR

*     Determine boundary fluxes
    do 240 N=1,NumNP
      if (Kode(N).LT.1) GO TO 240
      QN=B(N) + A(1,N)*P(N) + DS(N)
      do 230 J=2,MBand
        K=N-J+1
        if (K .LT. 1) goto 220
        QN=QN + A(J,K)*P(K)
220     K=N+J-1
        if (K .GT. NumNP) goto 230
        QN=QN + A(J,N)*P(K)
230   continue
      Q(N)=QN
240 continue

*     Complete construction of RHS vector and form effective matrix
    do 310 I=1,NumNP
      B(I)=Q(I) - B(I) + F(I)*P1(I)/dt - DS(I)
      A(1,I)=A(1,I) + F(I)/dt
310 continue

    return
```

```
      end

*************************************************************************

      subroutine Dirich (A,B,KODE,P,NUMNP,MBAND,NumNPD,MBandD)

      DOUBLE PRECISION A,B
      DIMENSION A(MBandD,NumNPD),B(NUMNP),KODE(NUMNP),P(NUMNP)

      DO 70 N=1,NUMNP
        IF (KODE(N) .LT. 1) GO TO 70
   10   DO 60 M=2,MBAND
          K=N-M+1
          IF (K) 30,30,20
   20     B(K)=B(K)-A(M,K)*P(N)
          A(M,K)=0.
   30     L=N+M-1
          IF (NUMNP-L) 50,40,40
   40     B(L)=B(L)-A(M,N)*P(N)
   50     A(M,N)=0.
   60   CONTINUE
        A(1,N)=1.
        B(N)=P(N)
   70 CONTINUE
      RETURN
      END

*************************************************************************

      subroutine Solve (A,B,NUMNP,MBAND,NumNPD,MBandD)

      DIMENSION A(MBandD,NumNPD),B(NUMNP)
      DOUBLE PRECISION A,B,C

*     Reduction
      DO 30 N = 1,NUMNP
      DO 20 M=2,MBAND
      IF (A(M,N) .EQ.0.) GO TO 20
      C = A(M,N)/A(1,N)
      I = N + M - 1
      IF (I .GT. NUMNP) GO TO 20
      J = 0
      DO 10 K = M,MBAND
      J = J + 1
   10 IF (A(K,N) .NE. 0.) A(J,I) = A(J,I) - C*A(K,N)
      A(M,N) = C
      B(I) = B(I) - A(M,N)*B(N)
   20 CONTINUE
      B(N) = B(N)/A(1,N)
   30 CONTINUE

*     Back substitution
      N = NUMNP
   40 DO 50 K = 2,MBAND
      L = N + K - 1
      IF (L .GT. NUMNP) GO TO 60
   50 IF (A(K,N) .NE. 0.) B(N)=B(N)-A(K,N)*B(L)
   60 N = N - 1
      IF (N .GT. 0) GO TO 40
```

```
      RETURN
      END

************************************************************************

      subroutine Shift (NumNP,NumBP,NSeepD,NumSPD,NSeep,NSP,NP,P,Q,Kode,
     !                  KodeS,EI,Width,KXB,hCritA,hCritS,SeepF,AtmInF)
      logical SeepF,AtmInF

      dimension Kode(NumNP),Q(NumNP),P(NumNP),Width(NumBP),KXB(NumBP),
     !          NP(NSeepD,NumSPD),NSP(NSeepD),KodeS(NSeepD)

*     Modify conditions on seepage faces
      if (SeepF) then
        do 310 I=1,NSEEP
          ICHECK=0
          NS=NSP(I)
          do 300 J=1,NS
            N=NP(I,J)
            if (KODE(N) .ne. -2) goto 280
            if (P(N) .lt. 0.) ICHECK=1
            if (ICHECK .gt. 0) goto 300
            KODE(N)=2
            P(N)=0.
            goto 300
 280        if (KODE(N) .ne. 2) goto 300
            if (ICHECK .gt. 0) goto 290
            if ((Q(N) .lt. 0. .AND. KODES(I) .lt. 0) .or.
     !          (Q(N) .gt. 0. .AND. KODES(I) .gt. 0)) goto 300
 290        KODE(N)=-2
            Q(N)=0.
            ICHECK=1
 300      continue
 310    continue
      endif

*     Modify potential surface flux boundaries
      if (AtmInF) then
        do 380 i=1,NumBP
          n=KXB(i)
          K=Kode(n)

*         Critical surface pressure on ...
          if (K.eq.4) then
            if(abs(Q(n)).gt.abs(-EI*Width(i)) .or. Q(n)*(-EI).le.0) then
              Kode(n)=-4
              Q(n)=-EI*Width(i)
            endif
            goto 380
          endif

*         Surface flux on ...
          if (K.eq.-4) then
            if (P(n).le.hCritA) then
              Kode(n)=4
              P(n)=hCritA
              goto 380
            endif
            if (P(n).ge.hCritS) then
              Kode(n)=4
```

```
          P(n)=hCritS
        endif
      endif

380   continue
    endif
    return
    end

********************************************************************************

    subroutine SetMat(NumNP,NTab,NTabD,NMat,NMatD,hTab,ConTab,CapTab,
   !                  hNew,hOld,MatNum,Par,Con,Cap,ConSat,Axz,Bxz,Dxz)

    dimension hTab(NTab),ConTab(NTabD,NMatD),CapTab(NTabD,NMatD),
   !          hNew(NumNP),hOld(NumNP),MatNum(NumNP),Par(20,NMatD),
   !          Con(NumNP),Cap(NumNP),ConSat(NMat),
   !          Axz(NumNP),Bxz(NumNP),Dxz(NumNP)

    alhTb1=alog10(-hTab(1))
    dlh =( alog10(-hTab(NTab))-alhTb1 )/(NTab-1)
    do 10 i=1,NumNP
      M=MatNum(i)
      Con(i)=ConSat(M)*Bxz(i)
      hNw=hNew(i)/Axz(i)
      if (  hNw.ge.hTab(NTab) .and. hNw.lt.hTab(1) ) then
        iTab=int( ( alog10(-hNw)-alhTb1 )/dlh )+1
        dhi=hTab(iTab+1)-hTab(iTab)
        slope=( ConTab(iTab+1,M)-ConTab(iTab,M) )/dhi
        Con(i)=( ConTab(iTab,M)+slope*(hNw-hTab(iTab)) )*Bxz(i)
      endif
      if ( hNw.ge.hTab(1) .and. hNw.lt.0
   !  .or. hNw.lt.hTab(NTab) ) Con(i)=FK(hNw,Par(1,M))*Bxz(i)
      Cap(i)=0
      hMean=(hNew(i)+hOld(i))/2/Axz(i)
      if ( hMean.ge.hTab(NTab) .and. hMean.lt.hTab(1) ) then
        iTab=int( ( alog10(-hMean)-alhTb1 )/dlh )+1
        dhi=hTab(iTab+1)-hTab(iTab)
        slope=(CapTab(iTab+1,M)-CapTab(iTab,M))/dhi
        Cap(i)=(CapTab(iTab,M)+slope*(hMean-hTab(iTab)))*Dxz(i)/Axz(i)
      endif
      if ( hMean.ge.hTab(1) .and. hMean.lt.0
   !  .or. hMean.lt.hTab(NTab))Cap(i)=FC(hMean,Par(1,M))*Dxz(i)/Axz(i)
 10 continue
    return
    end

* ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
```

```
* Source file TIME2.FOR !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

      subroutine TmCont (dt,dtMax,dtOpt,DMul,DMul2,dtMin,
     !                   ITER,tPrint,tAtm,t,tMax)

      tFix=amin1(tPrint,tAtm,tMax)
      if (ITER.le.3.and.(tFix-t).ge.DMul*dtOpt)
     !dtOpt=amin1( dtMax,DMul*dtOpt )
      if (ITER.ge.7)
     !dtOpt=amax1( dtMin,DMul2*dtOpt )
      dt=amin1( dtOpt,tFix-t )
      dt=amin1( (tFix-t)/anint((tFix-t)/dt),dtMax )
      return
      end

************************************************************************

      subroutine SetAtm (tAtm,rTop,rRoot,hCritA,Width,KXB,NumBP,
     !                   Kode,P,Q,NumNP,Aqh,Bqh,GWL0L,qGWLF)
      logical qGWLF
      dimension Width(NumBP),KXB(NumBP),KODE(NumNP),P(NumNP),Q(NumNP)

      read (31,*) tAtm,Prec,rSoil,rRoot,hCritA,rGWL,GWL
      Prec=abs(Prec)
      rSoil=abs(rSoil)
      rRoot=abs(rRoot)
      hCritA=-abs(hCritA)
      hGWL=GWL+GWL0L
      rTop=rSoil-Prec
      do 10 i=1,NumBP
        n=KXB(i)
        K=Kode(n)
        if( K.eq.4 .or. K.eq.-4 ) then
          Kode(n)=-4
          Q(n)=-Width(i)*rTop
          goto 10
        endif
        if (K.eq. 3) P(n)=hGWL
        if (K.eq.-3) then
          if (qGWLF) then
            Q(n)=-Width(i)*Fqh(P(n)-GWL0L,Aqh,Bqh)
          else
            Q(n)=-Width(i)*rGWL
          endif
        endif
   10 continue
      return
      end

************************************************************************

      real function Fqh(GWL,Aqh,Bqh)
      Fqh=-Aqh*exp(Bqh*abs(GWL))
      return
      end

* !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
* Source file MATERIA2.FOR ::::::::::::::::::::::::::::::::::::::::::::::::::

      real function FK (h,Par)

      implicit real*8(A-H,O-Z)
      real*8 n,m,Ks,Kr,Kk
      real h,Par(9)

      BPar=.5
      PPar=2
      Hr=-1e20
      FFQr=1
      Qr=Par(1)
      Qs=Par(2)
      Qa=Par(3)
      Qm=Par(4)
      Alfa=Par(5)
      n=Par(6)
      Ks=Par(7)
      Kk=Par(8)
      Qk=Par(9)
      m=1-1/n
      C1Qee=1/(Qm-Qa)
      C2Qee=-Qa/(Qm-Qa)
      Qeer=C1Qee*Qr + C2Qee
      Qees=C1Qee*Qs + C2Qee
      Qeek=C1Qee*Qk + C2Qee
      if (Qa.lt.Qr) Hr=-1/Alfa*(Qeer**(-1/m)-1)**(1/n)
      Hs=dmin1( 0d0 ,  -1/Alfa*(Qees**(-1/m)-1)**(1/n) )
      Hk=dmin1( Hs  ,  -1/Alfa*(Qeek**(-1/m)-1)**(1/n) )
      if ( h.gt.Hr .and. h.le.Hk ) then
        Q=Qa+(Qm-Qa)*(1+(-Alfa*h)**n)**(-m)
        Qee=C1Qee*Q + C2Qee
        if (Qeer.gt.0)
!       FFQr=( 1 - Qeer**(1/m) )**m
        FFQ =( 1 - Qee **(1/m) )**m
        FFQk=( 1 - Qeek**(1/m) )**m
        C1Qe=1/(Qs-Qr)
        C2Qe=-Qr/(Qs-Qr)
        Qe = C1Qe*Q + C2Qe
        Qek= C1Qe*Qk+ C2Qe
        Kr=(Qe/Qek)**Bpar * ( (FFQr-FFQ)/(FFQr-FFQk) )**PPar * Kk/Ks
        FK=Ks*Kr
        return
      endif
      if ( h.gt.Hk .and. h.lt.Hs ) then
        Kr=(1-Kk/Ks)/(Hs-Hk)*(h-Hs)+1
        FK=Ks*Kr
      endif
      if ( h.ge.Hs ) FK=Ks
      if ( h.le.Hr ) FK=0.0
      return
      end

**************************************************************************

      real function FC (h,Par)

      implicit real*8(A-H,O-Z)
      real*8 n,m
```

```
      real h,Par(9)

      Hr=-1e20
      Qr=Par(1)
      Qs=Par(2)
      Qa=Par(3)
      Qm=Par(4)
      Alfa=Par(5)
      n=Par(6)
      m=1-1/n
      C1Qee=1/(Qm-Qa)
      C2Qee=-Qa/(Qm-Qa)
      Qeer=C1Qee*Qr + C2Qee
      Qees=C1Qee*Qs + C2Qee
      if (Qa.lt.Qr) Hr=-1/Alfa*(Qeer**(-1/m)-1)**(1/n)
      Hs=dmin1( 0d0 ,  -1/Alfa*(Qees**(-1/m)-1)**(1/n) )

      if ( h.gt.Hr .and. h.lt.Hs ) then
        C1=(1+(-Alfa*h)**n)**(-m-1)
        FC=(Qm-Qa)*m*n*(Alfa**n)*(-h)**(n-1)*C1
        return
      endif
      if (h.ge.Hs) FC=0.0
      if (h.le.Hr) FC=0.0
      return
      end

*************************************************************************

      real function FQ (h,Par)

      implicit real*8(A-H,O-Z)
      real*8 n,m
      real h,Par(9)

      Hr=-1e20
      Qr=Par(1)
      Qs=Par(2)
      Qa=Par(3)
      Qm=Par(4)
      Alfa=Par(5)
      n=Par(6)
      m=1-1/n
      C1Qee=1/(Qm-Qa)
      C2Qee=-Qa/(Qm-Qa)
      Qeer=C1Qee*Qr + C2Qee
      Qees=C1Qee*Qs + C2Qee
      if (Qa.lt.Qr) Hr=-1/Alfa*(Qeer**(-1/m)-1)**(1/n)
      Hs=dmin1( 0d0 ,  -1/Alfa*(Qees**(-1/m)-1)**(1/n) )

      if ( h.gt.Hr .and. h.lt.Hs ) then
        Qee=(1+(-Alfa*h)**n)**(-m)
        FQ=Qa+(Qm-Qa)*Qee
        return
      endif
      if (h.ge.Hs) FQ=Qs
      if (h.le.Hr) FQ=Qr
      return
      end
```

```
*************************************************************************
      real function FH (Qe,Par)

      implicit real*8(A-H,O-Z)
      real*8 n,m
      real Qe,Par(9)

      Qr=Par(1)
      Qs=Par(2)
      Qa=Par(3)
      Qm=Par(4)
      Alfa=Par(5)
      n=Par(6)
      m=1-1/n
      Q=Qr+(Qs-Qr)*Qe
      Qee=(Q-Qa)/(Qm-Qa)
      FH=-1/Alfa*(Qee**(-1/m)-1)**(1/n)
      return
      end

*************************************************************************
```

```
* Source file SINK2.FOR ||||||||||||||||||||||||||||||||||||||||||||||||||

      subroutine SetSnk (NumNP,NumNPD,NMatD,MatNum,hNew,TPot,Sink,
     !                   P0,POptm,P2H,P2L,P3,r2H,r2L,Beta,Length)
      real Length
      dimension MatNum(NumNPD),hNew(NumNPD),POptm(NMatD),
     !          Beta(NumNPD),Sink(NumNPD)

      do 10 i=1,NumNP
        if (Beta(i).gt.0.) then
          M=MatNum(i)
          Alfa=FAlfa(TPot,hNew(i),P0,POptm(M),P2H,P2L,P3,r2H,r2L)
          Sink(i)=Alfa*Beta(i)*Length*TPot
        endif
10    continue
      return
      end

************************************************************************

      real function FAlfa (TPot,h,P0,P1,P2H,P2L,P3,r2H,r2L)

      if (TPot.lt.r2L)  P2=P2L
      if (TPot.gt.r2H)  P2=P2H
      if ((TPot.ge.r2L).and.(TPot.le.r2H))
     !P2=P2H+(r2H-TPot)/(r2H-r2L)*(P2L-P2H)
      FAlfa=0.0
      if ((h.gt.P3).and.(h.lt.P2)) FAlfa=(h-P3)/(P2-P3)
      if ((h.ge.P2).and.(h.le.P1)) FAlfa=1.0
      if ((h.gt.P1).and.(h.lt.P0)) FAlfa=(h-P0)/(P1-P0)
      return
      end

* ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
```

```
* Source file OUTPUT2.FOR ||||||||||||||||||||||||||||||||||||||||||||||||

      subroutine TLInf (NumNP,NumBP,Kode,Q,P,CumQ,Width,SWidth,KXB,
     !                  t,dt,TLevel,ShortF,TPrint,Iter,ItCum,rTop,
     !                  rRoot,vMeanR,hMeanT,hMeanR,hMeanG,AtmInF,SinkF)
      integer TLevel
      logical ShortF,SinkF,AtmInF
      dimension Q(NumNP),Kode(NumNP),Width(NumBP),SWidth(4),KXB(NumBP),
     !          P(NumNP),CumQ(20),hMean(4),vMean(4)

      do 5 i=1,4
        vMean(i)=0
        hMean(i)=0
5     continue
      CumQ(11)=CumQ(11)+rTop  *dt
      CumQ(12)=CumQ(12)+rRoot *dt
      CumQ(13)=CumQ(13)+vMeanR*dt
      do 10 i=1,NumBP
        n=KXB(i)
        j=iabs(Kode(n))
        if (j.eq.0) goto 10
        hMean(j)=hMean(j)+P(n)*Width(i)/SWidth(j)
        vMean(j)=vmean(j)-Q(n)/SWidth(j)
10    continue
      do 20 j=1,4
        CumQ(j)=CumQ(j)+vMean(j)*dt
20    continue
      hMeanT=hMean(4)
      hMeanG=hMean(3)
      if (TLevel.eq.1) then
        write(70,104)
        write(71,105)
        write(77,106)
        write(78,107)
      endif
      if ( .not.ShortF .or. abs(TPrint-t).lt.0.001*dt ) then
        write(71,110) t,rTop,rRoot,vMean(4),vMeanR,vMean(3),vMean(1),
     !  vMean(2),TLevel
        write(77,120) t,hMean(4),hMeanR,hMean(3),hMean(1),hMean(2),
     !  TLevel
        write(78,130) t,CumQ(11),CumQ(12),CumQ(4),CumQ(13),CumQ(3),
     !  CumQ(1),CumQ(2),TLevel
        write(70,140) TLevel,t,dt,Iter,ItCum
      endif
      if (SinkF) then
        write(*,150) t,Iter,ItCum,CumQ(4),CumQ(13),CumQ(3),
     !  hMean(4),hMeanR,hMean(3)
      else
        if (AtmInF) then
          write(*,150) t,Iter,ItCum,CumQ(4),CumQ(1),CumQ(3),
     !    hMean(4),hMean(1),hMean(3)
        else
          write(*,150) t,Iter,ItCum,vMean(1),CumQ(1),CumQ(2),
     !    hMean(1),hMean(2)
        endif
      endif
104   format(//' TLevel   Time         dt       Iter  ItCum'/)
105   format(' All mean fluxes (v) are positive out of the region'//
     !'     Time        rAtm        rRoot       vAtm        vRoot   vKode3',
     !'       vKode1       vSeep       t-level'/
```

```
   !'      [T]      [L/T]       [L/T]     [L/T]       [L/T]    [L/T]',
   !'      [L/T]        [L/T]'/)
 106 format(//
   !'     Time         hAtm       hRoot     hKode3     hKode1 ',
   !'     hSeep      t-level'/
   !'      [T]          [L]         [L]       [L]         [L]  ',
   !'      [L]'/)
 107 format(' All cumulative mean specific amounts (CumQ) are',
   !' positive out of the region'//
   !'     Time     CumQAP      CumQRP     CumQA      CumQR     CumQ3',
   !'       CumQ1      CumQS      t-level'/
   !'      [T]        [L]         [L]       [L]         [L]      [L]',
   !'         [L]        [L]'/)
 110 format(F12.4,7E11.3,i8)
 120 format(F12.4,5F11.1,i8)
 130 format(F12.4,7E11.3,i8)
 140 format(I5,E12.3,E12.3,I5,I6)
 150 format(F12.4,I3,I6,1X,3E11.3,3F7.0)
     return
     end

***************************************************************************

      subroutine ALInf (t,CumQ,hMeanT,hMeanR,hMeanG,ALevel)

      integer ALevel
      dimension CumQ(20)

      if (ALevel.eq.1) write(72,100)
      write(72,110) t,CumQ(11),CumQ(12),CumQ(4),CumQ(13),CumQ(3),
     !hMeanT,hMeanR,hMeanG,ALevel
 100 format(' All cumulative mean specific amounts (CumQ) are',
     !' positive out of the region'//
     !'    Time       CumQAP      CumQRP     CumQA      CumQR     ',
     !'  CumQ3        hAtm        hRoot      hKode3     A-level'/
     !'    [T]          [L]         [L]        [L]        [L]       ',
     !'    [L]        [L]          [L]        [L] '/)
 110 format(F12.4,5E11.3,3f11.1,i8)
      return
      end

***************************************************************************

      subroutine hOut (hNew,X,Y,NumNP,t,IJ)

      dimension hNew(NumNp),X(NumNP),Y(NumNP)

      write(75,50) t
      k=0
      m=1
      do 30 n=1,NumNP
        k=k+1
        if( k.eq.min0(IJ,10) .or. n.eq.NumNP ) then
          write(75,40) m,x(m),y(m),(hNew(j),j=m,n)
          m=n+1
          k=0
        endif
  30 continue
  40 format(I5,2F8.1,10F10.1)
  50 format(//' Time  ***',f12.4,' ***'//
```

```
     !         '    n     x(n)    z(n)        h(n)       h(n+1) ...'/)
      return
      end

************************************************************************

      subroutine thOut (NMatD,NumNP,t,IJ,Par,hNew,X,Y,MatNum,
     !                  thr,Axz,Dxz)

      dimension hNew(NumNp),X(NumNP),Y(NumNP),Par(20,NMatD),th(16),
     !          MatNum(NumNP),thr(NMatD),Axz(NumNP),Dxz(NumNP)

      write(76,50) t
      k=0
      m=1
      do 30 n=1,NumNP
        k=k+1
        Ma=MatNum(n)
        th(k)=thr(Ma)+Dxz(n)*( FQ(hNew(n)/Axz(n),Par(1,Ma))-thr(Ma) )
        if( k.eq.min0(IJ,10) .or. n.eq.NumNP ) then
          write(76,40) m,x(m),y(m),(th(j),j=1,k)
          m=n+1
          k=0
        endif
   30 continue
   40 format(I5,2F8.1,10F10.3)
   50 format(//' Time  ***',f12.4,' ***'//
     !         '    n     x(n)    z(n)       th(n)      th(n+1) ...'/)
      return
      end

************************************************************************

      subroutine QOut (Q,X,Y,NumNP,t,IJ)

      dimension Q(NumNP),X(NumNP),Y(NumNP)

      write(74,50) t
      k=0
      m=1
      do 30 n=1,NumNP
        k=k+1
        if( k.eq.min0(IJ,10) .or. n.eq.NumNP ) then
          write(74,40) m,x(m),y(m),(Q(j),j=m,n)
          m=n+1
          k=0
        endif
   30 continue
   40 format(I5,2F8.1,10E10.2)
   50 format(//' Time  ***',f12.4,' ***'//
     !         '    n     x(n)    z(n)        Q(n)       Q(n+1) ...'/)
      return
      end

************************************************************************
```

```
      subroutine FlxOut (NumNP,NumEl,NumElD,ConAxx,ConAzz,ConAxz,
     !                    P,X,Y,IJ,Con,KX,t)
      integer e
      dimension KX(NumElD,4),P(NumNP),Con(NumNP),X(NumNP),Y(NumNP),
     !          ConAxx(NumEl),ConAzz(NumEl),ConAxz(NumEl),
     !          vXBuff(10),vZBuff(10)

      write(81,100) t
      write(82,105) t
      iB=0
      do 40 e=1,NumEl
        iB=iB+1
        CAxx=ConAxx(e)
        CAzz=ConAzz(e)
        CAxz=ConAxz(e)
        NCorn=4
        if (KX(e,3).eq.KX(e,4)) NCorn=3
        if (iB.eq.1) then
          iEB=e
          XB=0.
          ZB=0.
          do 10 i=1,NCorn
            XB=XB+X( KX(e,i) )/NCorn
            ZB=ZB+Y( KX(e,i) )/NCorn
10        continue
        endif
        Vx=0.
        Vz=0.
        do 20 n=1,NCorn-2
          i=KX(e,1)
          j=KX(e,n+1)
          k=KX(e,n+2)
          vi=Y(j)-Y(k)
          vj=Y(k)-Y(i)
          vk=Y(i)-Y(j)
          wi=X(k)-X(j)
          wj=X(i)-X(k)
          wk=X(j)-X(i)
          CMean=(Con(i)+Con(j)+Con(k))/3
          Area=.5*(wk*vj-wj*vk)
          A=1/Area/2
          AI=CAxx*vi+CAxz*wi
          AJ=CAxx*vj+CAxz*wj
          AK=CAxx*vk+CAxz*wk
          Vx=Vx-CMean*(A*(AI*P(i)+AJ*P(j)+AK*P(k))+CAxz)
          AI=CAxz*vi+CAzz*wi
          AJ=CAxz*vj+CAzz*wj
          AK=CAxz*vk+CAzz*wk
          Vz=Vz-CMean*(A*(AI*P(i)+AJ*P(j)+AK*P(k))+CAzz)
20      continue
        vXBuff(iB)=Vx/(NCorn-2)
        vZBuff(iB)=Vz/(NCorn-2)
        if ( iB.eq.min0((IJ-1),10) .or. e.eq.NumEl ) then
          write(81,110) iEB,xB,zB,(vZBuff(n),n=1,iB)
          write(82,110) iEB,xB,zB,(vXBuff(n),n=1,iB)
          iB=0
        endif
40    continue
100   format(//' Time  ***',f12.4,' ***'//
     !       '     e    x(e)  z(e)      vz(e)      vz(e+1) ...'/)
```

```
105 format(//' Time  ***',f12.4,' ***'//
     !        '    e    x(e)  z(e)     vx(e)     vx(e+1) ...'/)
110 format(I5,2F8.1,10E10.2)
      return
      end

************************************************************************

      subroutine SubReg (NumEl,NumElD,NumNP,NMatD,hNew,hOld,x,y,MatNum,
     !                   LayNum,Par,KX,Kat,t,dt,NLay,PLevel,thr,Axz,Dxz)
      integer PLevel
      dimension hNew(NumNP),hOld(NumNP),x(NumNP),y(NumNP),MatNum(NumNP),
     !          LayNum(NumEl),Par(20,NMatD),KX(NumElD,4),
     !          SubVol(10),SubCha(10),hMean(10),Area(10),
     !          thr(NMatD),Axz(NumNP),Dxz(NumNP)
      xMul=1
      Volume=0
      Change=0
      ATot=0
      do 190 i=1,NLay
        SubVol(i)=0
        SubCha(i)=0
        hMean(i)=0
        Area(i)=0
190 continue
      do 200 N=1,NumEl
        Lay=LayNum(N)
        NUS=4
        IF (KX(N,3) .EQ. KX(N,4)) NUS=3
        do 120 K=1,NUS-2
          I=KX(N,1)
          J=KX(N,K+1)
          L=KX(N,K+2)
          MI=MatNum(I)
          MJ=MatNum(J)
          MK=MatNum(L)
          CJ=X(I)-X(L)
          CK=X(J)-X(I)
          BJ=Y(L)-Y(I)
          BK=Y(I)-Y(J)
          if (KAT.eq.1) xMul=2*3.1416*(X(I)+X(J)+X(L))/3
          AE=xMul*(CK*BJ-CJ*BK)/2
          hE=(hNew(I)+hNew(J)+hNew(L))/3
          thI=thr(MI)+(FQ(hNew(I)/Axz(I),Par(1,MI))-thr(MI))*Dxz(I)
          thJ=thr(MJ)+(FQ(hNew(J)/Axz(J),Par(1,MJ))-thr(MJ))*Dxz(J)
          thK=thr(MK)+(FQ(hNew(L)/Axz(L),Par(1,MK))-thr(MK))*Dxz(L)
          VNewE=AE*(thI+thJ+thK)/3
          hE=(hOld(I)+hOld(J)+hOld(L))/3
          thI=thr(MI)+(FQ(hOld(I)/Axz(I),Par(1,MI))-thr(MI))*Dxz(I)
          thJ=thr(MJ)+(FQ(hOld(J)/Axz(J),Par(1,MJ))-thr(MJ))*Dxz(J)
          thK=thr(MK)+(FQ(hOld(L)/Axz(L),Par(1,MK))-thr(MK))*Dxz(L)
          VOldE=AE*(thI+thJ+thK)/3
          Volume=Volume+VNewE
          Change=Change+(VNewE-VOldE)/dt
          SubVol(Lay)=SubVol(Lay)+VNewE
          SubCha(Lay)=SubCha(Lay)+(VNewE-VOldE)/dt
          hMean(Lay)=hMean(Lay)+hE*AE
          Area(Lay)=Area(Lay)+AE
120   continue
200 continue
```

```
      do 210 Lay=1,NLay
        hMean(Lay)=hMean(Lay)/Area(Lay)
        ATot=ATot+Area(Lay)
210   continue
      if (PLevel.eq.1) write(80,315)
      write(80,320) t,      (        i,i=1,NLay)
      write(80,325) ATot,  (  Area(i),i=1,NLay)
      write(80,330) Volume,(SubVol(i),i=1,NLay)
      write(80,340) Change,(SubCha(i),i=1,NLay)
      write(80,350)        ( hMean(i),i=1,NLay)
315   format(/' Time [T]         Total     Sub-region number ...')
320   format(' f12.4,         12x,  10(I7,4x))
325   format( ' Area   [V]   ',e11.3,10e11.3)
330   format( ' Volume [V]   ',e11.3,10e11.3)
340   format( ' InFlow [V/T] ',e11.3,10e11.3)
350   format( ' hMean  [L]   ',11x,  10f11.1)
      return
      end

************************************************************************

      subroutine BouOut (NumNP,NumBP,NMatD,t,hNew,Q,Width,Par,KXB,Kode,
     !                   MatNum,X,Y,thr,Axz,Dxz)

      dimension hNew(NumNP),Q(NumNP),Width(NumBP),Par(20,NMatD),
     !          KXB(NumBP),Kode(NumNP),MatNum(NumNP),x(NumNP),y(NumNP),
     !          thr(NMatD),Axz(NumNP),Dxz(NumNP)

      write(79,100) t
      do 10 i=1,NumBP
        n=KXB(i)
        Ma=MatNum(n)
        th=thr(Ma)+Dxz(n)*( FQ(hNew(n),Par(1,Ma)) - thr(Ma) )
        v=-Q(n)/Width(i)
        write(79,110) i,n,x(n),y(n),Kode(n),Q(n),v,hNew(n),th
10    continue
100   format(//' Time:',f12.4//
     !'    i    n     x       z    Code      Q           v       ',
     !'    h        th'/
     !'                                    [V/T]       [L/T]'/)
110   format(2i5,2f7.1,i5,2e11.3,f11.1,f7.3)
      return
      end

* ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
```

REFERENCES

Belmans, C., J.G. Wesseling and R.A. Feddes, Simulation model of the water balance of a cropped soil: SWATRE, J.Hydrol., 63, 271-286, 1983.

Cislerova, M., Comparison of simulated water balance for ordinary and scaled soil hydraulic characteristics, Publication 82, Dept. of Hydraulics and Catchment Hydrology, Agric. Univ., Wageningen, The Netherlands, 1987.

Davis, L.A., and S.P. Neuman, Documentation and User's Guide: UNSAT2 - Variably Saturated Flow Model, Final Report, WWL/TM-1791-1, Water, Waste & Land, Inc., Ft. Collins, Colorado 80526, 1983.

Feddes, R.A., E. Bresler, and S.P. Neuman, Field Test of a Modified Numerical Model for Water Uptake by Root Systems, Water Resour. Res., 10(6), 1199-1206, 1974.

Feddes, R.A., P.J. Kowalik, and H. Zaradny, Simulation of field water use and crop yield, Simulation Monographs, Pudoc, Wageningen, 1978.

Genuchten, M.Th.van, A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, Soil Sci. Soc. Am. J., 44, 892-898, 1980.

Hopmans, J.W., and J.N.M. Stricker, Application and evaluation of techniques which describe the spatial variation of soil-physical and hydrological variables: a final report, 1987.

Mls, J., Formulace a reseni zakladnich uloh pro vertikalni infiltraci, Vodohosp. Cas., 30, 304-313, (in Czech), 1982.

Mualem, Y., A New Model for Predicting the Hydraulic Conductivity of Unsaturated Porous Media, Water Resour. Res., 12(3), 513-522, 1976.

Neuman, S.P., Saturated - Unsaturated Seepage by Finite Elements, Proc. ASCE, J. Hydraul. Div., 99 (HY12), 2233-2250, 1973.

Neuman, S.P., R.A. Feddes, and E. Bresler, Finite Element Simulation of Flow in Saturated - Unsaturated Soils Considering Water Uptake by Plants, Development of Metods, Tools and Solutions for Unsaturated Flow, Third Annual Report, Technion, Haifa, Israel, 1974.

Neuman, S.P., Galerkin Approach to Saturated-Unsaturated Flow in Porous Media, Ch. 10 in Finite Elements in Fluids, Vol. I: Viscous Flow and Hydrodynamics, edited by R.H. Gallagher, J.T. Oden, C. Taylor, and O.C. Zienkiewicz, John Wiley and Sons, London, 201-217, 1975.

Skaggs, R.W., Monke, E.J., and Huggins, L.F., An Approximate Method for Determining the Hydraulic Conductivity Function of an Unsaturated Flow, Technical Report No. 11, Water Resources Research Center, Purdue University, Lafayette, IN, 1970.

Vogel, T., M. Cislerova, On the Reliability of Unsaturated Hydraulic Conductivity Calculated from the Moisture Retention Curve, Transport in Porous Media, 3, 1-15, 1988.