 OTA 1392 I

november 1982

Institute for Land and Water Management Research (ICW)
P.O. Box 35, 6700AA Wageningen, The Netherlands

# ASPECTEN van INFORMATIEVERWERKING

**35**

## GUIDE TO THE PROGRAM UNSLOW FOR THE CALCULATION OF STEADY STATE INFILTRATION AND CAPILLARY RISE

ing. G.W. Bloemen

ing. J.B.H.M. van Gils

# ASPECTEN van INFORMATIEVERWERKING

## 35

De nota's handelende over Aspecten van Informatieverwerking bevatten inlichtingen over de ontwikkeling van de informatieverwerking binnen het instituut. Naast meer concluderende en toelichtende beschouwingen wordt aandacht besteed aan het gebruik van programma's, programma-pakketten en apparatuur. Tevens worden inlichtingen gegeven over praktijkervaring met en toepassing van informatieverwerking

C O N T E N T S

## 1. INTRODUCTION

The program UNSLOW calculates:

Suction profiles above a specified groundwater table depth for steady state up- and downward fluxes in soil profiles with a maximum of 100 layers and with a maximum of 50 suction values.

## 2. INFORMATION PREPARATORY TO INSTRUCTIONS FOR USE

### 2.1. Basic formulas

Calculations with the program UNSLOW are based on two formulas:

- the modified formula of Brooks and Corey for capillary conductivity k
- Darcys law for unsaturated flow

The modified formula of Brooks and Corey (BLOEMEN, 1980a) is

$$k = 0.5\, k_s \left(\frac{h_a}{r.h}\right)^{n_s} \qquad (cm.d^{-1}) \qquad (1)$$

where

$k_s$ = vertical saturated conductivity $(cm.d^{-1})$

$h_a$ = suction (cm) up to which 0.5 $k_s$ is maintained during desorption from saturation

r = factor to convert $h_a$ into the suction at which incomplete saturation after rewetting of the dry soil is attained, giving a maximum vertical saturated conductivity of 0.5 $k_s$

$n_s$ = the ratio $\frac{d(\log k)}{d(\log h)}$ of the k(h) relationship which holds as an average between drying and rewetting conditions. This ratio can be calculated as:

$$n_s = \log\left[2\left(\frac{h_a}{h_o}\right)^{n_d}\right] / \log\left(\frac{h_a}{r.h_o}\right) \qquad (2)$$

where

$h_o$ = suction (cm) where hydraulic conductivity becomes negligibly small

$n_d$ = the ratio $\frac{d(\log k)}{d(\log h)}$ of the k(h) relationship which holds for desorption after saturation

Darcy's formula reads:

$$v = k\left(\frac{dh}{dz} - 1\right) \qquad (cm.d^{-1}) \qquad (3)$$

where

v = flux in $cm.d^{-1}$

h = suction in cm (positive in unsaturated soil)

z = height in cm above the groundwater table (positive in upward direction)

## 2.2. Calculation of relationships between height z and suction h

The substitution of eq. (2) in eq. (1) followed by an integration with respect to h gives steady state flux equations. These allow the calculation of the relationship between the height z and suction h when steady state fluxes v occur (BLOEMEN, 1980b). In the suction range where k = 0.5 $k_s$ the equation is:

$$z = \frac{0.5\, k_s \,.\, h}{v + 0.5\, k_s} \qquad (cm) \qquad \text{for } h < \frac{h_a}{r} \qquad (4)$$

In the suction range where $k < 0.5\ k_s$ the equation reads as:

$$z_2 = z_1 + \frac{k(\frac{h_1+h_2}{2}).(h_2-h_1)}{v + k(\frac{h_1+h_2}{2})} \quad \text{(cm)} \quad \text{for} \quad h > \frac{h_a}{r} \qquad (5)$$

The program UNSLOW calculates the height z with eq. (4) and eq. (5). The value of k in eq. (5) with increasing h-values is calculated with eq. (1). Therefore the value of $0.5\ k_s$, $h_a/r$ and $n_s$ of the various layers of the soil profile must be known. It is not of importance in what way these constants have been determined. The possibilities for use of the model are largely increased however when these constants are easily evaluated from texture and organic matter content in case of mineral soils and from dry bulk density in case of peat soils.

## 2.3. Constants per layer

The constants which are specified in section 2.2 must be determined for each single layer in the soil profile. The constants $k_s$, $n_a$ and $n_d$ can be evaluated from organic matter content and two textural characteristics of a mineral soil layer:

- the median grainsize Md, which is the diameter that has half of the mineral parts by weight finer and half coarser. It can easily be read in a diagram plotting cumulative weight percentages against grain size;
- a dimensionless index f for grain size distribution. This index is defined as the weighted mean of the slope of the cumulative curve of grain size distribution. For the calculation of f a complete granular analysis with a sufficing number of grain size intervals is needed.

NOTE: The theory of the following sections 2.4 - 2.10 is now introduced in order to make the calculation by a handoperated calculator possible of the constants $0.5\ k_s$, $h_a/r$, and $n_s$ of eq.(1). In section 4.1 an example of this procedure is given.

## 2.4. Calculation of grain size distribution index f

An example of the calculation of the index f is given below. It refers to layer no. 1 of Table 1.

$i$ = consecutive numbers of grain size limits

$j = i + 1$

$S_i$ = grain size limits

$W$ = weight

$P_i$ = cumulative weight percentages

$tg_i$ = slope of cumulative curve between two size limits:

$$tg_i = \frac{\log(P_j/P_i)}{\log(S_j/S_i)}$$

$f_i$ = slope of cumulative curve between two size limits in proportion to weight percentage: $f_i = W \cdot tg_i$

Table 1. Example of calculation of index f

| 1 | $S_i$ (µm) | $W_i$ % | $P_i$ | $\log \frac{S_j}{S_i}$ | $\log \frac{P_j}{P_i}$ | $tg_i$ | $f_i$ |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 6.6 | 6.6 | | | | |
| 2 | 16 | 2.7 | 9.3 | 0.903 | 0.149 | 0.165 | 0.446 |
| 3 | 50 | 13.9 | 23.2 | 0.485 | 0.397 | 0.802 | 11.148 |
| 4 | 75 | 37.5 | 60.7 | 0.176 | 0.418 | 2,375 | 89.063 |
| 5 | 105 | 34.9 | 95.6 | 0.146 | 0.187 | 1.349 | 47.091 |
| 6 | 150 | 3.8 | 99.4 | 0.155 | 0.017 | 0.110 | 0.417 |
| 7 | 210 | 0.5 | 99.9 | 0.146 | 0.002 | 0.014 | 0.007 |
| 8 | 300 | 0.1 | 100.0 | 0.155 | 0.000 | 0.000 | 0.000 |

$$f = \frac{\sum_{i=2}^{n} f_i}{\sum_{i=2}^{n} W_i} = \frac{148.172}{93.4} = 1.586 \qquad (n = 8)$$

## 2.5. Constants for peat

The constants $k_s$, $h_a$ and $n_d$ for layers of peat can be evaluated from dry bulk density $\rho_b$ (BLOEMEN, 1983). This is the mass of dry material per unit wet bulk volume. It can be determined as ovendry weight at 105 degrees Celsius of a core of undisturbed saturated peat.

## 2.6. Calculation of vertical saturated conductivity

Vertical saturated conductivity $k_s$ can now be calculated for mineral soils with:

$$k_s = 0.02\ Md^{1.93} \cdot f^{-0.74} \quad (\text{cm.d}^{-1}) \qquad (6)$$

for fen peat with:

$$k_s = 0.00266\ \rho_b^{-3.625} \quad (\text{cm.d}^{-1}) \qquad (7)$$

for high bog peat with:

$$k_s = 0.0036\ \rho_b^{-2.83} \quad (\text{cm.d}^{-1}) \qquad (8)$$

## 2.7. Calculation of $h_a$

Suction $h_a$ can now be calculated for mineral soils with:

$$h_a = 2914\ Md^{-0.96}\ f^{0.79} \quad (\text{cm}) \qquad (9)$$

for fen peat with:

$$h_a = 416\ \rho_b^{1.12} \quad (\text{cm}) \qquad (10)$$

for high bog peat with:

$$h_a = 794\ \rho_b^{1.17} \quad (\text{cm}) \qquad (11)$$

## 2.8. Calculation of $n_d$

The slope factor $n_d$ can now be calculated for mineral soils with:

$$n_d = 1.4 + 4.536\,(e^{0.3f} - 1) - 0.75\, f^{1.6} \,.\, (\log H) \qquad (12)$$

(with H being the percentage of organic matter),

for fen peat with:

$$n_d = 2.54 - 2.42\,\rho_b \qquad (13)$$

for high bog peat with:

$$n_d = 2.57 - 2.27\,\rho_b \qquad (14)$$

## 2.9. Value of reduction factor r

The factor r for the reduction of $h_a$ has a value of

4.5 for sandy soils
2.9 for clay soils
3.1 for fen peat
1.9 for high bog peat($\rho_b$ < 0.1 $g.cm^{-3}$)
3.4 for high bog peat($\rho_b$ > 0.1 $g.cm^{-3}$)

## 2.10. Values of suction $h_0$

Suction $h_0$, where k-values are negligibly small, can be estimated on the basis of the values below:

| | | |
|---|---|---|
| $h_0$ = | 500 cm | for very coarse sand |
| $h_0$ = | 700 cm | for coarse sand and dune sand |
| $h_0$ = | 1000 cm | for moderately fine sand, marine sand, slightly loamy cover sand |
| $h_0$ = | 5000 cm | for very fine sand, loamy cover sand, peat with low dry bulk density ( < 0.2 $g.cm^{-3}$) |
| $h_0$ = | 7000 cm | for very loamy cover sand, light sandy clay |

$h_0 = 10^4$ cm for medium sandy clay, peat with high dry bulk density ( > 0.2 $g.cm^{-3}$)

$h_0 = 10^5$ cm for heavy sandy clay, loess loam, light clay

$h_0 = 10^6$ cm for heavy clay

## 2.11. Corrections for horizontal cracking

After $h_a/r$ and $n_s$ have been calculated, these constants must be corrected, if they apply to soil layers which form horizontal cracks when drying. This correction is as follows:

if h > 100 cm $n_s$ becomes ($n_s$ + 1.7) and the corrected value $h_a^*/_r$ is:

$$\frac{h_a^*}{r} = 100.(0.01.\frac{h_a}{r})^{\frac{n_s}{n_s + 1.7}} \quad \text{(cm)} \qquad (15)$$

These corrections are performed automatically in UNSLOW.

# 3. THE PROGRAM UNSLOW

## 3.1. Differences with the program CRISP

The differences between the program UNSLOW and the program CRISP as described by BLOEMEN (1980b) are:

- the program CRISP deals with upward flow only, UNSLOW deals with both up- and downward flow;
- integration steps are chosen differently. A constant number of steps is chosen between suction $h_a/r$ of the first layer and the next larger stated suction, and further between successive stated suctions;
- 0.5 $k_s$ values (from here called 'effective saturated conductivity') are given for each separate layer, consequently $h_a/r$ values (from here called 'water entry value') are not transformed;

- the effect of cracking on unsaturated flow is accounted for by means of transformation of water entry value and slope factor $n_s$ if suctions are > 100 cm.
- the following version of the FORTRAN program has been developed for a PDP-11 computer working with operating system RSX-11M PLUS.

## 3.2. Preliminary remarks

Per profile the calculated heights z for stated fluxes and stated suctions are printed in tables for a progression of groundwater-table depths.

The following fluxes ($cm.d^{-1}$) are stated:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.6 | 0.55 | 0.5 | 0.45 | 0.4 | 0.35 | 0.3 | 0.25 | 0.2 |
| 0.15 | 0.1 | 0.08 | 0.06 | 0.04 | 0.02 | 0.01 | 0.0 | -0.01 |
| -0.02 | -0.04 | -0.06 | -0.08 | -0.1 | -0.15 | -0.2 | -0.25 | -0.3 |
| -0.4 | -0.5 | | | | | | | |

The minimum flux and the maximum flux can be chosen.
A table with a maximum of 7 fluxes fits within 80 characters.

The following suctions (cm) are stated:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 |
| 125 | 150 | 175 | 200 | 500 | 750 | 1000 | 2000 | 5000 | 10000 |
| 16000 | | | | | | | | | |

The lowest suction in the table is the next larger value above $h_a/r$ of the layer with the groundwater level in it. The highest suction which is of interest in a specific case can be chosen freely.

The following depths of the groundwater table (cm below surface) are stated:

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 60 | 80 | 100 | 120 | 140 | 160 | 180 | 200 | 220 | 240 | 260 | 280 | 300 |

The minimum and the maximum depth of the groundwater table can be chosen freely.

The greatest possible height z in the tables is equal to the groundwater table depth. If the height z does not change with increasing suction, this height is indicated with "-".

If in program code the expression (SUCT(I)/SUCAV)**A(I) becomes less then 10**-12 or KI(K) becomes less then or equal to -V(I) the relative height z is restricted to groundwater depth or, when height z can exceed groundwater depth, z is replaced by "*".

The instruction input of program UNSLOW is the same as in program - package VAZAL which has been developed at the Institute for Land and Water Management Research (ICW). Only Dutch texts have been replaced by English texts.

## 3.3. Instructions for use of the program UNSLOW

When the system-prompt > appears on the screen the following instructions must be typed in, where "uic" is the users identification code:

| Type (write) | Action |
|---|---|
| @[uic]RUN<br>or<br>@[uic]BATCH | The possibilities of VAZAL are used |
| or | |
| RUN [uic]UNSLOW | The program "[uic]UNSLOW.TSK" then asks to state where the instructions come from. This can be from the terminal or from an instruction file.<br>The records of an instruction file contain the answers to the questions put by the program. |

These answers must be successively:

| for | record | answer (with comment) |
|---|---|---|
| files | | |
| | 0 | Identification of the instruction-file in maximally 35 characters<br><RET> = instructions from terminal<br>(It is the first answer required on the file "INITIEREN.INS" or, in case this file is missing, on the terminal or, in case of instructions of a stated file are already read on the last record to be read of this instruction file.) |

instructions for use: cont'd

| for | record | answer (with comment) |
|---|---|---|
| files | 1 | Y = request display of the instructions from file on the screen<br>N = no display<br>(This question is skipped when instructions come from the terminal.) |
| | 2 | Identification of the list file in max. 35 characters |
| profile | | |
| | 3 | Maximally 80 characters of a line with identification of the profile<br><RET> = Skip a line in the table<br>(As soon as input of 80 characters of a line is completed a question follows for the input of a maximum of 52 characters. With <RET> alone 80 characters are used.) |
| per layer | | |
| | 4 | Depth below surface in cm . . . . . . .(integer)<br><RET> = end of profile<br>(For layer 1 depth below surface = groundwater depth and this question is skipped. The layers must be entered from low to high.) |
| | 5 | Effective saturated conductivity in $cm.d^{-1}$ . . . (real)<br><RET> = end of profile |
| | 6 | Water entry value in cm . . . . . . . . . (real)<br><RET> = end of profile |
| | 7 | Slope factor . . . . . . . . . . . . . . .(real)<br><RET> = end of profile |
| | 8 | Y = this layer is a cracking soil type<br>N = this layer is not a cracking soil type<br>(For a next layer continue with record 4.) |

instructions for use: cont'd

| for | record | answer (with comment) |
|---|---|---|
| fluxes | | |
| | 9 | Minimum flux in $cm.d^{-1}$ . . . . . . . . . . . . (real)<br><RET> = lowest stated flux |
| | 10 | Maximum flux in $cm.d^{-1}$ . . . . . . . . . . . . (real)<br><RET> = highest stated flux |
| groundwater depths | | |
| | 11 | Minimum groundwater depth in cm below soil surface (integer)<br><RET> = lowest stated groundwater depth |
| | 12 | Maximum groundwater depth in cm below soil surface (integer)<br><RET> = highest stated groundwater depth<br>(If the choice is between two stated values one table for the chosen minimum groundwater depth is calculated.) |
| suctions | | |
| | 13 | Maximum suction in cm . . . . . . . . . . (integer)<br><RET> = highest stated suction |
| | 14 | Number of integration steps between 2 stated suctions<br><RET> = 30 steps |
| height z | | |
| | 15 | Y = height z can exceed groundwater level<br>N = height z cannot exceed groundwater level<br>(This question is not skipped as a consequence of a control answer.) |
| program control | | |
| | 16 | Y = the program is continued , else N |
| | 17 | Y = change the instruction file, record 0..1, " N |
| | 18 | Y = " the list file , " 2 , " N |
| | 19 | Y = " the profile , " 3..7, " N |
| | 20 | Y = " the fluxes , " 9..10, " N |
| | 21 | Y = " the groundwater depths, " 11..12, " N |
| | 22 | Y = " the suctions , " 13 , " N<br>(Unchanged instructions repeat the operation.) |

### 3.4. Program information

Subroutines:

| | |
|---|---|
| FLN | reads a filename from instruction input |
| FLNERR | treats file errors |
| FLNI | calls an instruction file |
| FLNSIO | calls a list file |
| FRMAT | makes a run-time format of a question |
| DECODR | decodes a real |
| READI | reads an integer from instruction input |
| READR | reads a real from instruction input |
| DECODI | decodes an integer |
| LINE | reads a record instruction input |
| STRING | reads a string characters instruction input |
| TIMDAY | writes time and date |
| YORNO | asks Yes or No |

Logical Unit Numbers:

LUN 1 is used for instruction input from file
" 4 " " " formatted output
" 5 " " " terminal

Compilation:

```
F4P UNSLOW/CK/NO14/NOSP=UNSLOW
```

Task building:

```
TKB @commandfile
```

Then the content of the command file is:

```
UNSLOW/FP=UNSLOW.OBJ,DRO:[1,1]F4POTS/LB
/
CLSTR=F4PCLS,FCSCLS:RO
MAXBUF=512
ACTFIL=3
UNITS=5
FMTBUF=140
//
```

(Compilation and task building:) or with command file FOR.CMD of VAZAL

**Magtape:**

The text of this guide inclusive the listing of the program is put on 9 track magnetic tape in one file with the command:

FLX MM:[50,6]/DO/FA=DR1:[50,6]UNSLOW.TXT

This file contains control sequences to print the text with a LA-120 printer on the auxiliary port of a DT80/1 terminal.

A copy of this tape is available on request.

**Variable names in the main program:**

| Theory | Name | Particular value | Comment |
|---|---|---|---|
| | ACCEPT | 5 | LUN of the terminal in-/output |
| | C | | scratch real variable |
| | DZ | | increase of Z if suction SUC increases |
| | DZOLD | | previous DZ value |
| | EOF | | = .TRUE. an end-of-file is read |
| | EXCEED | | = .TRUE. height z can exceed groundwater level |
| | F | | scratch real variable |
| | FNOT | -1.7E+38 | value of an unknown real |
| | G | | scratch real variable |
| | I | | scratch integer variable |
| | IDIM | 17 | core is reserved for 17 fluxes |
| | ILAYER | | current sequence value of a layer |
| | ISTEP | | current number of integrationsteps |
| | ISUC | | lowest sequence number of used suctions |
| | IV | | sequence number of the minimum flux |
| | IWATER | | lowest sequence number of used groundwater depths |
| | IZDIM | | maximum value (integer) of a height Z |
| | J | | scratch integer variable |
| | JDIM | 101 | core is reserved for 100 layers |
| | JLAYER | | current sequence value of a layer |
| | JWATER | | current sequence value of the groundwater depth |

Variable names in the main program, cont'd

| Theory | Name | Particular value | Comment |
|---|---|---|---|
| | K | | scratch integer variable |
| | KDIM | 50 | core is reserved for 50 installed suctions |
| | KSUC | | maximum suction |
| | KWATER | | current groundwater depth |
| | L | | scratch integer variable |
| | LASTIM | | last saved time |
| | LDIM | | number of characters in the table of a height z-value |
| | M | | scratch integer variable |
| | MSUC | | highest sequence number of used suctions |
| | MV | | sequence number of the maximum flux |
| | MWATER | | highest sequence number of used groundwater depths |
| | N | | scratch integer variable |
| | NCHAR | | number of characters in a line of the table |
| | NLAYER | | number of layers in the profile |
| | NNAME | | number of characters of the identification of the profile |
| | NSTEP | | number of steps between 2 stated suctions |
| | NSUC | | number of installed suctions |
| | NV | | number of used fluxes |
| | NWATER | 13 | number of installed groundwater depths |
| | ONCE | | = .TRUE. first time the height z can become greater than suction h |
| | REPORT | | = .TRUE. displays instruction input |
| | SLASH | 47 | the character slash (/) |
| | STEP | | integration step |
| | SUCAV | | SUCB + 0.5 STEP |

Variable names in the main program, cont'd

| Theory | Name | Particular value | Comment |
|---|---|---|---|
| | SUCB | | current SUC-value |
| | TRUE | .TRUE. | byte variable with the value .TRUE. |
| | UNITI | | LUN of the instruction input |
| | UNITO | | LUN of the instruction output |
| | UNIT4 | | LUN of the list file |
| | ZDIM | | maximum value (real) of a height z |
| | ZI | | current z value |
| $n_s$ | A | | slope factors of each layer |
| | CRACK | | = .TRUE. the layers are cracking soil types |
| | D | | string characters |
| | DEPTH | | depths below surface of the bottom of each layer |
| | FILEI | | name of the instruction file |
| | FILE4 | | name of the list file |
| | FMT | | run time format |
| | IGOTO | | control values |
| z | IZ | | heights z in cm (integers) |
| | IZMAX | | the number of the layer reached by flux v |
| | IZOLD | | IZ-values of the previous line in the table |
| | JSUC | | installed suctions |
| $0.5\ k_s$ | KE | | effective saturated conductivities of each layer |
| k | KI | | unsaturated conductivity |
| | NAME | | the identification of the profile |
| h | SUC | | suction values |
| $h_a/r$ | SUCT | | water entry values of each layer |
| v | V | | installed fluxes in cm/d |
| | WATER | | installed groundwater depths |
| z | Z | | heights z in cm (reals) |
| | ZMAX | | heights above groundwater level of the bottom of each layer |

## 4. AN EXAMPLE OF CALCULATION

### 4.1. Example of preparation of input for a layered soil profile

The input for UNSLOW for a soil profile has to be prepared down to the lowest groundwater level that is of interest. Per layer the constants 0.5 $k_s$, $h_a/r$ and $n_s$ are given, as well as the depth below soil surface where the layers begin, as seen from below.

The input starts with the lowest layer and its depth below surface is undefined.

In Table 2 relevant data of a marine soil profile are given as an example of the calculation of the input for the program, which is given in Table 3.

Table 2. Relevant data of a marine soil profile

| Soil-type | Depth cm | org. % | Md μm | 2μm % | 16μm % | 50μm % | 75μm % | 105μm % | 150μm % | 210μm % | f |
|---|---|---|---|---|---|---|---|---|---|---|---|
| sandy clay | 0–15 | 5.4 | 31 | 19.1 | 11.3 | 43.8 | 12.6 | 9.4 | 2.6 | 1.2 | 0.55 |
| heavy clay | 15–50 | 3.7 | 2 | 50.0 | 23.9 | 21.5 | 2.3 | 1.3 | 0.5 | 0.5 | 0.19 |
| peat | 50–75 | (dry bulk density $\rho_b$ = 0.24 $g.cm^{-3}$) | | | | | | | | | |
| clayey sand | 75–150 | 0.8 | 70 | 6.6 | 2.7 | 13.9 | 38.1 | 34.9 | 3.7 | 0.1 | 1.63 |
| sand | > 150 | 0.6 | 91 | 3.3 | 2.6 | 11.9 | 19.8 | 27.7 | 27.4 | 6.7 | 1.28 |

Table 3. Constants and input for UNSLOW of the marine soil profile of Table 2. The input is given between the heavy lines. The *-marked layer is cracking

| Depth | $k_s$ ($cm.d^{-1}$) | $h_a$ (cm) | $n_d$ | r | $h_o$ (cm) | 0.5 $k_s$ ($cm.d^{-1}$) | $h_a/r$ (cm) | $n_s$ | |
|---|---|---|---|---|---|---|---|---|---|
| undef. | 100.6 | 46 | 3.77 | 4.5 | 1000 | 50.3 | 10 | 2.37 | |
| 150 | 50.7 | 72 | 4.42 | 4.5 | 1000 | 25.4 | 16 | 2.64 | |
| 75 | 0.47 | 84 | 1.96 | 3.1 | 10000 | 0.24 | 27 | 1.47 | |
| 50 | 0.26 | 403 | 1.64 | 2.9 | 1000000 | 0.13 | 139 | 1.37 | * |
| 15 | 23.5 | 67 | 2.00 | 2.9 | 10000 | 11.75 | 23 | 1.53 | |

4.2. Instruction file

The following instructions in file UNSLOW.INS refer to the marine soil profile in Table 2:

```
Y            / instructions to be shown
UNSLOW.LST / filename of the data output
                                              marine soil profile of Table 2
50.3         / effective saturated conductivity of    layer 1
10           / water entry value
2.37         / slope factor
N            / this layer is not a cracking soil type
150          / depth below surface of the bottom of   layer 2
25.4         / effective saturated conductivity
16           / water entry value
2.64         / slope factor
N            / this layer is not a cracking soil type
75           / depth below surface of the bottom of   layer 3
0.24         / effective saturated conductivity
27           / water entry value
1.47         / slope factor
N            / this layer is not a cracking soil type
50           / depth below surface of the bottom of   layer 4
0.13         / effective saturated conductivity
139          / water entry value
1.37         / slope factor
Y            / this layer is     a cracking soil type
15           / depth below surface of the bottom of   layer 5
11.8         / effective saturated conductivity
23           / water entry value
1.53         / slope factor
N            / this layer is not a cracking soil type
             / depth below surface of the bottom      end of layers
-0.08        / minimum flux
0.1          / maximum flux
80           / minimum groundwater depth
160          / maximum groundwater depth
10000        / maximum suction
             / 30 steps between 2 stated suctions (default 30)
N            / height z cannot exceed groundwater level
N            / no continuation of program UNSLOW
```

### 4.3. Conversation on the screen

The followins conversation is the result of a run with prosram UNSLOW and instruction-file UNSLOW.INS in section 4.2.
( The first command after the prompt > is RUN [uic]UNSLOW )

```
>RUN [uic]UNSLOW

        Instructions from a file? [filename]: UNSLOW.INS
             instructions to be shown? [Y/N]: Y

                     data output? [filename]: UNSLOW.LST

Type the identification of the profile      - the  first 80 characters? [STRING]:
....+....+....+....+....+....+....+....+....+....+....+....+....+....+....+....+
                                           marine soil profile of Table 2
layer  1 with the groundwater in it:
     effective saturated conductivity? [real]= 50.3
                    water entry value? [real]= 10.0
                         slope factor? [real]= 2.37
    is this layer a cracking soil type? [Y/N]: N
layer  2
 depth below surface of its bottom? [integer]=      150
     effective saturated conductivity? [real]= 25.4
                    water entry value? [real]= 16.0
                         slope factor? [real]= 2.64
    is this layer a cracking soil type? [Y/N]: N
layer  3
 depth below surface of its bottom? [integer]=       75
     effective saturated conductivity? [real]= 0.240
                    water entry value? [real]= 27.0
                         slope factor? [real]= 1.47
    is this layer a cracking soil type? [Y/N]: N
layer  4
 depth below surface of its bottom? [integer]=       50
     effective saturated conductivity? [real]= 0.130
                    water entry value? [real]= 139.
                         slope factor? [real]= 1.37
    is this layer a cracking soil type? [Y/N]: Y
layer  5
 depth below surface of its bottom? [integer]=       15
     effective saturated conductivity? [real]= 11.8
                    water entry value? [real]= 23.0
                         slope factor? [real]= 1.53
    is this layer a cracking soil type? [Y/N]: N
layer  6
 depth below surface of its bottom? [integer]=  -32767

                                             minimum flux? [real]=-0.8E-01
                                             maximum flux? [real]= 0.100
                           minimum groundwater depth? [integer]=       80
                           maximum groundwater depth? [integer]=      160
                                     maximum suction? [integer]=    10000
how many steps between 2 stated suctions, <RET>=30? [integer]=       30
                  can height z exceed groundwater level? [Y/N]: N


 conversation, cont'd:

                                                                 10:16:12 19-NOV-82
                                                                 10:16:56 19-NOV-82
                                                                 10:18:00 19-NOV-82
                                                                 10:18:47 19-NOV-82
                                                                 10:19:22 19-NOV-82
                                                                 10:19:50 19-NOV-82


        Do you continue  program UNSLOW? [Y/N]: N
 TT15  --   STOP program UNSLOW
```

## 4.4. A table for a defined groundwater depth

10:19:23 19-NOV-8?

CALCULATION OF SUCTION PROFILES

marine soil profile of Table 2

ground water depth = 160 cm

| layer | height above groundwater (cm) | effective saturated conductivity (cm/d) | water entry value (cm) | slope factor | |
|---|---|---|---|---|---|
| 1 | 0 | 50.300 | 10.0 | 2.37 | |
| 2 | 10 | 25.400 | 16.0 | 2.64 | |
| 3 | 85 | 0.240 | 27.0 | 1.47 | |
| 4 | 110 | 0.130 | 139.0 | 1.37 | cracking soil type |
| 5 | 145 | 11.800 | 23.0 | 1.53 | |

number of steps between two stated suctions = 30

| suction h (cm) | flux v (cm/d) 0.100 | 0.080 | 0.060 | 0.040 | 0.020 | 0.010 | 0.000 | -0.010 | -0.020 | -0.040 | -0.060 | -0.080 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | height z (cm) | | | | | | | | | | | |
| 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 |
| 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 | 30 |
| 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 |
| 50 | 49 | 49 | 49 | 50 | 50 | 50 | 50 | 50 | 50 | 50 | 51 | 51 |
| 60 | 58 | 58 | 59 | 59 | 60 | 60 | 60 | 60 | 60 | 61 | 61 | 62 |
| 70 | 67 | 67 | 68 | 69 | 69 | 70 | 70 | 70 | 71 | 72 | 72 | 73 |
| 80 | 75 | 76 | 77 | 78 | 79 | 79 | 80 | 81 | 81 | 83 | 84 | 160 |
| 90 | 82 | 84 | 85 | 86 | 87 | 88 | 90 | 92 | 97 | 116 | 160 | - |
| 100 | 87 | 88 | 89 | 91 | 94 | 96 | 100 | 106 | 114 | 129 | - | - |
| 125 | 92 | 94 | 97 | 101 | 109 | 116 | 125 | 134 | 143 | 158 | - | - |
| 150 | 97 | 100 | 104 | 111 | 128 | 138 | 150 | 160 | 160 | 160 | - | - |
| 175 | 101 | 104 | 109 | 124 | 145 | 160 | 160 | - | - | - | - | - |
| 200 | 104 | 108 | 117 | 135 | 160 | - | - | - | - | - | - | - |
| 500 | 119 | 128 | 152 | 160 | - | - | - | - | - | - | - | - |
| 750 | 121 | 131 | 160 | - | - | - | - | - | - | - | - | - |
| 1000 | - | 132 | - | - | - | - | - | - | - | - | - | - |
| 2000 | 122 | - | - | - | - | - | - | - | - | - | - | - |
| 5000 | - | 133 | - | - | - | - | - | - | - | - | - | - |
| 10000 | - | - | - | - | - | - | - | - | - | - | - | - |

ACKNOWLEDGEMENT

The authors are indebted to Dr. R.A. Feddes and Dr. Ph.Th. Stol for critically reviewing this paper.

REFERENCES

BLOEMEN, G.W., 1980a. Calculation of hydraulic conductivities from texture and organic matter content. Zeitschrift für Pflanzenernährung und Bodenkunde. 143. Band, Heft 5, seite 581-605 (also Technical Bulletin 120,ICW)

_____ 1980b. Calculation of steady state capillary rise from the groundwater table in multi-layered soil profiles. Zeitschrift für Pflanzenernährung und Bodenkunde. 143. Band, Heft 6, seite 701-719 (also Technical Bulletin 121, ICW)

_____ 1983.. Calculation of hydraulic conductivities and steady state capillary rise in peat soils from bulk density and solid matter volume. Zeitschrift für Pflanzenernährung und Bodenkunde. (In press)

```
      PROGRAM UNSLOW
C                                                                  nov.'82
C
C Model for the calculation of suction profiles in multi-layered soil
C profiles with specified depths of groundwater table.
C
C This model was developed by G.W. Bloemen, Institute for Land and Water
C Management Research with the cooperation of J.B.H.M. van Gils of the
C same institute, who wrote the program in fortran.
C
      REAL*4 KE(101),SUCT(101),A(101),FNOT,V(29)
      REAL*8 C,DZ,SUCAV,SUCB,KI(101),Z(29),ZI,ZDIM,STEP,F,P
      BYTE NAME(132),IGOTO(6),REPORT,TRUE,FMT(140),EOF,D(132),MIN,
     *FILEI(35),FILE4(35),EXCEED,SLASH,CRACK(101)
      INTEGER IZMAX(29),UNITO,UNITI,ACCEPT,UNIT4,SUC(50),ZMAX(101),
     *WATER(13),JSUC(21),DEPTH(101),IZ(29),IZOLD(29)
      EQUIVALENCE(SUC(1),JSUC(1))
      DATA V/0.6,0.55,0.5,0.45,0.4,0.35,0.3,0.25,0.2,0.15,0.1,0.08,0.06,
     *0.04,0.02,0.01,0.0,-0.01,-0.02,-0.04,-0.06,-0.08,-0.1,-0.15,-0.2,
     *-0.25,-0.3,-0.4,-0.5/IDIM/29/
      DATA JSUC/10,20,30,40,50,60,70,80,90,100,125,150,175,200,500,750,
     *1000,2000,5000,10000,16000/NSUC/21/
      DATA WATER/60,80,100,120,140,160,180,200,220,240,260,280,300/
     *NWATER/13/
      DATA TRUE/.TRUE./JDIM/101/KDIM/50/SLASH/47/MIN/45/
C
C++++++++++++++++++++++++++++++++++++++++++++++++++++ format statements
C
  201 FORMAT(' '132A1)
  202 FORMAT(1H1)
  203 FORMAT('0'42X'CALCULATION OF SUCTION PROFILES'//
     *' ',<NNAME>A1/<K>(' ')'ground water depth = 'I5' cm')
  204 FORMAT(27(' ')'effective   water'/9(' ')'height above'6(' ')'satur
     *ated   entry    slope'/' layer   groundwater   conductivity   val
     *ue   factor'/17(' ')'(cm)'9(' ')'(cm/d)    (cm)'/' '52('-'))
  213 FORMAT(I6,I15,F15.3,F8.1,F9.2' cracking soil type')
  214 FORMAT(I6,I15,F15.3,F8.1,F9.2)
  205 FORMAT(<K>(' ')'                number of steps'/
     *        <K>(' ')'between two stated suctions ='I5)
  206 FORMAT('0'15(' '),<J>('-')/'0'<K>(' ')'flux v (cm/d)'/'0'11(' ')
     *,<NV>F<LDIM>.3/'0'15(' '),<J>('-')/'0 suction h (cm)'<L>(' ')'heig
     *ht z (cm)'/)
  207 FORMAT(' ')
  208 FORMAT('0Type the array suc-values in rising order and in integer
     *form.'/' Separate the values with comma or <RET> and close with /
     *'/' Only / saves all the last values, only 1 value saves the other
     * values.')
  209 FORMAT(' layer  1 with the groundwater in it!')
  210 FORMAT(' layer'I3)
C
  300 FORMAT(I8,2(' '),<NV>I<LDIM>)
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++ starting values
C
      ACCEPT=5
      UNITI=ACCEPT
      UNITO=ACCEPT
      UNIT4=UNITO
      DO 1 I=1,6
    1 IGOTO(I)=TRUE
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ files
C
    2 IF(UNITI.EQ.ACCEPT.OR.REPORT.EQ.TRUE) WRITE(UNITO,207)
      IF(IGOTO(1).EQ.TRUE) CALL FLNI(UNITI,ACCEPT,UNITO,REPORT,FMT,FILEI
     *)
      IF(IGOTO(2).EQ.TRUE) CALL FLNSIO(UNITI,ACCEPT,UNITO,REPORT,FMT,
     *FILE4,UNIT4,UNIT4,1)
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ profile
C
      IF(IGOTO(3).NE.TRUE) GOTO 8
      IF(UNITI.EQ.ACCEPT.OR.REPORT.EQ.TRUE) WRITE(UNITO,207)
      CALL LINE(UNITI,ACCEPT,UNITO,REPORT,NNAME,NAME,EOF,41,
     *'Type the identification of the profile    ')
      IF(NNAME.GT.0) GOTO 3
      NAME(1)=0
      NNAME=1
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ per layer
C
    3 NLAYER=0
      FNOT=-1.7E+38
      DEPTH(1)=32767
```

```
      4 NLAYER=NLAYER+1
        IF(UNITI.NE.ACCEPT.AND.REPORT.NE.TRUE) GOTO 5
        IF(NLAYER.EQ.1) WRITE(UNITO,209)
        IF(NLAYER.NE.1) WRITE(UNITO,210) NLAYER
C
      5 IF(NLAYER.EQ.1) GOTO 6
        CALL READI(UNITI,ACCEPT,UNITO,REPORT,FMT,-32767,
       *I,EOF,36,' depth below surface of its bottom')
        IF(I.EQ.-32767) GOTO 7
        DEPTH(NLAYER)=I
C
      6 CALL READR(UNITI,ACCEPT,UNITO,REPORT,FMT,FNOT,KE(NLAYER),EOF
       *,37,'      effective saturated conductivity')
        IF(KE(NLAYER).EQ.FNOT) GOTO 7
C
        CALL READR(UNITI,ACCEPT,UNITO,REPORT,FMT,FNOT,SUCT(NLAYER),EOF,37,
       *'                      water entry value')
        IF(SUCT(NLAYER).EQ.FNOT)   GOTO 7
C
        CALL READR(UNITI,ACCEPT,UNITO,REPORT,FMT,FNOT,A(NLAYER),EOF,37,
       *'                           slope factor')
C
        CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,CRACK(NLAYER),EOF,38,
       *'    is this layer a cracking soil type')
C
        IF(A(NLAYER).NE.FNOT.AND.NLAYER.LT.JDIM)      GOTO 4
      7 NLAYER=NLAYER-1
        IF(NLAYER.LE.0.OR.NLAYER.GT.JDIM) STOP 'error in layer'
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ fluxes
C
      8 IF(IGOTO(4).NE.TRUE) GOTO 10
        IF(UNITI.EQ.ACCEPT.OR.REPORT.EQ.TRUE) WRITE(UNITO,207)
        CALL READR(UNITI,ACCEPT,UNITO,REPORT,FMT,V(IDIM),G,EOF,53,
       *'                                            minimum flux')
        CALL READR(UNITI,ACCEPT,UNITO,REPORT,FMT,V(1),H,EOF,53,
       *'                                            maximum flux')
        IV=0
        MV=IDIM
        DO 9 I=1,IDIM
        IF(IV.EQ.0.AND.H.GE.V(I)) IV=I
        IF(G.LE.V(I)) MV=I
      9 CONTINUE
        NV=MV-IV+1
        LDIM=(133-10)/NV
        IF(LDIM.LT.6.OR.IV.LT.1) STOP 'wrong V-values'
        IF(LDIM.GT.10) LDIM=10
        ZDIM=32767
        IZDIM=32767
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++ groundwater depths
C
     10 IF(IGOTO(5).NE.TRUE) GOTO 13
        CALL READI(UNITI,ACCEPT,UNITO,REPORT,FMT,WATER( 1),J,EOF,50,
       *'                            minimum groundwater depth')
        CALL READI(UNITI,ACCEPT,UNITO,REPORT,FMT,J,K,EOF,50,
       *'                            maximum groundwater depth')
        IWATER=0
        MWATER=0
        DO 11 I=1,NWATER
        IF(WATER(I).LE.K) MWATER=I
        IF(IWATER.EQ.0.AND.WATER(I).GE.J) IWATER=I
     11 CONTINUE
        IF(IWATER.LT.MWATER) GOTO 12
        IWATER=WATER(IWATER)
        IF(J.EQ.K) IWATER=J
        MWATER=IWATER
     12 IF(IWATER.EQ.0.OR.MWATER.EQ.0) STOP 'wrong groundwater depths'
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ suctions
C
     13 IF(IGOTO(6).NE.TRUE) GOTO 14
        CALL READI(UNITI,ACCEPT,UNITO,REPORT,FMT,SUC(NSUC),KSUC,EOF,50,
       *'                                     maximum suction')
        CALL READI(UNITI,ACCEPT,UNITO,REPORT,FMT,30,NSTEP,EOF,50,
       *'how many steps between 2 stated suctions, <RET>=30')
C
C+++++++++++++++++++++++++++++++++++++++ height exceeds groundwater level
C
     14 CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,EXCEED,EOF,54,
       *'                  can height z exceed groundwater level')
```

```
C
C+++++++++++++++++++++++++++++++++++++++++++ table per groundwater depth
C
      LASTIM=-1
      DO 29 JWATER=IWATER,MWATER
      IF(IWATER.EQ.MWATER) KWATER=IWATER
      IF(IWATER.NE.MWATER) KWATER=(WATER(JWATER))
      IF(KWATER.LT.0) GOTO 29
      ILAYER=1
      DO 15 I=1,NLAYER
      J=KWATER-DEPTH(I)
      IF(J.GT.0) GOTO 15
      J=0
      ILAYER=I
      JLAYER=I+1
   15 ZMAX(I)=J
C
      MSUC=0
      ISUC=0
      DO 16 I=1,NSUC
      IF(ISUC.EQ.0.AND.SUCT(ILAYER).LT.SUC(I)) ISUC=I
      IF(MSUC.EQ.0.AND.KSUC.LE.SUC(I)) MSUC=I
   16 CONTINUE
      IF(MSUC-ISUC.LT.0.OR.ISUC.EQ.0.OR.MSUC.EQ.0)
     *STOP 'wrong SUC-values'
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++ head of table
C
      IF(UNIT4.NE.UNIT0) CALL TIMDAY(UNIT0,80,LASTIM)
      I=-1
      CALL TIMDAY(UNIT4,132,I)
      K=NV*LDIM-17
      IF(K.LT.50) K=50
      WRITE(UNIT4,203) (NAME(J),J=1,NNAME),KWATER
      WRITE(UNIT4,204)
      DO 51 I=ILAYER,NLAYER
      IF(CRACK(I).EQ.TRUE) WRITE(UNIT4,213) I,ZMAX(I),KE(I),SUCT(I),A(I)
      IF(CRACK(I).NE.TRUE) WRITE(UNIT4,214) I,ZMAX(I),KE(I),SUCT(I),A(I)
   51 CONTINUE
      K=K-6
      WRITE(UNIT4,205) NSTEP
      I=NV*LDIM
      J=I-3
      I=I-I/2
      K=I+6
      L=I-8
      IF(L.LT.2) L=2
      WRITE(UNIT4,206) (V(I),I=IV,MV)
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++ starting values
C
      SUCB=SUCT(ILAYER)
      STEP=(SUC(ISUC)-SUCB)/NSTEP
      NCHAR=10+NV*LDIM
      ENCODE(132,201,D) (SLASH,I=1,131)
      DO 17 I=IV,MV
      IZOLD(I)=-32767
      IZMAX(I)=ILAYER
   17 Z(I)=KE(ILAYER)*SUCB/(V(I)+KE(ILAYER))
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++ per SUC-step, per V
C
      DO 28 J=ISUC,MSUC
      ISTEP=0
      IF(STEP.LT.1.D-8) GOTO 23
   18 CONTINUE
      SUCAV=SUCB+STEP/2.D0
      CALL TIMDAY(UNIT0,80,LASTIM)
      DO 19 I=ILAYER,NLAYER
      F=SUCT(I)
      C=A(I)
      IF(CRACK(I).NE.TRUE.OR.SUCAV.LE.100.D0) GOTO 50
      F=0.01D0*F
      P=C
      C=C+1.7D0
      P=P/C
      P=P*DLOG(F)
      GOTO 1000
C
C F=DEXP(P) has been shifted to the end of the program to avoid the
C occurrence of an unexplained error.
C
 1001 CONTINUE
      F=100.D0*F
   50 IF(F.LE.0.D0.OR.C.LE.0.D0) GOTO 19
```

```
      C=(F/SUCAV)**C
      IF(C.LE.1.D-12) C=0.DO
      KI(I)=KE(I)*C
   19 CONTINUE
      DO 22 I=IV,MV
      ZI=Z(I)
      IF(ZI.GE.ZDIM) GOTO 22
      K=IZMAX(I)
      C=KI(K)
      DZ=ZDIM
      P=-V(I)+0.1D-10
      IF(C.NE.0.DO.AND.C.GT.P) DZ=STEP/(1.+V(I)/C)
      ZI=ZI+DZ
      C=ZMAX(2)
      IF(ZI.LE.C) IZMAX(I)=ILAYER
      DO 21 L=JLAYER,NLAYER
      C=ZMAX(L)
   21 IF(C.GT.0.DO.AND.ZI.GT.C) IZMAX(I)=L
   22 Z(I)=ZI
      SUCB=SUCB+STEP
      ISTEP=ISTEP+1
      IF(ISTEP.LT.NSTEP) GOTO 18
   23 IF(J.LT.NSUC) C=SUC(J+1)-SUC(J)
      STEP=C/NSTEP
C
C++++++++++++++++++++++++++++++++++++++++++++++ line for stated SUC-value
C
      DO 24 I=IV,MV
      ZI=Z(I)
      IF(ZI.GT.ZDIM) ZI=ZDIM
      K=ZI+0.5
      IF(EXCEED.NE.TRUE.AND.K.GT.KWATER) K=KWATER
      M=11+(I-IV+1)*LDIM-2
      IF(D(M).EQ.MIN.AND.(K.EQ.KWATER.OR.K.EQ.IZDIM)) IZOLD(I)=K
   24 IZ(I)=K
      K=SUCB+0.5
      ENCODE(NCHAR,300,D) K,(IZ(I),I=IV,MV)
C
      DO 27 I=IV,MV
      N=IZ(I)
      IF(N.NE.IZOLD(I).AND.N.NE.IZDIM) GOTO 27
C
      M=11+(I-IV)*LDIM
      K=M+LDIM-1
      DO 26 L=M,K
   26 D(L)=32
      M=K-1
      D(M)=45
      IF(N.EQ.IZDIM) D(M)=42
C
   27 IZOLD(I)=N
      WRITE(UNIT4,201) (D(I),I=1,NCHAR)
   28 CONTINUE
C
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ next table
C
      WRITE(UNIT4,202)
      LASTIM=-1
      IF(UNIT4.NE.UNITO) CALL TIMDAY(UNITO,80,LASTIM)
   29 CONTINUE
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ control
C
      IF(UNITI.EQ.ACCEPT.OR.REPORT.EQ.TRUE) WRITE(UNITO,207)
      CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,IGOTO(1),EOF,36,
     *'      Do you continue  program UNSLOW')
      IF(IGOTO(1).NE.TRUE) STOP 'program UNSLOW'
      CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,IGOTO(1),EOF,36,
     *'       with an other instruction-file')
      CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,IGOTO(2),EOF,36,
     *'       with an other           listfile')
      CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,IGOTO(3),EOF,36,
     *'       with an other            profile')
      CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,IGOTO(4),EOF,36,
     *'       with    other            fluxes')
      CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,IGOTO(5),EOF,36,
     *'       with  other groundwater depths')
      CALL YORNO(UNITI,ACCEPT,UNITO,REPORT,D,IGOTO(6),EOF,36,
     *'       with other              suctions')
      GOTO 2
C
 1000 CONTINUE
      F=DEXP(P)
      GOTO 1001
      END
```

```
      SUBROUTINE FLN(UNITI,ACCEPT,UNITO,REPORT,FMT,IUNIT,FILE,N,TEXT)
C                                                            december '81
C  Reads a filename from instruction input.
C
      BYTE TEXT(N),FILE(35),BLANK,REPORT,TRUE,NULL,FMT(140),A(14)
      INTEGER UNITI,ACCEPT,UNITO,DIM
      DATA DIM/35/
      DATA TRUE/.TRUE./BLANK/' '/NULL/0/
      DATA A( 1)/ 63/A( 2)/ 32/A( 3)/ 91/A( 4)/102/A( 5)/105/A( 6)/108/
     *     A( 7)/101/A( 8)/110/A( 9)/ 97/A(10)/109/A(11)/101/A(12)/ 93/
     *     A(13)/ 58/A(14)/ 32/
  100 FORMAT(35A1)
  201 FORMAT('+'35A1)
  202 FORMAT(' Error in filename')
C
      CALL ERRSET(29,,.FALSE.,,.TRUE.)
      CALL ERRSET(30,,.FALSE.,,.FALSE.)
      CALL ERRSET(37,,.FALSE.)
      CALL ERRSET(43,,.FALSE.)
    1 DO 2 I=1,DIM
    2 FILE(I)=NULL
      IF(UNITI.NE.ACCEPT.AND.REPORT.NE.TRUE) GOTO 3
      CALL FRMAT(140,FMT,N,TEXT,14,A)
      WRITE(UNITO,FMT)
    3 READ(UNITI,100,ERR=998,END=999) (FILE(I),I=1,DIM)
      IF(REPORT.EQ.TRUE.AND.UNITI.NE.ACCEPT) WRITE(UNITO,201) (FILE(I),I
     *=1,DIM)
      DO 4 I=1,DIM
      IF(FILE(I).EQ.BLANK) FILE(I)=NULL
    4 CONTINUE
      IF(IUNIT.NE.ACCEPT.AND.IUNIT.NE.UNITO) CLOSE (UNIT=IUNIT)
      RETURN
  998 IF(UNITI.NE.ACCEPT) STOP 'error in filename'
      WRITE(UNITO,202)
      GOTO 1
  999 STOP 'EOF instead of filename'
      END
```

```
      SUBROUTINE FLNERR(UNITI,ACCEPT,IUNIT,IGOTO)
C                                                            december '81
C  Treats file errors.
C
      INTEGER ACCEPT,UNITI
    1 IGOTO=3
      CALL ERRSNS(IERR)
      IF(IERR.NE.30) GOTO 2
      IF(IOPEN.NE.0) STOP 'open failure, error 30'
      IOPEN=1
      IGOTO=2
      CLOSE(UNIT=IUNIT)
      IUNIT=ACCEPT
      RETURN
    2 IOPEN=0
      IF(IERR.NE.29.AND.IERR.NE.37.AND.IERR.NE.43) RETURN
      IGOTO=1
      IF(UNITI.NE.ACCEPT) STOP 'file errors'
      RETURN
      END
```

```
      SUBROUTINE FLNI(UNITI,ACCEPT,UNITO,REPORT,FMT,FILE)
C                                                               december '81
C  Calls an instruction file
C
      BYTE FILE(35),NULL,REPORT,TRUE,EOF,FMT(140)
      INTEGER UNITI,ACCEPT,UNITO
      DATA NULL/0/TRUE/.TRUE./
  200 FORMAT(' ')
      IF(UNITI.NE.ACCEPT) GOTO 1
      CALL ERRSET(29,,.FALSE.,,,.FALSE.)
      OPEN(UNIT=1,NAME='INITIEREN.INS',TYPE='OLD',ACCESS='SEQUENTIAL',
     *FORM='FORMATTED',ERR=1,READONLY,RECORDSIZE=132)
      UNITI=1
    1 CALL FLN(UNITI,ACCEPT,UNITO,TRUE,FMT,UNITI,FILE,31,
     *'        Instructions from a file')
      CALL ERRSNS
      IOLD=UNITI
      UNITI=ACCEPT
      IF(FILE(1).NE.NULL) UNITI=1
      IF(UNITI.EQ.ACCEPT.OR.UNITI.EQ.UNITO) GOTO 4
    2 OPEN(UNIT=UNITI,NAME=FILE,TYPE='OLD',ACCESS='SEQUENTIAL',FORM='FOR
     *MATTED',ERR=998,READONLY,RECORDSIZE=132)
  998 CALL FLNERR(IOLD,ACCEPT,IOLD,IGOTO)
      IF(IGOTO.EQ.1) UNITI=ACCEPT
      GOTO (1,2,3),IGOTO
    3 CALL YORNO(UNITI,ACCEPT,UNITO,TRUE,FMT,REPORT,EOF,36,
     *'          instructions to be shown')
    4 IF(UNITI.NE.ACCEPT.AND.REPORT.NE.TRUE) RETURN
      WRITE(UNITO,200)
      RETURN
      END



      SUBROUTINE FLNSID(UNITI,ACCEPT,UNITO,REPORT,FMT,FILE,IUNIT,JUNIT,
     *IORO)
C                                                               december '81
C  Calls a sequential file.
C
      BYTE FILE(35),NULL,REPORT,FMT(140)
      INTEGER UNITI,ACCEPT,UNITO
      DATA NULL/0/
    1 IF(IORO.NE.0) GOTO 3
      CALL FLN(UNITI,ACCEPT,UNITO,REPORT,FMT,IUNIT,FILE,31,
     *'                      data input')
      IUNIT=ACCEPT
      IF(FILE(1).NE.NULL) IUNIT=3
    2 IF(IUNIT.EQ.3) OPEN(UNIT=IUNIT,NAME=FILE,READONLY,
     *TYPE='OLD',ACCESS='SEQUENTIAL',FORM='FORMATTED',ERR=998,
     *RECORDSIZE=1792)
  998 CALL FLNERR(UNITI,ACCEPT,JUNIT,IGOTO)
      IF(IGOTO.EQ.1) IUNIT=ACCEPT
      GOTO (1,2,5),IGOTO
    3 CALL FLN(UNITI,ACCEPT,UNITO,REPORT,FMT,JUNIT,FILE,31,
     *'                     data output')
      JUNIT=UNITO
      IF(FILE(1).NE.NULL) JUNIT=4
    4 IF(JUNIT.EQ.4) OPEN(UNIT=JUNIT,NAME=FILE,
     * TYPE='NEW',ACCESS='SEQUENTIAL',FORM='FORMATTED',ERR=999,
     * RECORDSIZE=512)
      IF(JUNIT.NE.ACCEPT) GOTO 999
      CLOSE(UNIT=JUNIT)
      OPEN(UNIT=JUNIT,NAME='TI:',TYPE='OLD',RECORDSIZE=134,ERR=999)
  999 CALL FLNERR(UNITI,ACCEPT,IUNIT,IGOTO)
      IF(IGOTO.EQ.1) JUNIT=UNITO
      GOTO(3,4,5),IGOTO
    5 RETURN
      END
```

```
      SUBROUTINE FRMAT(DIM,FMT,NTEXT,TEXT,N,A)
C                                                               June '82
C     Makes a run-time format of a question.
C
      INTEGER DIM
      BYTE FMT(DIM),TEXT(NTEXT),A(N)
  300 FORMAT('('I3'H ')
C
      IF(DIM.LT.NTEXT+N+8) STOP 'error in format'
      I=NTEXT+N+1
      ENCODE(6,300,FMT) I
      J=6
      IF(NTEXT.LE.0) GOTO 2
      DO 1 I=1,NTEXT
      J=J+1
    1 FMT(J)=TEXT(I)
    2 IF(N.LE.0)     GOTO 4
      DO 3 I=1,N
      J=J+1
    3 FMT(J)=A(I)
    4 J=J+1
      FMT(J)=36
      J=J+1
      FMT(J)=41
      IF(J.EQ.DIM) RETURN
      J=J+1
      DO 5 I=J,DIM
    5 FMT(I)=32
      RETURN
      END



      SUBROUTINE DECODR(A,ICHAR,NCHAR,FNOT,F)
C                                                           december '81
C     Decodes a real.
C
      BYTE A(NCHAR),B(8)
  300 FORMAT(F8.0)
C                                       first and last numeric character
      L=0
      K=32767
      DO 1 J=ICHAR,NCHAR
      M=A(J)
      IF(K.EQ.32767.AND.(M.EQ.32.OR.M.EQ.0.OR.M.EQ.9.OR.M.EQ.11)) GOTO 1
      IF(K.EQ.32767) K=J
      IF(M.LT.43.OR.M.EQ.44.OR.M.EQ.47.OR.(M.GT.57.AND.M.NE.63))  GOTO 2
      L=J
      IF(M.EQ.63) GOTO 2
    1 CONTINUE
C                                           take over numeric characters
    2 DO 3 J=1,8
      M=L-J+1
      I=9-J
      B(I)=32
      IF(M.GE.K) B(I)=A(M)
    3 CONTINUE
C                                                                 decode
      F=FNOT
      M=B(8)
      IF(M.NE.63.AND.M.NE.32) DECODE(8,300,B,ERR=998) F
    4 ICHAR=NCHAR+1
      RETURN
  998 NCHAR=-1
      GOTO 4
      END
```

```
      SUBROUTINE READI(UNITI,ACCEPT,UNITO,REPORT,FMT,NOT,GETAL,EOF,N,
     *TEXT)
C                                                          december '81
C  Reads an integer from instruction input
C
      BYTE TEXT(N),REPORT,TRUE,A(80),FMT(140),B(13),EOF
      INTEGER GETAL,UNITI,ACCEPT,UNITO
      DATA TRUE/.TRUE./
      DATA B( 1)/ 63/B( 2)/ 32/B( 3)/ 91/B( 4)/105/B( 5)/110/B( 6)/116/
     *     B( 7)/101/B( 8)/103/B( 9)/101/B(10)/114/B(11)/ 93/B(12)/ 61/
     *     B(13)/ 32/
C
  100 FORMAT(Q,80A1)
  202 FORMAT('+'I7)
  203 FORMAT(' Error in integer')
  300 FORMAT(F19.0)
C                                                        write question
      EOF=.FALSE.
    1 IF(UNITI.NE.ACCEPT.AND.REPORT.NE.TRUE) GOTO 2
      CALL FRMAT(140,FMT,N,TEXT,13,B)
      WRITE(UNITO,FMT)
C                                                           read answer
    2 GETAL=NOT
      READ(UNITI,100,ERR=998,END=999) J,(A(I),I=1,J)
      IF(J.EQ.0) GOTO 3
C                                                                decode
      I=1
      CALL DECODI(A,I,J,NOT,GETAL)
      IF(J.LT.0) GOTO 998
    3 IF(REPORT.EQ.TRUE.AND.UNITI.NE.ACCEPT) WRITE(UNITO,202) GETAL
      RETURN
C
  998 IF(UNITI.NE.ACCEPT) STOP 'error in integer'
      WRITE(UNITO,203)
      GOTO 1
  999 EOF=TRUE
      GOTO 3
      END
```

```
      SUBROUTINE READR(UNITI,ACCEPT,UNITO,REPORT,FMT,FNOT,F,EOF,N,
     *TEXT)
C                                                        december '81
C  Reads a real from instruction input
C
      BYTE TEXT(N),REPORT,TRUE,A(80),FMT(140),B(10),EOF,BLANK
      INTEGER UNITI,ACCEPT,UNITO
      EQUIVALENCE (BLANK,B(2))
      DATA TRUE/.TRUE./
      DATA B( 1)/ 63/B( 2)/ 32/B( 3)/ 91/B( 4)/114/B( 5)/101/B(6)/97/
     *     B( 7)/108/B( 8)/ 93/B( 9)/ 61/B(10)/ 32/
C
  100 FORMAT(Q,80A1)
  202 FORMAT('+'8A1)
  203 FORMAT(' Error in real')
  300 FORMAT(G12.3)
C                                                      write question
      EOF=.FALSE.
    1 IF(UNITI.NE.ACCEPT.AND.REPORT.NE.TRUE) GOTO 2
      CALL FRMAT(138,FMT,N,TEXT,10,B)
      WRITE(UNITO,FMT)
C                                                         read answer
    2 F=FNOT
      READ(UNITI,100,ERR=998,END=999) J,(A(I),I=1,J)
      IF(J.EQ.0) GOTO 3
C                                                              decode
      I=1
      CALL DECODR(A,I,J,FNOT,F)
      IF(J.LT.0) GOTO 998
C                                            display in 8 characters
    3 IF(REPORT.NE.TRUE.OR.UNITI.EQ.ACCEPT) RETURN
      ENCODE(12,300,A) F
      DO 4 I=1,8
      IF(A(I).NE.B(2)) GOTO 5
    4 CONTINUE
    5 DO 6 J=I,8
    6 A(J-I+1)=A(J)
      J=8-I+1
      IF(A(9).EQ.BLANK) GOTO 8
      J=8
      DO 7 I=5,8
    7 A(I)=A(I+4)
    8 WRITE(UNITO,202) (A(I),I=1,J)
      RETURN
C
  998 IF(UNITI.NE.ACCEPT) STOP 'error in real'
      WRITE(UNITO,203)
      GOTO 1
  999 EOF=TRUE
      GOTO 3
      END


      SUBROUTINE DECODI(A,ICHAR,NCHAR,NOT,I)
C                                                        december '81
C     Decodes an integer.
C
      BYTE A(NCHAR)
      FNOT=NOT
      I=NOT
      CALL DECODR(A,ICHAR,NCHAR,FNOT,F)
      F=F*1.0000001
      IF(F.GE.-32767.5.AND.F.LE.32767.5) GOTO 1
      IF(NCHAR.GE.0) NCHAR=-1
      RETURN
    1 I=F
      RETURN
      END
```

```
      SUBROUTINE LINE (UNITI,ACCEPT,UNITO,REPORT,NTEKST,LIJN,EOF,N,
     *                  TEXT)
C                                                          december '81
C  Reads a record of instruction input.
C
      BYTE REPORT,EOF,TEXT(N),LIJN(132),TEKST(69),TRUE
      INTEGER UNITI,ACCEPT,UNITO
  300 FORMAT(41(' ')'  - the  first 80 characters')
      DATA TRUE/.TRUE./
      ENCODE(69,300,TEKST)
      DO 1 I=1,N
    1 TEKST(I)=TEXT(I)
      CALL STRING(UNITI,ACCEPT,UNITO,REPORT,80,NTEKST,LIJN(1),EOF,69,
     *            TEKST)
      IF(EOF.EQ.TRUE.OR.NTEKST.LT.80) RETURN
      CALL STRING(UNITI,ACCEPT,UNITO,REPORT,52,I,LIJN(81),EOF,69,
     *'                                       - the   next 52 charac
     *ters')
      NTEKST=NTEKST+I
      RETURN
      END



      SUBROUTINE STRING(UNITI,ACCEPT,UNITO,REPORT,NMAX,NCHAR,REGEL,EOF,
     *N,TEXT)
C                                                          december '81
C  Reads a string of characters from instruction input.
C
      BYTE TEXT(N),REPORT,TRUE,REGEL(NMAX),BLANK,NULL,TAB,EOF
      INTEGER UNITI,ACCEPT,UNITO
      DATA TRUE/.TRUE./
  100 FORMAT(Q,8(255A1))
  200 FORMAT(' '69A1'? [STRING]!')
  201 FORMAT(26(' '80A1))
  202 FORMAT('  Incorrect length of the string!')
  203 FORMAT(' '80A1)
  204 FORMAT(' Error in string')
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++++++ write question
C
      EOF=.FALSE.
    1 IF(UNITI.NE.ACCEPT.AND.REPORT.NE.TRUE) GOTO 3
      WRITE(UNITO,200) (TEXT(I),I=1,N)
      I=80
      IF(NMAX.LT.80) I=NMAX
      DO 2 J=1,I
      REGEL(J)='.'
      IF(J-J/5*5.EQ.0) REGEL(J)='+'
    2 CONTINUE
      WRITE(UNITO,203) (REGEL(J),J=1,I)
C
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ read answer
C
    3 DO 4 I=1,NMAX
    4 REGEL(I)=NULL
      NCHAR=0
      READ(UNITI,100,ERR=998,END=999) NCHAR,(REGEL(I),I=1,NCHAR)
C
C+++++++++++++++++++++++++++++++++++++++++++++++++++++ incorrect format
C
      IF(NCHAR.LE.NMAX) GOTO 5
      IF(UNITI.EQ.ACCEPT.OR.REPORT.EQ.TRUE) WRITE(UNITO,202)
      IF(UNITI.EQ.ACCEPT) GOTO 1
      NCHAR=NMAX
C
C++++++++++++++++++++++++++++++++++++++++++++++++++++ output on terminal
C
    5 IF(REPORT.EQ.TRUE.AND.UNITI.NE.ACCEPT) WRITE(UNITO,201) (REGEL(I),
     *                                        I=1,NCHAR)
      RETURN
  999 EOF=.TRUE.
      RETURN
  998 IF(UNITI.NE.ACCEPT) STOP 'error in string'
      WRITE(UNITO,204)
      GOTO 1
      END
```

```
      SUBROUTINE TIMDAY(UNITO,L,LAST)
C                                                          december '81
C Writes time and date.
C
      BYTE A(17)
      INTEGER UNITO
      EQUIVALENCE (I,A(9)),(J,A(11))
  200 FORMAT( 62(' '),8A1,(' '),9A1)
  201 FORMAT(114(' '),8A1,(' '),9A1)
  300 FORMAT(I2,(' '),I2)
      CALL TIME(A(1))
      DECODE(5,300,A) I,J
      I=60*I+J
      IF(LAST.LT.0) GOTO 1
      IF(LAST-I.GE.1440) LAST=LAST-1440
      IF(I-LAST.LT.3) RETURN
    1 LAST=I
      CALL DATE(A(9))
      IF(L.LE.80) WRITE(UNITO,200) A
      IF(L.GT.80) WRITE(UNITO,201) A
      RETURN
      END


      SUBROUTINE YORNO(UNITI,ACCEPT,UNITO,REPORT,FMT,LOG,EOF,N,TEXT)
C                                                          december '81
C  Asks Yes or No.
C
      BYTE LOG,NO,Y,FALSE,TRUE,BLANK,NUL,REPORT,TEXT(N),FMT(140),EOF,
     *     A(9)
      INTEGER UNITI,ACCEPT,UNITO
      DATA BLANK/32/NO/78/Y/89/TRUE/.TRUE./FALSE/.FALSE./NUL/0/
      DATA A(1)/63/A(2)/32/A(3)/91/A(4)/89/A(5)/47/A(6)/78/A(7)/93/
     *     A(8)/58/A(9)/32/
C
  100 FORMAT(A1)
  202 FORMAT(/' Please type Y,N or <RET>. ',$)
  203 FORMAT('+'A1)
  204 FORMAT(' Error instead of Yes or No')
C
    1 EOF=FALSE
      LOG=1HN
      IF(UNITI.NE.ACCEPT.AND.REPORT.NE.TRUE) GOTO 2
      CALL FRMAT(140,FMT,N,TEXT,9,A)
      WRITE(UNITO,FMT)
    2 READ(UNITI,100,ERR=998,END=999) LOG
      IF(LOG.EQ.Y.OR.LOG.EQ.NO.OR.LOG.EQ.BLANK.OR.LOG.EQ.NUL) GOTO 3
      IF(UNITI.EQ.ACCEPT.OR.REPORT.EQ.TRUE) WRITE(UNITO,202)
      IF(UNITI.EQ.ACCEPT) GOTO 1
      GOTO 998
    3 IF(LOG.NE.Y) LOG=NO
      IF(UNITI.NE.ACCEPT.AND.REPORT.EQ.TRUE) WRITE(UNITO,203) LOG
      IF(LOG.EQ.Y) LOG=TRUE
      IF(LOG.NE.TRUE) LOG=FALSE
      RETURN
  998 IF(UNITI.NE.ACCEPT) STOP 'Error instead of Yes or No'
      WRITE(UNITO,204)
      GOTO 1
  999 EOF=TRUE
      GOTO 3
      END
```