The WINAND STARING CENTRE for Integrated Land, Soil and Water Research

Assessment of soil nutrient depletion in Sub-Saharan Africa: 1983-2000

Volume IV: Computer programmes

J.J. Stoorvogel
E.M.A. Smaling

Report 28

Wageningen (The Netherlands), 1990

Assessment of soil nutrient depletion in Sub-Saharan Africa: 1983-2000

Food and Agricultural Organization of the United Nation

Assessment of soil nutrient depletion in Sub-Saharan Africa: 1983-2000

Volume IV: Computer programmes

J.J. Stoorvogel
E.M.A. Smaling

Report 28

THE WINAND STARING CENTRE, Wageningen, the NETHERLANDS, 1990

ABSTRACT

J.J. Stoorvogel, E.M.A. Smaling, 1990. Assessment of soil nutrient depletion in Sub-Saharan Africa: 1983-20000. Wageningen (The Netherlands), The Winand Staring Centre. Report 28, 4 Volumes; Volume IV.
45 p.; 9 tables.

At the request of FAO a methodology was developed to assess the state of soil nutrient depletion under agriculture in Sub-Saharan Africa for 1983 and the year 2000. The nutrient balance was described by five input and five output factors, resulting in a nutrient loss rate. Production figures and data on fertilizer consumption for 1983 and projections for the year 2000 were provided by FAO. Data on nutrient balances as well as additional country information were collected from the literature. Nutrient depletion rates for Sub-Saharan Africa are approximately 22 kg N, 7 kg P_2O_5 and 18 kg K_2O per ha on average up to a maximum of 60 kg N, 20 kg P_2O_5 and 60 kg K_2O per ha in East Africa.

Keywords: nutrient depletion, nutrient balance, soil fertility, Sub-Saharan Africa.

ISSN 0924-3062

The WINAND STARING CENTRE for Integrated Land, Soil and Water Research, Postbus 125, 6700 AC Wageningen (The Netherlands).
Phone: +318370-74200; fax: +318370-24812; telex: 75230 VISI-NL

The WINAND STARING CENTRE is continuing the research of: Institute for Land and Water Management Research (ICW), Institute for Pesticide Research, Environment Division (IOB), Dorschkamp Research Institute for Forestry and Landscape Planning, Division of Landscape Planning (LB), and Soil Survey Institute (STIBOKA).

Project 8563 005bw/11.90

TABLE OF CONTENTS

		Page
1	INTRODUCTION	7
2	THE CONVERSION PROGRAM: TRANSDAT.PAS	9
2.1	Introduction	9
2.2	Required input	9
2.3	Output	11
2.4	Listing of TRANSDAT.PAS	11
3	THE CALCULATION PROGRAM: NUTDEP.PAS	15
3.1	Introduction	15
3.2	Required input	15
3.3	Output	17
3.4	Listing of NUTDEP.PAS	21
4	EVALUATION	45

Volume I : Main report

Volume II : Nutrient balances per crop and per LUS

Volume III: Literature review and description of land use systems

1 INTRODUCTION

The Department of International Cooperation of the Winand Staring Centre for Integrated Land, Soil and Water Research was requested by the Land and Water Development Division (AGL) of FAO, to assess the state of soil nutrient depletion in Sub-Saharan Africa,
a. in the recent past (1983);
b. in the near future (2000).
This assessment should provide, on a country basis, data on the net removal (depletion minus accumulation) of nutrients from the rootable soil layer.

The assessment was done by means of a nutrient input-output model. A detailed description is given in Volume I. Main input provided by FAO consisted of production data, fertilizer consumption data and the 1 : 5 000 000 soil map. In addition, a literature study provided data on Land Use Systems.

The present volume provides the details on the computer programmes developed for this study. They are programmed in Turbo-Pascal 5.0[1] on a personal computer. All data files are ASCII-files and hence directly accessible.

[1] Turbo Pascal Version 5.0
Copyright by Borland International, Inc.

2 THE CONVERSION PROGRAM: TRANSDAT.PAS

2.1 Introduction

The FAO data used in the present study were arranged in two datafiles. The program TRANSDAT.PAS was developed to extract the relevant information from the two datafiles and arrange the data in the optimal format for subsequent calculation. This proved to save computer time and capacity.
The program first shows a menu, where the default filenames for the input and output files can be changed. In case the filenames are accepted, the procedure will be started.

2.2 Required input

The first datafile, CROSIT.DAT, contains information on arable land, crop yield per hectare and total production on the harvested land. These figures are given per country, per Land/Water Class (LWC) and per crop (Table 2.1-2.3).

The second datafile, INPCRO.DAT, provides information per country, i.e. size of the Land/Water Classes, total amount of fertilizer and other inputs, applied to crops.

Table 2.1 Countries included in CROSIT.DAT and INPCRO.DAT.

1. Angola	20. Malawi
2. Benin	21. Mali
3. Botswana	22. Mauritania
4. Burkina Faso	23. Mauritius
5. Burundi	24. Mozambique
6. Cameroon	25. Niger
7. Central African Republic	26. Nigeria
	27. Rwanda
8. Chad	28. Senegal
9. Congo	29. Sierra Leone
10. Côte d'Ivoire	30. Somalia
11. Ethiopia	31. Sudan
12. Gabon	32. Swaziland
13. Gambia	33. Tanzania
14. Ghana	34. Togo
15. Guinea	35. Uganda
16. Kenya	36. Zaire
17. Lesotho	37. Zambia
18. Liberia	38. Zimbabwe
19. Madagascar	

Table 2.2 Land/Water Classes included in CROSIT.DAT.

Land/Water Class	Specification
Low rainfall, rainfed land (LR)	Rainfall providing 1-119 growing days and soil quality very suitable, suitable or marginally suitable
Uncertain rainfall, rainfed land (UR)	Rainfall providing 120-179 growing days and soil quality very suitable or suitable
Good rainfall, rainfed land (GR)	Rainfall providing 180-269 growing days and soil quality very suitable or suitable
Problem, rainfed land (PR)	Rainfall providing >269 growing days, and soil quality very suitable, suitable or marginally suitable; or rainfall providing 120-269 growing days and soil quality marginally suitable
Naturally flooded land (NF)	Land under water for part of the year and lowland non-irrigated paddy fields
Irrigated land (IR)	Fully and partially irrigated land

Table 2.3 Crops included in CROSIT.DAT and INPCRO.DAT.

1. Wheat	19. Other fruit
2. Rice	20. Oil crops other than 21-26
3. Maize	21. Palm oil
4. Barley	22. Soybeans
5. Millet	23. Groundnuts
6. Sorghum	24. Sunflower seed
7. Other cereals	25. Sesame seed
8. Potatoes	26. Coconut
9. Sweet potatoes and yams	27. Cocoa beans
10. Cassava	28. Coffee beans
11. Other roots	29. Tea
12. Plantains	30. Tobacco
13. Beet	31. Seed cotton
14. Cane	32. Jute and hard fibres
15. Pulses	33. Rubber
16. Vegetables	34. Fodder
17. Bananas	35. Other crops
18. Citrus fruit	

2.3 Output

The output of TRANSDAT.PAS and its format is displayed in Table 2.4. This output constitutes the input for the calculation program NUTDEP.PAS. All data are presented as integers delimited by several blanks.

Table 2.4 The format of the output table of TRANSDAT.PAS.

COUNTRY			YEAR											
N	P	K	A1	Y1	A2	Y2	A3	Y3	A4	Y4	A5	Y5	A6	Y6 (CROP 1)
N	P	K	A1	Y1	A2	Y2	A3	Y3	A4	Y4	A5	Y5	A6	Y6 (CROP 2)
N	P	K	A1	Y1	A2	Y2	A3	Y3	A4	Y4	A5	Y5	A6	Y6 (CROP 3)
|	|	|	|	|	|	|	|	|	|	|	|	|	|	|
|	|	|	|	|	|	|	|	|	|	|	|	|	|
|	|	|	|	|	|	|	|	|	|	|	|	|	|
|	|	|	|	|	|	|	|	|	|	|	|	|	|
N	P	K	A1	Y1	A2	Y2	A3	Y3	A4	Y4	A5	Y5	A6	Y6 (CROP 34)
0	0	0	F1	0	F2	0	F3	0	F4	0	F5	0	F6	0 (FALLOW)

In which:
COUNTRY : country serial number
YEAR : 1=1983/2=2000
N : Total N application (ton/country,crop)
P : Total P_2O_5 application (ton/country,crop)
K : Total K_2O application (ton/country,crop)
A1 : Harvested area in Land/Water Class 1 (000 ha)
Y1 : Average yield in Land Water Class 1 (00 kg/ha)
F1 : Fallow area in Land/Water Class

2.4 Listing of TRANSDAT.PAS

```
PROGRAM TRANSDAT (INPUT,OUTPUT,FAO1,FAO2,OUTDAT);


{-------------------------------------------------------------}
{                                                             }
{    Program TRANSDAT                                         }
{    Function  : Select a number of data out of several       }
{                datafiles, and write it to one new           }
{                database.                                    }
{    Interface : Input of FAO data files, after               }
{                selection of data output to one out-         }
{                put file.                                    }
{    Author    : Winand Staring Centre                        }
{                Ir. J.J.Stoorvogel                           }
{                                                             }
{-------------------------------------------------------------}

TYPE
   STRING          =  VARYING[14] OF CHAR;
   BLOK            =  ARRAY[1..37,1..16,1..2] OF INTEGER;

VAR
   FILE1,FILE2,FILE3                 :   STRING;
   ALL                               :   BLOK;
   FAO1,FAO2,OUTDAT                  :   TEXT;
   I,I1,I2,J1,J2,J3,COUNTRY,YEAR     :   INTEGER;
   Z,YIELD                           :   REAL;
```

```
BEGIN
FILE1:='CROSIT'; FILE2:='INPCRO'; FILE3:='NUTDEP';
REPEAT
   WRITELN ('-----------------------------------------------');
   WRITELN;
   WRITELN ('  1. CROP PRODUCTION PROGRAM :',FILE1);
   WRITELN ('  2. INPUT REQUIREMENTS      :',FILE2);
   WRITELN ('  3. OUTPUT FILE             :',FILE3);
   WRITELN ('  9. EXIT TRANSDAT');
   WRITELN ('  0. CONTINUE');
   WRITELN;
   WRITELN ('-----------------------------------------------');
   WRITE ('# GIVE CHOICE: ');
   READLN (I);
   WRITELN ('-----------------------------------------------');
 CASE I OF
      1: BEGIN
         WRITE (' GEVE NAME FILE: CROP PRODUCTION: ');
         READLN (FILE1);
         END;
      2: BEGIN
         WRITE (' GIVE NAME FILE: INPUT REQUIREMENTS: ');
         READLN (FILE2);
         END;
      3: BEGIN
         WRITE (' GIVE NAME FILE: OUTPUT: ');
         READLN (FILE3);
         END;
   END;  { CASE }
UNTIL (I=0) OR (I=9);
IF (I=0) THEN
   BEGIN
   OPEN(FAO1,FILE1,OLD);
   RESET(FAO1);
   OPEN(FAO2,FILE2,OLD);
   RESET(FAO2);
   OPEN(OUTDAT,FILE3,NEW);
   REWRITE(OUTDAT);
   FOR I1:=1 TO 38 DO
         BEGIN
      { 2. INPUT FROM DATA OUT OF FAO1 }
      READLN(FAO1);
      FOR J1:=1 TO 37 DO
         BEGIN
         FOR J2:=1 TO 2 DO
            BEGIN
            FOR J3:=1 TO 6 DO
               BEGIN
               READ(FAO1,ALL[J1,(2+2*J3),J2],YIELD,Z);
               ALL[J1,(3+2*J3),J2]:=TRUNC(10*YIELD);
               END;
            READLN(FAO1);
            END; { FOR J2:=1 TO 2    }
         END; { FOR J1:=1 TO 37 }
      { 3. INPUT FROM DATA OUT OF FAO2 }
      FOR J1:=1 TO 2 DO
         BEGIN
         READLN(FAO2,COUNTRY,YEAR);
         WRITELN('LAND: ',COUNTRY:3,' YEAR: ',YEAR:5);
         FOR J2:=1 TO 34 DO
```

```
            BEGIN
            FOR J3:=1 TO 11 DO
               READ(FAO2,Z);

READLN(FAO2,ALL[J2,1,J1],ALL[J2,2,J1],ALL[J2,3,J1]);
            END; { FOR J2:=1 TO 35 }
         END; { FOR J1:=1 TO 2 }
      { 4. OUTPUT OF DATA TO OUTDAT }
      FOR J1:=1 TO 2 DO
         BEGIN
         WRITELN(OUTDAT,I1:2,J1:3);
         FOR J2:=1 TO 37 DO
            BEGIN
            IF (J2<>36) THEN
               BEGIN
               FOR J3:=1 TO 15 DO
                  WRITE(OUTDAT,ALL[J2,J3,J1]:6);
               WRITELN(OUTDAT);
               END;
            END; { FOR J2:=1 TO 35 }
         END { FOR J1:=1 TO 2 }
      END; { FOR I1:=1 TO 38 }
   END; { IF I=0 }
END. { TRANSDAT }
```

3 THE CALCULATION PROGRAM: NUTDEP.PAS

3.1 Introduction

The nutrient input-output model, developed for the present study and described in Volume I, has been programmed into NUTDEP.PAS. Three input files are described in the following section. They are needed to run the program for one country. If the program is run for all 38 countries, 40 datafiles are necessary. The program imports default filenames from one datafile. The first menu provides the option to change these defaults. The second menu asks the operator to choose one country for analysis or to perform the analysis for the entire Sub-Saharan Africa. For each country, two output files were created. One contains full results per Land Use System and per crop, the other gives only a summary of results displayed per Land/Water Class. Section 3.2 goes into detail as to required input and Section 3.3 is concerned with the results.

3.2 Required input

Two datafiles, CHEMCOM.DAT and TRANSDAT.DAT, are needed to run the program. In addition, a datafile is needed for every country with the Land Use System information (country.LUS).

CHEMCOM.DAT
CHEMCOM.DAT contains the N, P and K contents of harvested product and residues for all 35 crops. The contents of CHEMCOM.DAT are given in Table 3.1. The sources of these data are given in Volume 3.

TRANSDAT.DAT
TRANSDAT.DAT has been described in Section 2.3, as the output of the program TRANSDAT.PAS. It contains data on area, production and fertilizer application.

Country.LUS
This file consists of an abbreviation of the country name with the extension .LUS. Abbreviated country names are listed in Table 3.2. The file provides information on the Land Use Systems that occur within the specific country. Table 3.37 displays the structure of these files. The data for the different countries are given in a readable format in Volume 3.

Table 3.1 The contents of CHEMCOM.DAT.

Harvested product (0.1 kg/ton)			Crop residues (0.1 kg/ton)						Average yields (0.1 ton/ha)		
N	P_2O_5	K_2O	N min	N max	P_2O_5 min	P_2O_5 max	K_2O min	K_2O max	min	max	
223	99	70	43	43	41	41	320	320	10	21	(Wheat)
116	78	41	90	136	36	70	340	518	12	20	(Rice)
168	94	57	76	118	30	58	230	284	9	13	(Maize)
155	64	72	70	70	23	23	252	252	15	25	(Barley)
192	137	65	161	246	84	98	663	771	6	9	(Millet)
145	126	45	81	135	78	132	305	395	6	11	(Sorghum)
167	101	58	82	136	42	64	270	386	6	9	(O cereals)
44	30	83	23	23	16	16	54	54	60	83	(Potatoes)
48	18	88	21	21	27	27	39	39	57	87	(S potatoes)
42	11	51	24	68	7	35	16	18	51	85	(Cassava)
46	7	35	19	19	11	11	37	37	37	64	(Other roots)
7	2	41	12	12	7	7	77	77	58	73	(Plantain)
999	999	999	999	999	999	999	999	999	000	000	(Beet)
6	5	14	3	3	7	7	4	4	52	89	(Cane)
200	78	133	104	104	23	23	157	157	5	7	(Pulses)
90	21	31	32	32	32	32	94	94	38	56	(Vegetables)
12	7	54	16	16	7	7	143	143	63	83	(Bananas)
18	5	28	6	6	5	5	53	53	47	70	(Citrus)
20	5	24	18	18	5	5	59	59	46	70	(Other fruits)
26	11	53	3	3	14	14	65	65	3	4	(Oil crops)
29	16	49	33	41	12	14	25	55	7	17	(Palm oil)
621	251	240	130	222	63	73	164	182	5	12	(Soybeans)
372	137	98	122	196	28	82	116	242	5	8	(Groundnuts)
240	80	66	230	230	73	73	496	496	6	8	(Sunflowers)
300	140	81	150	150	124	124	253	253	3	4	(Sesam)
610	165	118	270	270	131	131	304	304	20	38	(Coconut)
400	195	231	199	199	108	108	399	399	2	3	(Cacao)
350	60	202	43	43	87	87	111	111	4	7	(Coffee)
350	87	161	1	1	0	0	0	0	11	18	(Tea)
560	188	872	1	1	0	0	2	2	7	10	(Tobacco)
187	222	108	114	164	111	167	314	402	6	11	(Cotton)
50	9	72	21	21	16	16	108	108	4	7	(Fibres)
69	27	55	10	10	5	5	48	48	5	6	(Rubber)
68	30	57	0	0	0	0	0	0	44	59	(Fodder)

Table 3.2 Abbreviated country names.

1.	ANGO	Angola	20.	MALA	Malawi
2.	BENI	Benin	21.	MALI	Mali
3.	BOTS	Botswana	22.	MAUA	Mauritania
4.	BURK	Burkina Faso	23.	MAUS	Mauritius
5.	BURU	Burundi	24.	MOZA	Mozambique
6.	CAME	Cameroon	25.	NIGE	Niger
7.	CARE	Central African Republic	26.	NIGA	Nigeria
			27.	RWAN	Rwanda
8.	CHAD	Chad	28.	SENE	Senegal
9.	CONG	Congo	29.	SIER	Sierra Leone
10.	CDIV	Côte d'Ivoire	30.	SOMA	Somalia
11.	ETHI	Ethiopia	31.	SUDA	Sudan
12.	GABO	Gabon	32.	SWAZ	Swaziland
13.	GAMB	Gambia	33.	TANZ	Tanzania
14.	GHAN	Ghana	34.	TOGO	Togo
15.	GUIN	Guinea	35.	UGAN	Uganda
16.	KENY	Kenya	36.	ZAIR	Zaire
17.	LESO	Lesotho	37.	ZAMB	Zambia
18.	LIBE	Liberia	38.	ZIMB	Zimbabwe
19.	MADA	Madagascar			

Table 3.3 The format of the country.LUS files.

Country number	Year		
LWC number	Number of LUS	Rainfall	Fertility
LUS code	Number of crops		
Fertilizer	Manure	Residues	Erosion
Crop number	Percentage		
Crop number	Percentage		
LUS code	Number of crops		
Fertilizer	Manure	Residues	Erosion
Crop number	Percentage		
Crop number	Percentage		
Crop number	Percentage		
LWC number	Number of LUS	Rainfall	Fertility
LUS code	Number of crops		
Fertilizer	Manure	Residues	Erosion
Crop number	Percentage		
Crop number	Percentage		
LUS code	Number of crops		
Fertilizer	Manure	Residues	Erosion
Crop number	Percentage		
Crop number	Percentage		
Crop number	Percentage		

In which:
Rainfall : Average for Land/Water Class, in mm/yr
Fertility : Inherent Soil fertility (three classes)
LUS code : Management level
Fertilizer: Weighing factor showing regional distribution of total national consumption
Manure : in kg/ha,yr or 'during grazing'
Residues : Percentage of crop residues removed or 'during grazing'
Erosion : Soil loss in ton/ha,yr
Percentage: Percentage of the crop that occurs in a certain LUS

3.3 Output

The output of the model consists of two files for every country, country.RES and country.SUM. Country.RES contains full results as presented in Volume 2. All the nutrient balances per Land Use System and per crop are given for the two years and the three nutrients. In country.SUM only a summary of the results is given as presented in Volume 1. The nutrient balances per Land Water Class for the two years and the three nutrients are listed.
Table 3.4 and 3.5 give an example of these two files for Senegal.

Table 3.4 The contents of SENE.RES.

LWC/LUS	CROP	AREA 000 HA	FERT	MANU	DEPO	FIXA	SEDI	PROD	RESI	LEAC	GASE	EROS	FALL	TOTAL
COUNTRY (TON/COUNTRY)	TOTAL	5235	8805	7516	9538	37335	1256	43677	18454	9761	21192	41075	5554	-64157
LR (TON/LWC)	TOTAL	1100	603	1560	1602	5615	0	5949	3467	1577	2767	5720	1056	-9044
LR-L.1 (KG/HA)	TOTAL	572	1	3	3	10	0	10	6	3	5	10		-18
	MILL	255	0	3	3	3	0	6	6	3	5	10		-21
	SORG	31	1	2	3	3	0	7	5	3	5	10		-22
	GROU	286	2	3	3	17	0	15	6	2	5	10		-14
FALLOW		528											2	2
UR (TON/LWC)	TOTAL	3189	3296	5956	6027	26194	0	29506	13333	5631	12701	29130	3508	-45321
UR-L.1 (KG/HA)	TOTAL	527	1	3	4	4	0	12	7	4	9	20		-40
	MAIZ	25	2	2	4	4	0	15	5	4	9	20		-41
	MILL	447	1	3	4	4	0	12	7	4	9	20		-39
	SORG	53	1	3	4	4	0	16	7	4	8	20		-43
	OTHC	2	0	0	4	4	0	0	0	5	10	20		-27
UR-L.2 (KG/HA)	TOTAL	865	2	5	4	28	0	26	11	4	8	20		-29
	PULS	9	0	1	4	8	0	4	2	5	10	20		-28
	GROU	856	2	5	4	28	0	26	11	4	8	20		-29
UR-L.3 (KG/HA)	TOTAL	13	3	0	4	4	0	15	7	3	8	30		-52
	SPOT	1	0	0	4	4	0	22	5	2	7	30		-57
	CASS	7	0	0	4	4	0	12	10	2	7	30		-53
	VEGE	2	1	0	4	4	0	25	4	2	7	30		-60
	FRUI	3	10	0	4	4	0	11	5	6	11	30		-45
UR-L.4 (KG/HA)	TOTAL	30	25	0	4	4	0	21	0	9	32	30		-58
	COTT	30	25	0	4	4	0	21	0	9	32	30		-58
FALLOW		1754											2	2
GR (TON/LWC)	TOTAL	298	1468	0	737	2150	0	3383	1136	884	1992	4480	292	-7227
GR-L.1 (KG/HA)	TOTAL	85	5	0	5	5	0	16	4	5	12	30		-52
	MAIZ	50	7	0	5	5	0	18	4	5	12	30		-53
	MILL	31	2	0	5	5	0	12	4	5	11	30		-49
	SORG	4	3	0	5	5	0	15	4	4	11	30		-52
GR-L.2 (KG/HA)	TOTAL	51	8	0	5	32	0	30	15	4	11	30		-45
	GROU	51	8	0	5	32	0	30	15	4	11	30		-45
GR-L.3 (KG/HA)	TOTAL	1	0	0	5	5	0	29	4	2	9	20		-54
	POTA	1	0	0	5	5	0	29	4	2	9	20		-54
GR-L.4 (KG/HA)	TOTAL	3	0	0	5	5	0	85	8	0	1	10		-94
	COCN	3	0	0	5	5	0	85	8	0	1	10		-94
GR-H.1 (KG/HA)	TOTAL	1	67	0	5	5	0	15	1	24	30	20		-13
	FRUI	1	67	0	5	5	0	15	1	24	30	20		-13
GR-H.2 (KG/HA)	TOTAL	11	50	0	5	5	0	21	0	18	37	30		-46
	COTT	11	50	0	5	5	0	21	0	18	37	30		-46
FALLOW		146											2	2

LWC/LUS	CROP	AREA 000 HA	FERT	MANU	DEPO	FIXA	SEDI	PROD	RESI	LEAC	GASE	EROS	FALL	TOTAL
PR (TON/LWC)	TOTAL	489	853	0	739	1913	0	2051	346	726	1198	1745	626	-1934
PR-L.1 (KG/HA)	TOTAL	173	5	0	4	11	0	12	2	4	7	10		-15
	RICE	11	12	0	4	5	0	12	3	6	9	10		-18
	MILL	88	2	0	4	5	0	8	2	4	6	10		-18
	SORG	11	3	0	4	5	0	9	2	4	7	10		-19
	PULS	3	0	0	4	9	0	4	0	4	7	10		-13
	GROU	60	8	0	4	22	0	19	2	4	7	10		-8
PR-L.2 (KG/HA)	TOTAL	3	0	0	4	5	0	8	2	3	6	5		-14
	PALM	3	0	0	4	5	0	8	2	3	6	5		-14
FALLOW		313											2	2
NF (TON/LWC)	TOTAL	20	121	0	80	216	226	233	27	91	292	0	0	0
NF-L.1 (KG/HA)	TOTAL	18	7	0	4	12	13	13	1	5	16	0		0
	RICE	9	12	0	4	22	3	16	2	6	17	0		0
	MILL	5	2	0	4	2	22	10	1	4	15	0		0
	SORG	1	3	0	4	2	21	9	1	5	16	0		0
	OTHC	3	0	0	4	2	24	10	1	4	15	0		0
FALLOW		2											0	0
IR	TOTAL	139	2463	0	353	1247	1030	2556	148	853	2242	0	72	-631
IR-H.1 (KG/HA)	TOTAL	70	35	0	3	14	10	34	1	10	24	0		-6
	RICE	43	18	0	3	22	10	24	1	6	20	0		3
	CANE	6	233	0	3	2	10	59	1	66	80	0		43
	VEGE	14	7	0	3	2	10	61	1	0	12	0		-52
	BANA	1	0	0	3	2	10	7	0	3	17	0		-12
	CITR	4	25	0	3	2	10	9	0	11	25	0		-5
	FRUI	2	50	0	3	2	10	29	1	14	28	0		-7
IR-H.2 (KG/HA)	TOTAL	33	0	0	3	7	10	6	2	4	18	0		-9
	PULS	33	0	0	3	7	10	6	2	4	18	0		-9
IR-H.3 (KG/HA)	TOTAL	0	0	0	0	0	0	0	0	0	0	0		0
FALLOW		36											2	2

Table 3.5 The contents of SENE.SUM.

Table 3.5a LWC-distribution in Senegal (in % of total arable area).

	LR	UR	GR	PR	NF	IR
1983	21	61	6	9	0	3
2000	22	60	6	9	0	3

Table 3.5b Senegal; nutrient balances for LWC s and for country total.

		AREA 000HA	FERT	MANU	DEPO	FIXA	SEDI	PROD	RESI	LEAC	GASE	EROS	FALL	TOTAL (TON)	(KG/HA)
* LOW RAINFALL															
N	1983	1100	603	1567	1602	5615	0	5903	3482	1582	2771	5720	1056	-9015	-8
	2000	1233	1872	2857	2002	7990	0	9951	6350	2075	3562	7150	1036	-13331	-11
P205	1983	1110	1037	640	606	0	0	2743	1423	0	0	1716	1056	-2543	-2
	2000	1233	2944	1162	758	0	0	4898	2583	0	0	2145	1036	-3726	-3
K20	1983	1100	491	3378	1258	0	0	1679	7507	1881	0	4290	528	-9702	-9
	2000	1233	1266	6657	1573	0	0	2890	14793	3139	0	5362	518	-16170	-13
* UNCERTAIN RAINFALL															
N	1983	3189	3296	5955	6027	26194	0	29271	13330	5654	12725	29130	3508	-45130	-14
	2000	3444	8893	11304	8245	44211	0	54439	25280	7436	17126	39840	2962	-68506	-20
P205	1983	3189	5097	2461	2282	0	0	13327	5524	0	0	10924	3508	-16427	-5
	2000	3444	13847	4667	3121	0	0	25249	10463	0	0	14940	2962	-26055	-8
K20	1983	3189	2475	10578	4735	0	0	8530	23609	9803	0	21847	1754	-44247	-14
	2000	3444	6308	21020	6478	0	0	15920	46902	15176	0	29880	1481	-72591	-21
* GOOD RAINFALL															
N	1983	298	1468	0	737	2150	0	3371	1138	884	1993	4480	292	-7219	-24
	2000	324	2429	0	660	1873	0	4542	1344	884	1834	3960	376	-7226	-22
P205	1983	298	1766	0	279	0	0	1725	490	0	0	1680	292	-1558	-5
	2000	324	2729	0	250	0	0	2313	597	0	0	1485	376	-1040	-3
K20	1983	298	794	0	579	0	0	1102	1933	1025	0	3360	146	-5901	-20
	2000	324	1047	0	518	0	0	1507	2588	742	0	2970	188	-6054	-19
* PROBLEM AREA															
N	1983	489	853	0	739	1913	0	2039	350	725	1197	1745	626	-1925	-4
	2000	517	3407	0	1021	3145	0	4182	748	1401	2053	2410	548	-2673	-5
P205	1983	489	1306	0	280	0	0	1057	145	0	0	523	626	487	1
	2000	517	4608	0	386	0	0	2237	310	0	0	723	548	2272	4
K20	1983	489	2176	0	581	0	0	644	881	1185	0	1309	313	-949	-2
	2000	517	5794	0	802	0	0	1330	1952	2659	0	1807	274	-878	-2
* NATURALLY FLOODED															
N	1983	20	121	0	80	216	226	234	27	90	292	0	0	0	0
	2000	25	371	0	102	246	323	400	47	158	415	0	0	22	1
P205	1983	20	112	0	30	0	33	158	13	0	0	0	0	4	0
	2000	25	342	0	39	0	28	273	21	0	0	0	0	115	5
K20	1983	20	36	0	63	0	99	82	95	20	0	0	0	1	0
	2000	25	127	0	80	0	118	139	159	26	0	0	0	1	0
* IRRIGATED AREA															
N	1983	139	2463	0	353	1247	1030	2556	146	853	2242	0	72	-632	-5
	2000	177	8634	0	545	2141	1590	5650	318	2517	4648	0	35	-188	-1
P205	1983	139	2387	0	134	0	309	1304	71	0	0	0	72	1527	11
	2000	177	8531	0	206	0	477	3126	168	0	0	0	35	5955	34
K20	1983	139	1326	0	278	0	515	1777	345	544	0	0	36	-511	-4
	2000	177	4811	0	428	0	795	4637	749	1725	0	0	18	-1059	-6
* TOTAL															
N	1983	5235	8804	7522	9538	37335	1256	43374	18473	9788	21220	41075	5554	-63921	-12
	2000	5720	25606	14161	12575	59606	1913	79164	34087	14471	29638	53360	4956	-91902	-16
P205	1983	5235	11705	3101	3611	0	342	20314	7666	0	0	14843	5554	-18510	-4
	2000	5720	33001	5829	4760	0	505	38096	14142	0	0	19293	4956	-22479	-4
K20	1983	5235	7298	13956	7494	0	614	13814	34370	14458	0	30806	2777	-61309	-12
	2000	5720	19353	27677	9879	0	913	26423	67143	23467	0	40019	2478	-96751	-17

3.4 Listing of NUTDEP.PAS

```
{-------------------------------------------------------}
{ Program  : NUTDEP - NUTrient DEPletion                 }
{ Purpose  : The assessment of the nutrient depletion    }
{ Interface: Data are retrieved from several datafiles.  }
{            After the calculation of the nutrient       }
{            depletion the results are written to two    }
{            datafiles with extensions .sum and .res.    }
{            The first will contain the nutrient balances }
{            per LWC and country. The second will contain }
{            the nutrient balances per crop and LUS.     }
{  Author  : WINAND STARING CENTRE - Ir. J.J. Stoorvogel }
{-------------------------------------------------------}

PROGRAM NUTDEP (INPUT,OUTPUT,IFL,OFL);

USES CRT;

VAR
 CNTRNR,                    { country number              }
 MENU1,                     { answer to the first menu    }
 MENU2,                     { answer to the second menu   }
 NOLUS,                     { number of land use systems  }
 YEAR,                      { year (1=1983, 2=2000)       }
 NOCROPS,                   { number of crops             }
 LWC,                       { land water class            }
 LUS,                       { land use system             }
 CROP,                      { relative crop number        }
 CROPNR,                    { absolute crop number        }
 NPK,                       { nutrient (1=N,2=P2O5,3=K2O) }
 SFERT,                     { soil fertility class        }
 MANU,                      { manure application class    }
 MANA,                      { management level            }
 FLOW                       { flow number                 }
            : SHORTINT;
 LINE,                      { line number                 }
 LUSLINE,                   { line with lus data          }
 LWCLINE,                   { line with lwc data          }
 RAIN,                      { rainfall                    }
 RESI,                      { % of crop residues removed  }
 CHEMCONC,                  { nutrient content of the crop
                              residues used for the
                              calculations                }
 CHECK,                     { check on input              }
 TOTAREA,                   { total harvested area        }
 SUBAREA,                   { harvested area under multiple
                              cropping                    }
 DEPDIF,                    { differences in depletion    }
 AREA,                      { area under cultivation      }
 YIELD,                     { yield                       }
 I1,I2          : INTEGER;
 FERTFAC,                   { weighing factor for
                              fertilizer application      }
 AREAFAC,                   { factor for the calculation
                              of the area under multiple  }
                              cropping
 TOTAL          : REAL;     { total                       }
 IFL,                       { input file                  }
```

```
OFL          : TEXT;          { output file                    }
BURNING,                      { true if burning of the crop
                                residues takes place           }
ONCE1,                        { check on procedure for unit    }
ONCE2,                        { check on procedure for fallow  }
CONTINUE,                     { quit NUTDEP (Y/N)              }
ALL                           { calculations for all countries}
             : BOOLEAN;
SUMMARY                       { summary of results             }
             : ARRAY[1..42,0..14] OF LONGINT;
CI                            { cropping intensity             }
             : ARRAY[1..6] OF INTEGER;
DEPCLASS                      { depletion classes              }
             : ARRAY[1..6] OF SHORTINT;
CRPFRT                        { Σ(area*rel fert use)           }
             : ARRAY[1..35] OF INTEGER;
CC                            { chemical compositions          }
             : ARRAY[1..35,1..9] OF INTEGER;
LI                            { LUS data                       }
             : ARRAY[1..200,1..5] OF INTEGER;
PD                            { production data                }
             : ARRAY[1..37,1..15] OF INTEGER;
OT                            { output data                    }
             : ARRAY[0..200,1..38] OF REAL;
FALLOW                        { area and chemical data,fallow  }
             : ARRAY[0..6,1..4] OF REAL;
UPTAKE                        { uptake of N, P2O5 and K2O      }
             : ARRAY[1..6] OF REAL;   LUSCODE
                              { code for LUS                   }
             : ARRAY[1..6,1..6] OF STRING[3];
  CNTR4,                      { country abbreviation           }
  CRP4                        { crop abbreviation              }
             : STRING[4];
  FN1,                        { file name CC                   }
  FN2,                        { file name LI                   }
  FN3,                        { file name PD                   }
  DIR,                        { directory                      }
  ON1,                        { output file all results        }
  ON2,                        { output file summary of results}
  CNTR25                      { full country name              }
             : STRING[25];
  CNTRNMS,                    { all country abbreviations      }
  CRPNMS                      { all crop abbreviations         }
             : STRING[152];
```

```
PROCEDURE Country;

BEGIN
 CASE CNTRNR OF
    1: CNTR25:='Benin';
    2: CNTR25:='Gambia';
    3: CNTR25:='Ghana';
    4: CNTR25:='Guinea';
    5: CNTR25:='Côte d ivoire';
    6: CNTR25:='Liberia';
    7: CNTR25:='Mali';
    8: CNTR25:='Mauritania';
    9: CNTR25:='Niger';
   10: CNTR25:='Nigeria';
   11: CNTR25:='Senegal';
   12: CNTR25:='Sierra Leone';
   13: CNTR25:='Togo';
   14: CNTR25:='Burkina Faso';
   15: CNTR25:='Angola';
   16: CNTR25:='Cameroon';
   17: CNTR25:='Central African Republic';
   18: CNTR25:='Chad';
   19: CNTR25:='Congo';
   20: CNTR25:='Gabon';
   21: CNTR25:='Zaire';
   22: CNTR25:='Burundi';
   23: CNTR25:='Ethiopia';
   24: CNTR25:='Kenya';
   25: CNTR25:='Madagascar';
   26: CNTR25:='Malawi';
   27: CNTR25:='Mauritius';
   28: CNTR25:='Mozambique';
   29: CNTR25:='Zimbabwe';
   30: CNTR25:='Rwanda';
   31: CNTR25:='Somalia';
   32: CNTR25:='Tanzania';
   33: CNTR25:='Uganda';
   34: CNTR25:='Zambia';
   35: CNTR25:='Sudan';
   36: CNTR25:='Botswana';
   37: CNTR25:='Lesotho';
   38: CNTR25:='Swaziland';
 END; { CASE }
 CNTR4:=COPY(CNTRNMS,(CNTRNR*4-3),4);
END;
```

```
PROCEDURE MenuCntr;

BEGIN
 WRITELN('┌──────────────────────────────────────────────────┐');
 WRITELN('│ 1 BENI  11 SENE   21 ZAIR   31 SOMA              │');
 WRITELN('│ 2 GAMB  12 SIER   22 BURU   32 TANZ              │');
 WRITELN('│ 3 GHAN  13 TOGO   23 ETHI   33 UGAN              │');
 WRITELN('│ 4 GUIN  14 BURK   24 KENY   34 ZAMB              │');
 WRITELN('│ 5 CDIV  15 ANGO   25 MADA   35 SUDA              │');
 WRITELN('│ 6 LIBE  16 CAME   26 MALA   36 BOTS              │');
 WRITELN('│ 7 MALI  17 CARE   27 MAUS   37 LESO              │');
 WRITELN('│ 8 MAUA  18 CHAD   28 MOZA   38 SWAZ              │');
 WRITELN('│ 9 NIGE  19 CONG   29 ZIMB                        │');
 WRITELN('│10 NIGA  20 GABO   30 RWAN  100 ALL               │');
 WRITELN('└──────────────────────────┐                       ┘');
 WRITELN('│ GIVE COUNTRY NUMBER:     │ ');
 WRITELN('└──────────────────────────┘ ');
 REPEAT
   GOTOXY(24,23);
   READLN(CNTRNR);
 UNTIL CNTRNR IN [1..38,100];
END;


PROCEDURE Menu_1;

BEGIN
GOTOXY(1,11);
 FOR I1:=1 TO 14 DO DELLINE;
 DIR:='C:\NUTDEP\';
 FN1:='CHEMCOM.DAT';
 FN2:='DATA\'+CNTR4+'.LUS';
 FN3:='NUTDEP.DAT';
 ON1:=CNTR4+'.RES';
 ON2:=CNTR4+'.SUM';
 IF ALL=FALSE THEN
   BEGIN
   WRITELN('┌──────────────────────────────────────────────┐');
   WRITELN('│ 1. DIRECTORY                 : ',DIR);
   WRITELN('│ 2. FILENAME CHEMCOM          : ',FN1);
   WRITELN('│ 3. FILENAME LUSINFO          : ',FN2);
   WRITELN('│ 4. FILENAME PRODDAT          : ',FN3);
   WRITELN('│ 5. FILENAME OUTFILE          : ',ON1);
   WRITELN('│ 6. FILENAME SUMMARY          : ',ON2);
   WRITELN('│ 0. ACCEPT FILENAMES AND CONTINUE');
   WRITELN('│ 9. QUIT NUTDEP');
   WRITELN('└──────────────────────┐                       ┘');
   WRITELN('│ GIVE YOUR CHOICE:    │ ');
   WRITELN('└──────────────────────┘ ');
   END;
```

```
 FOR I1:=1 TO 8 DO
    BEGIN
    GOTOXY(50,(11+I1));
    WRITE('|');
    END;
 REPEAT { UNTIL MENU1 IN [0,9]  }
 IF ALL=FALSE THEN
    BEGIN
    REPEAT { UNTIL MENU1 IN [0..5,9]  }
       GOTOXY(21,21);
       READLN(MENU1);
    UNTIL MENU1 IN [0..6,9];
    END
 ELSE
    MENU1:=0;
 CASE MENU1 OF
    1: BEGIN
       GOTOXY(32,12);
       WRITE('                    ');
       GOTOXY(32,12);
       READLN(DIR)
       END;
    2: BEGIN
       GOTOXY(32,13);
       WRITE('                    ');
       GOTOXY(32,13);
       READLN(FN1)
       END;
    3: BEGIN
       GOTOXY(32,14);
       WRITE('                    ');
       GOTOXY(32,14);
       READLN(FN2)
       END;
    4: BEGIN
       GOTOXY(32,15);
       WRITE('                    ');
       GOTOXY(32,15);
       READLN(FN3)
       END;
    5: BEGIN
       GOTOXY(32,16);
       WRITE('                    ');
       GOTOXY(32,16);
       READLN(ON1)
       END;
    6: BEGIN
       GOTOXY(32,17);
       WRITE('                    ');
       GOTOXY(32,17);
       READLN(ON2)
       END;
 END; {  CASE  }
 UNTIL MENU1 IN [0,9];
END;
```

```
PROCEDURE Menu_2;

BEGIN
 GOTOXY(1,1);
 CLREOL;
 GOTOXY(1,11);
 FOR I1:=1 TO 11 DO DELLINE;
 WRITELN('                                          ');
 WRITELN('  1. PERFORM NEW CALCULATION              ');
 WRITELN('  2. QUIT NUTDEP                          ');
 WRITELN('                                          ');
 WRITELN('  GIVE YOUR CHOICE:                       ');
 WRITELN('                                          ');
 REPEAT
    GOTOXY(21,15);
    READLN(MENU2);
 UNTIL MENU2 IN [1..2];
END;


   {  MAIN PROGRAM NUTRIENT DEPLETION STUDY                }

   {  Structure:                                           }
   {  1. Initialization of variables                       }
   {  2. Two menus to be filled by operator                }
   {  3. Input of CHEMCOM-datafile                         }
   {  4. Input of LUS-datafile                             }
   {  5. Input of PRODDAT-datafile                         }
   {  6. Translation of area-information from PRODDAT to   }
   {     the output table                                  }
   {  7. Calculation of nutrient depletion                 }
   {     7.1.  IN-1  : Mineral fertilizers                 }
   {     7.2.  OUT-1 : Removal by products                 }
   {     7.3.  OUT-2 : Removal by crop residues            }
   {     7.4.  IN-2  : Manure                              }
   {     7.5.  IN-3  : Deposition                          }
   {     7.6.  IN-4  : N fixation                          }
   {     7.7.  OUT-5 : Erosion                             }
   {     7.8.  OUT-3 : Leaching                            }
   {     7.9.  OUT-4 : Denitrification                     }
   {     7.10. IN-5  : Sedimentation                       }
   {     7.11. Fallow                                      }
   {  8. Calculation of totals                             }
   {  9. Input of data in summary table                    }
   { 10. Output of detailed results                        }
   { 11. Output of results                                 }
   { 12. Menu 3                                            }

BEGIN
{---------------------------------------------------------}
{ 1. Initialization of variables                          }
{---------------------------------------------------------}
CLRSCR;
CNTRNMS:='BENIGAMBGHANGUINCDIVLIBEMALIMAUANIGENIGA'+
         'SENESIERTOGOBURKANGOCAMECARECHADCONGGABO'+
         'ZAIRBURUETHIKENYMADAMALAMAUSMOZAZIMBRWAN'+
         'SOMATANZUGANZAMBSUDABOTSLESOSWAZ';
CRPNMS:='WHEARICEMAIZBARLMILLSORGOTHCPOTASPOTCASS'+
        'OTHRPLANBEETCANEPULSVEGEBANACITRFRUIOILC'+
        'PALMSOYBGROUSUNFSESACOCNCOCOCOFFTEASTOBA'+
```

```
        'COTTFIBRRUBBFODDOTHR';
CONTINUE:=TRUE;
REPEAT { UNTIL CONTINUE=FALSE }
 CLRSCR;

{-------------------------------------------------------}
{ 2. Two menus to be filled by operator                 }
{-------------------------------------------------------}
 WRITELN;
 WRITELN('                                              ');
 WRITELN('                                              ');
 WRITELN('                                              ');
 WRITELN('                                              ');
 WRITELN('                                              ');
 WRITELN;
 WRITELN;
 TEXTBACKGROUND(7);
 TEXTCOLOR(0);
 WRITELN('COUNTRY :                                     ',
         '                  ');
 TEXTBACKGROUND(0);
 TEXTCOLOR(7);
 GOTOXY(1,11);
 MenuCntr; { Procedure }
 IF CNTRNR=100 THEN
    BEGIN
    ALL:=TRUE;
    CNTRNR:=0;
    END
 ELSE
    ALL:=FALSE;
 REPEAT  { UNTIL ALL=FALSE }
 IF ALL=TRUE THEN
    CNTRNR:=CNTRNR+1;
 Country; { Procedure }
 TEXTBACKGROUND(7);
 TEXTCOLOR(0);
 GOTOXY(11,9);
 WRITE('                          ');
 GOTOXY(11,9);
 WRITE(CNTR25);
 TEXTBACKGROUND(0);
 TEXTCOLOR(7);
 Menu_1; { Procedure }
 IF CNTRNR=38 THEN ALL:=FALSE;
 GOTOXY(1,11);
 FOR I1:=1 TO 12 DO
    DELLINE;
 WRITELN('Calculating.......');

{-------------------------------------------------------}
{ 3. Input of CHEMCOM-datafile                          }
{-------------------------------------------------------}
 IF MENU1=0 THEN
    BEGIN
    ASSIGN (IFL,(DIR+FN1));
    RESET (IFL);
    CHECK:=0;
    FOR CROPNR:=1 TO 35 DO
       BEGIN
```

```
      FOR I1:=1 TO 9 DO
          BEGIN
          READ(IFL,CC[CROPNR,I1]);
          IF (CC[CROPNR,I1]<0) OR (CC[CROPNR,I1]>999) THEN
             CHECK:=CROPNR*10+1;
          END;
      IF CROPNR<35 THEN READLN(IFL);
      FOR NPK:=1 TO 3 DO
         BEGIN
         IF (CC[CROPNR,(2+NPK*2)]>CC[CROPNR,(3+NPK*2)]) THEN
         CHECK:=CROPNR*10+2;
         END;
      END; { FOR CROPNR }
   CLOSE (IFL);
   IF CHECK>0 THEN
      BEGIN
      WRITELN('WARNING!!!!!!!!!!!!!!!!!!!!!!!!');
      WRITELN('INPUT IN CHEMCOM TABLE OUT OF RANGE ');
      WRITELN('INPUT CODE = ',CHECK:3);
      END;
```

```
{-------------------------------------------------------------}
{ 4. Input of LUS-datafile                                    }
{-------------------------------------------------------------}
    FOR LINE:=1 TO 200  DO
      FOR I1:=1 TO 5 DO
         LI[LINE,I1]:=0;
    ASSIGN(IFL,(DIR+FN2));
    RESET(IFL);
    LINE:=0;
    FOR LWC:=1 TO 6 DO
       BEGIN
       CHECK:=0;
       LINE:=LINE+1;
       FOR I1:=1 TO 4 DO
           READ(IFL,LI[LINE,I1]);
       READLN(IFL);
       IF (LI[LINE,1]<1) OR (LI[LINE,1]>6) THEN
           CHECK:=LWC*100+1;
       IF (LI[LINE,2]<0) OR (LI[LINE,2]>6) THEN
           CHECK:=LWC*100+2;
       IF (LI[LINE,4]<0) OR (LI[LINE,4]>3000) THEN
           CHECK:=LWC*100+3;
       IF (LI[LINE,4]<1) OR (LI[LINE,4]>3) THEN
           CHECK:=LWC*100+4;
       NOLUS:=(LI[LINE,2]);
       IF NOLUS>0 THEN
          BEGIN
          FOR LUS:=1 TO NOLUS DO
             BEGIN
             LINE:=LINE+1;
             READLN(IFL,LUSCODE[LWC,LUS],LI[LINE,2]);
             IF (LI[LINE,2]<1) OR (LI[LINE,2]>30) THEN
                 CHECK:=LWC*100+5;
             LI[LINE,1]:=LUS;
             NOCROPS:=(LI[LINE,2]);
             LINE:=LINE+1;
             READ(IFL,FERTFAC);
             LI[LINE,1]:=ROUND(FERTFAC*10);
             FOR I1:=2 TO 5 DO
                 READ(IFL,LI[LINE,I1]);
             READLN(IFL);
             IF (LI[LINE,1]<0) OR (LI[LINE,1]>30) THEN
                CHECK:=LWC*100+6;
             IF (LI[LINE,2]<0) OR (LI[LINE,2]>4) THEN
                CHECK:=LWC*100+7;
             IF (LI[LINE,3]<0) OR
                ((LI[LINE,3]>100) AND (LI[LINE,3]<>200)) THEN
                CHECK:=LWC*100+8;
             IF (LI[LINE,4]<0) OR (LI[LINE,4]>50) THEN
                CHECK:=LWC*100+9;
             IF (LI[LINE,5]<>1) AND (LI[LINE,5]<>2) THEN
                CHECK:=LWC*100+10;
             FOR I1:=1 TO NOCROPS DO
                BEGIN
                LINE:=LINE+1;
                READLN(IFL,LI[LINE,1],LI[LINE,2]);
                IF (LI[LINE,1]<1) OR (LI[LINE,1]>35) THEN
                   CHECK:=LWC*100+12;
                IF (LI[LINE,2]<1) OR (LI[LINE,2]>100) THEN
                   CHECK:=LWC*100+13;
```

```
                END; {  FOR I1   }
              END;   {  FOR LUS  }
           END;    { IF NOLUS>0   }
        END;   {  FOR LWC   }
     IF CHECK>0 THEN
        BEGIN
        WRITELN('WARNING!!!!!!!!!!!!!!!!!!!!!!!!!');
        WRITELN('INPUT IN LUS TABLE OUT OF RANGE ');
        WRITELN('INPUT CODE = ',CHECK:3);
        END;
     CLOSE(IFL);

{-------------------------------------------------------------}
{ 5. Input of PRODDAT-datafile                                }
{-------------------------------------------------------------}
     ASSIGN(IFL,(DIR+FN3));
     RESET(IFL);
     ASSIGN(OFL,(DIR+ON1));
     REWRITE(OFL);
     IF (CNTRNR>1) THEN
        FOR I1:=1 TO (74*(CNTRNR-1)) DO
           READLN(IFL);
     FOR YEAR:=1 TO 2 DO
        BEGIN
        CHECK:=0;
        READLN(IFL);
        FOR I1:=1 TO 35 DO
           BEGIN
           FOR I2:=1 TO 15 DO
              READ(IFL,PD[I1,I2]);
           READLN(IFL);
           END;
        FOR I1:=1 TO 2 DO
            READ(IFL,CI[1]);
        FOR LWC:=1 TO 6 DO
           FOR I1:=1 TO 2 DO
            READ(IFL,CI[LWC]);
        READLN(IFL);
        IF YEAR=2 THEN CLOSE(IFL);
```

```
{-------------------------------------------------------------}
{ 6. Translation of area-information from PRODDAT to          }
{    the output table                                         }
{-------------------------------------------------------------}
      FOR LINE:=0 TO 200 DO
        FOR I1:=1 TO 34 DO
           OT[LINE,I1]:=0;
      FOR LWC:=0 TO 6 DO
         FOR NPK:=1 TO 4 DO
            FALLOW[LWC,NPK]:=0;
      LINE:=0;
      FOR LWC:=1 TO 6 DO
         BEGIN
         TOTAREA:=0;
         LINE:=LINE+1;
         NOLUS:=LI[LINE,2];
         LWCLINE:=LINE;
         IF NOLUS>0 THEN
            BEGIN
            FOR LUS:=1 TO NOLUS DO
               BEGIN
               LINE:=LINE+1;
               NOCROPS:=LI[LINE,2];
               LINE:=LINE+1;
               FOR CROP:=1 TO NOCROPS DO
                  BEGIN
                  LINE:=LINE+1;
                  CROPNR:=LI[LINE,1];
                  OT[LINE,1]:=
                     (PD[CROPNR,(2+LWC*2)]/100)*LI[LINE,2];
                  TOTAREA:=TOTAREA+ROUND(OT[LINE,1]);
                  IF CI[LWC]>100 THEN
                     BEGIN
                     CASE CROPNR OF
                     1,2,3,4,5,6,7,8,9,15,16,22,23,24,25,30,34:
                          SUBAREA:=SUBAREA+ROUND(OT[LINE,1]);
                     END;  { CASE }
                     END;  { IF CI }
                  END; {  FOR CROP }
               END;  {  FOR LUS }
            IF (CI[LWC]=0) AND (TOTAREA>0) THEN CI[LWC]:=100;
            IF (CI[LWC]<100) AND (TOTAREA>0) THEN
             FALLOW[LWC,1]:=((100/CI[LWC])-1)*TOTAREA;
            IF CI[LWC]=100 THEN FALLOW[LWC,1]:=0;
            IF CI[LWC]>100 THEN
               BEGIN
               FALLOW[LWC,1]:=0;
               AREAFAC:=
                   SUBAREA/(SUBAREA+TOTAREA*(CI[LWC]-100)/100);
               LINE:=LWCLINE;
               NOLUS:=LI[LINE,2];
```

```
                FOR LUS:=1 TO NOLUS DO
                   BEGIN
                   LINE:=LINE+1;
                   NOCROPS:=LI[LINE,2];
                   LINE:=LINE+1;
                   FOR CROP:=1 TO NOCROPS DO
                      BEGIN
                      LINE:=LINE+1;
                      CROPNR:=LI[LINE,1];
                      CASE CROPNR OF
                      1,2,3,4,5,6,7,8,9,15,16,22,23,24,25,30,34:
                               BEGIN
                               OT[LINE,1]:=OT[LINE,1]*AREAFAC;
                               PD[CROPNR,(3+LWC*2)]:=ROUND

(PD[CROPNR,(3+LWC*2)]/AREAFAC);
                               END;
                      END;  { CASE }
                      END; { FOR CROPNR }
                   END;  { FOR LUS }
                END;  { IF CI }
             END;  { IF NOLUS>0 }
          END;  {   FOR LWC }

{-------------------------------------------------------------}
{ 7. Calculation of nutrient depletion                        }
{     7.1 IN-1  : Mineral fertilizers                         }
{-------------------------------------------------------------}
       FOR CROPNR:=1 TO 35 DO
           CRPFRT[CROPNR]:=0;
       LINE:=0;
       FOR LWC:=1 TO 6 DO
          BEGIN
          LINE:=LINE+1;
          NOLUS:=LI[LINE,2];
          IF NOLUS>0 THEN
             BEGIN
             FOR LUS:=1 TO NOLUS DO
                BEGIN
                LINE:=LINE+1;
                NOCROPS:=LI[LINE,2];
                LINE:=LINE+1;
                FOR CROP:=1 TO NOCROPS DO
                   BEGIN
                   LINE:=LINE+1;
                   CROPNR:=LI[LINE,1];
                   CRPFRT[CROPNR]:=
                       CRPFRT[CROPNR]+ROUND(OT[LINE,1]*
                       (LI[(LINE-CROP),1]/10));
                   END;  { FOR CROP }
                END;  {  FOR LUS  }
             END; { IF NOLUS>0  }
          END;  { FOR LWC  }
       LINE:=0;
       FOR LWC:=1 TO 6 DO
          BEGIN
          LINE:=LINE+1;
          NOLUS:=(LI[LINE,2]);
          IF NOLUS>0 THEN
             BEGIN
```

```
            FOR LUS:=1 TO NOLUS DO
               BEGIN
               LINE:=LINE+1;
               NOCROPS:=LI[LINE,2];
               LINE:=LINE+1;
               FOR CROP:=1 TO NOCROPS DO
                  BEGIN
                  LINE:=LINE+1;
                  CROPNR:=LI[LINE,1];
                  FOR NPK:=1 TO 3 DO
                     BEGIN
                     IF CRPFRT[CROPNR]>0.1 THEN
                        OT[LINE,(NPK*11-9)]:=
                           (PD[CROPNR,NPK]*
                           LI[(LINE-CROP),1]/CRPFRT[CROPNR])*10
                     ELSE
                        OT[LINE,(NPK*11-9)]:=0;
                     END;  { FOR NPK }
                  END;  { FOR CROP }
               END;  { FOR LUS }
            END;  {  IF NOLUS>0   }
         END;  { FOR LWC }

{-------------------------------------------------------}
{     7.2  OUT-1 Removal by products                    }
{-------------------------------------------------------}
       LINE:=0;
       FOR LWC:=1 TO 6 DO
          BEGIN
          LINE:=LINE+1;
          NOLUS:=LI[LINE,2];
          RAIN:=LI[LINE,3];
          SFERT:=LI[LINE,4];
          IF NOLUS>0 THEN
             BEGIN
             FOR LUS:=1 TO NOLUS DO
                BEGIN
                LINE:=LINE+1;
                NOCROPS:=LI[LINE,2];
                LINE:=LINE+1;
                MANU:=LI[LINE,2];
                RESI:=LI[LINE,3];
                MANA:=LI[LINE,5];
                BURNING:=FALSE;
                IF RESI=200 THEN
                   BEGIN
                   RESI:=0;
                   BURNING:=TRUE;
                   END;
                FOR CROP:=1 TO NOCROPS DO
                   BEGIN
                   LINE:=LINE+1;
                   CROPNR:=LI[LINE,1];
                   AREA:=ROUND(OT[LINE,1]);
                   YIELD:=PD[CROPNR,(3+LWC*2)];
```

```
                    FOR NPK:=1 TO 3 DO
                       BEGIN
                       IF AREA>0 THEN
                          UPTAKE[NPK]:=
                             (YIELD/10)*(CC[CROPNR,NPK]/10)
                       ELSE UPTAKE[NPK]:=0;
                       OT[LINE,(NPK*11-4)]:=UPTAKE[NPK];
                       IF CROP= 34 THEN
                          OT[LINE,(NPK*11-4)]:=0.8*UPTAKE[NPK];
                       END; { FOR NPK }

{-------------------------------------------------------}
{   7.3  OUT-2 : Removal by crop residues                }
{-------------------------------------------------------}
                    IF AREA>0 THEN
                       BEGIN
                       FOR NPK:=1 TO 3 DO
                          BEGIN
                          IF MANA=1 THEN
                             CHEMCONC:=CC[CROPNR,(3+NPK*2)]
                          ELSE
                             CHEMCONC:=CC[CROPNR,(2+NPK*2)];
                          UPTAKE[(3+NPK)]:=
                             (YIELD/10)*(CHEMCONC/10);
                          END;  { FOR NPK }
                       END    { IF AREA>0 }
                    ELSE
                       UPTAKE[(3+NPK)]:=0;
                    FOR NPK:=1 TO 3 DO
                       OT[LINE,(NPK*11-3)]:=
                             (RESI/100)*UPTAKE[(3+NPK)];

{-------------------------------------------------------}
{     7.4 IN-2   : Manure                               }
{-------------------------------------------------------}
                    CASE LWC OF
                       1,2,6 : BEGIN
                               OT[LINE,3]:=8;
                               OT[LINE,14]:=4;
                               OT[LINE,25]:=13;
                               END;
                       3     : IF RAIN<1200 THEN
                                  BEGIN
                                  OT[LINE,3]:=8;
                                  OT[LINE,14]:=4;
                                  OT[LINE,25]:=13;
                                  END
                               ELSE
                                  BEGIN
                                  OT[LINE,3]:=7;
                                  OT[LINE,14]:=3.5;
                                  OT[LINE,25]:=11;
                                  END;
                       4,5   : BEGIN
                               OT[LINE,3]:=7;
                               OT[LINE,14]:=3.5;
                               OT[LINE,25]:=11;
                               END;
```

```
                    END; { CASE LWC }
                    CASE MANU OF
                       0,1,2,3: FOR NPK:=1 TO 3 DO
                                   OT[LINE,(NPK*11-8)]:=
                                       OT[LINE,(NPK*11-8)]*
                                           0.5*MANU;
                       4      : BEGIN
                                  FOR NPK:=1 TO 3 DO
                                     OT[LINE,(NPK*11-8)]:=
                                        RESI*0.0045*UPTAKE[3+NPK];
                                  END;
                    END; { CASE  }
                    IF CROP=34 THEN
                       FOR NPK:=1 TO 3 DO
                          OT[LINE,(NPK*11-8)]:=
                              0.8*0.45*UPTAKE[3+NPK];

{-------------------------------------------------------}
{     7.5  IN-3 : Deposition                            }
{-------------------------------------------------------}
                    OT[LINE,4]:=0.14*SQRT(RAIN);
                    OT[LINE,15]:=0.053*SQRT(RAIN);
                    OT[LINE,26]:=0.11*SQRT(RAIN);

{-------------------------------------------------------}
{     7.6  IN-4 : N fixation                            }
{-------------------------------------------------------}
                    IF CROPNR IN [15,22,23] THEN
                       OT[LINE,5]:=0.6*(UPTAKE[1]+UPTAKE[4])
                    ELSE
                       OT[LINE,5]:=0;
                    IF CROPNR=2 THEN
                       IF LWC>4 THEN
                          OT[LINE,5]:=20;
                    CASE LWC OF
                       1: OT[LINE,5]:=OT[LINE,5]+3;
                       2: OT[LINE,5]:=OT[LINE,5]+4;
                       3: BEGIN
                          IF RAIN<1200 THEN
                             OT[LINE,5]:=OT[LINE,5]+2
                          ELSE
                             OT[LINE,5]:=OT[LINE,5]+5;
                          END;
                       4: OT[LINE,5]:=OT[LINE,5]+5;
                       5: OT[LINE,5]:=OT[LINE,5]+2;
                       6: OT[LINE,5]:=OT[LINE,5]+2;
                    END; { CASE }
                    OT[LINE,16]:=0;
                    OT[LINE,27]:=0;
```

```
{-------------------------------------------------------}
{     7.7  OUT-5 : Erosion                              }
{-------------------------------------------------------}
                    CASE SFERT OF
                       1: BEGIN
                          OT[LINE,11]:=LI[(LINE-CROP),4]*0.05*20;
                          OT[LINE,22]:=
                             LI[(LINE-CROP),4]*0.02*20*0.75;
                          OT[LINE,33]:=
                             LI[(LINE-CROP),4]*0.05*20*0.75;
                          END;
                       2: BEGIN
                          OT[LINE,11]:=LI[(LINE-CROP),4]*0.1*20;
                          OT[LINE,22]:=
                             LI[(LINE-CROP),4]*0.05*20*0.75;
                          OT[LINE,33]:=
                             LI[(LINE-CROP),4]*0.1*20*0.75;
                          END;
                       3: BEGIN
                          OT[LINE,11]:=LI[(LINE-CROP),4]*0.2*20;
                          OT[LINE,22]:=
                             LI[(LINE-CROP),4]*0.1*20*0.75;
                          OT[LINE,33]:=
                             LI[(LINE-CROP),4]*0.2*20*0.75;
                          END;
                    END; {CASE}

{-------------------------------------------------------}
{     7.8  OUT-3 : Leaching                             }
{-------------------------------------------------------}
                    OT[LINE,9]:=
                       2.3+(0.0021+SFERT*0.0007)*RAIN+
                       0.3*(OT[LINE,2]+OT[LINE,3])-
                       0.1*(UPTAKE[1]+UPTAKE[4]);
                    OT[LINE,20]:=0;
OT[LINE,31]:=
                       0.6+(0.0011+SFERT*0.002)*RAIN+
                       0.5*(OT[LINE,24]+OT[LINE,25])-
                       0.1*(UPTAKE[3]+UPTAKE[6]);
                    IF BURNING=TRUE THEN
                       OT[LINE,31]:=OT[LINE,31]+0.5*UPTAKE[6];
                    IF OT[LINE,9]<0 THEN OT[LINE,9]:=0;
                    IF OT[LINE,31]<0 THEN OT[LINE,31]:=0;
                    IF (CNTRNR=35) AND (SFERT=3) THEN
                       OT[LINE,31]:=0;

{-------------------------------------------------------}
{     7.9  OUT-4 : Denitrification                      }
{-------------------------------------------------------}
                     CASE LWC OF
                     1: OT[LINE,10]:=3;
                     2: OT[LINE,10]:=5;
                     3: OT[LINE,10]:=8;
                     4: IF RAIN<1200 THEN
                           OT[LINE,10]:=5
                        ELSE
                           OT[LINE,10]:=12;
                     5: OT[LINE,10]:=12;
                     6: OT[LINE,10]:=11;
                    END; {  CASE  }
```

```
                    OT[LINE,10]:=
                       OT[LINE,10]+2.5*SFERT+
                       0.3*OT[LINE,2]+0.3*OT[LINE,3]-
                       0.1*(UPTAKE[1]+UPTAKE[4]);
                    IF OT[LINE,10]<0 THEN OT[LINE,10]:=3;
                    OT[LINE,21]:=0;
                    OT[LINE,32]:=0;
                    IF BURNING=TRUE THEN
                       OT[LINE,10]:=OT[LINE,10]+UPTAKE[4];

{-------------------------------------------------------}
{   7.10  IN-5 : Sedimentation                          }
{-------------------------------------------------------}
                    FOR NPK:=1 TO 3 DO
                       BEGIN
                       CASE LWC OF
                       1,2,3,4 : OT[LINE,(NPK*11-5)]:=0;
                       5: BEGIN
                          OT[LINE,(NPK*11-5)]:=0;
                          FOR FLOW:=1 TO 4 DO
                             OT[LINE,(NPK*11-5)]:=
                                OT[LINE,(NPK*11-5)]-
                                OT[LINE,(NPK*11+FLOW-10)];
                          FOR FLOW:=6 TO 10 DO
                             OT[LINE,(NPK*11-5)]:=
                                OT[LINE,(NPK*11-5)]+
                                OT[LINE,(NPK*11+FLOW-10)];
                          IF OT[LINE,(NPK*11-5)]<0 THEN
                             OT[LINE,(NPK*11-5)]:=0;
                          END;
                       6: CASE NPK OF
                          1: OT[LINE,(NPK*11-5)]:=10;
                          2: OT[LINE,(NPK*11-5)]:=3;
                          3: OT[LINE,(NPK*11-5)]:=5;
                          END;  { CASE }
                       END;  { CASE }
                       END;  { FOR NPK }
                    END;  {  FOR CROP  }
                 END;  { FOR LUS  }
              END;  { IF NOLUS }

{-------------------------------------------------------}
{   7.11 Fallow                                         }
{-------------------------------------------------------}
           IF (FALLOW[LWC,1]>0.5) AND (LWC<>5) THEN
              BEGIN
              FALLOW[LWC,2]:=2*FALLOW[LWC,1];
              FALLOW[LWC,3]:=2*FALLOW[LWC,1];
              FALLOW[LWC,4]:=1*FALLOW[LWC,1];
              END
           ELSE
              FOR NPK:=1 TO 3 DO
                 FALLOW[LWC,(NPK+1)]:=0;
           FOR NPK:=1 TO 3 DO
              FALLOW[0,(1+NPK)]:=FALLOW[0,(NPK+1)]+
                                      FALLOW[LWC,(NPK+1)];
           END;  {  FOR LWC }
```

```
{-------------------------------------------------------}
{   8. Calculation of totals                            }
{-------------------------------------------------------}
      LINE:=0;
      FOR I1:=1 TO 34 DO
         OT[0,I1]:=0;
      FOR LWC:=1 TO 6 DO
         BEGIN
         LINE:=LINE+1;
         LWCLINE:=LINE;
         NOLUS:=LI[LINE,2];
         FOR I1:=1 TO 34 DO
            OT[LWCLINE,I1]:=0;
         IF NOLUS>0 THEN
            BEGIN
            FOR LUS:=1 TO NOLUS DO
               BEGIN
               LINE:=LINE+1;
               NOCROPS:=LI[LINE,2];
               FOR I1:=1 TO 34 DO
                  OT[LINE,I1]:=0;
               LUSLINE:=LINE;
               LINE:=LINE+1;
               FOR CROP:=1 TO NOCROPS DO
                  BEGIN
                  LINE:=LINE+1;
                  FOR NPK:=1 TO 3 DO
                     BEGIN
                     FOR FLOW:=1 TO 5 DO
                        OT[LINE,(11*NPK+1)]:=
                           OT[LINE,(11*NPK+1)]+
                           OT[LINE,(11*NPK+FLOW-10)];
                     FOR FLOW:=6 TO 10 DO
                        OT[LINE,(11*NPK+1)]:=
                           OT[LINE,(11*NPK+1)]-
                           OT[LINE,(11*NPK+FLOW-10)];
                     END; { FOR NPK }
                  OT[LUSLINE,1]:=OT[LUSLINE,1]+OT[LINE,1];
                  FOR I1:=2 TO 34 DO
                     OT[LUSLINE,I1]:=OT[LUSLINE,I1]+
                        OT[LINE,I1]*OT[LINE,1];
                  END;  { FOR CROP }
               FOR I1:=1 TO 34 DO
                  OT[LWCLINE,I1]:=OT[LWCLINE,I1]+OT[LUSLINE,I1];
               FOR I1:=2 TO 34 DO
                  IF OT[LUSLINE,1]>0 THEN
                     OT[LUSLINE,I1]:=
                        OT[LUSLINE,I1]/OT[LUSLINE,1]
                  ELSE
                     OT[LUSLINE,I1]:=0;
               END; { FOR LUS  }
            END;  { IF NOLUS }
         OT[LWCLINE,1]:=OT[LWCLINE,1]+FALLOW[LWC,1];
         FOR NPK:=1 TO 3 DO
            OT[LWCLINE,(11*NPK+1)]:=FALLOW[LWC,(NPK+1)]+
                                    OT[LWCLINE,(11*NPK+1)];
         FOR I1:=1 TO 34 DO
            OT[0,I1]:=OT[0,I1]+OT[LWCLINE,I1];
         END;  { FOR LWC }
{-------------------------------------------------------}
```

```
{-------------------------------------------------------}
{  9.  Input of data in summary table                   }
{-------------------------------------------------------}
       LINE:=0;
       FOR LWC:=1 TO 6 DO
          BEGIN
          LINE:=LINE+1;
          NOLUS:=LI[LINE,2];
          FOR NPK:=1 TO 3 DO
             BEGIN
             SUMMARY[(LWC*6+NPK*2+YEAR-8),1]:=ROUND(OT[LINE,1]);
             FOR FLOW:=1 TO 10 DO
                SUMMARY[(LWC*6+NPK*2+YEAR-8),(FLOW+1)]:=
                   ROUND(OT[LINE,(NPK*11+FLOW-10)]);
             SUMMARY[(LWC*6+NPK*2+YEAR-8),12]:=
                   ROUND(FALLOW[LWC,(NPK+1)]);
             SUMMARY[(LWC*6+NPK*2+YEAR-8),13]:=
                   ROUND(OT[LINE,(NPK*11+1)]);
             IF OT[LINE,1]>0 THEN
                SUMMARY[(LWC*6+NPK*2+YEAR-8),14]:=
                   ROUND(OT[LINE,(NPK*11+1)]/OT[LINE,1])
             ELSE
                SUMMARY[(LWC*6+NPK*2+YEAR-8),14]:=0;
             END;
          IF NOLUS>0 THEN
             BEGIN
             FOR LUS:=1 TO NOLUS DO
                BEGIN
                LINE:=LINE+1;
                NOCROPS:=LI[LINE,2];
                LINE:=LINE+1;
                FOR CROP:=1 TO NOCROPS DO
                   BEGIN
                   LINE:=LINE+1;
                   END;  { FOR CROP }
                END; { FOR LUS  }
             END;  { IF NOLUS }
          END;  { FOR LWC }
       FOR NPK:=1 TO 3 DO
          BEGIN
          SUMMARY[(NPK*2+YEAR+34),1]:=ROUND(OT[0,1]);
          FOR FLOW:=1 TO 10 DO
             SUMMARY[(NPK*2+YEAR+34),(1+FLOW)]:=
                ROUND(OT[0,(NPK*11+FLOW-10)]);
          SUMMARY[(NPK*2+YEAR+34),12]:=ROUND(FALLOW[0,NPK+1]);
          SUMMARY[(NPK*2+YEAR+34),13]:=ROUND(OT[0,(NPK*11+1)]);
          SUMMARY[(NPK*2+YEAR+34),14]:=
             ROUND(OT[0,(NPK*11+1)]/OT[0,1]);
          END;  { FOR NPK }
```

```
{-------------------------------------------------------}
{ 10. Output of detailed results                        }
{-------------------------------------------------------}
       WRITELN(OFL);
       WRITELN(OFL);
       WRITE(OFL,'RESULTS FOR ');
       WRITE(OFL,CNTR25);
       CASE YEAR OF
          1 : WRITELN(OFL,', 1983.');
          2 : WRITELN(OFL,', 2000.');
       END; { CASE }
       FOR I1:=1 TO 2 DO
           WRITELN(OFL);
       FOR NPK:=1 TO 3 DO
          BEGIN
          FOR I1:=1 TO 2 DO WRITELN(OFL);
          WRITELN(OFL,' ╔═══════════════╗ ');
          CASE NPK OF
              1: WRITELN(OFL,' ║   N-BALANCE   ║ ');
              2: WRITELN(OFL,' ║ P2O5-BALANCE  ║ ');
              3: WRITELN(OFL,' ║ K2O-BALANCE   ║ ');
          END;    { CASE NPK  }
          WRITELN(OFL,' ╚═══════════════╝ ');
          WRITELN(OFL);
          ONCE1:=FALSE;
          ONCE2:=FALSE;
          WRITELN(OFL,'     AREA |  FERT  MANU  DEPO  FIXA'
                    ,' SEDI |  PROD  RESI  LEAC  GASE  EROS |',
                    '  FALL | TOTAL');
          WRITELN(OFL,'          000 HA ');
          WRITELN(OFL,'* TOTALS');
          WRITE(OFL,'         ',SUMMARY[(NPK*2+YEAR+34),1]:6,
                  ' |');
          FOR FLOW:=1 TO 5 DO
             WRITE(OFL,SUMMARY[(NPK*2+YEAR+34),(FLOW+1)]:6);
          WRITE(OFL,' |');
          FOR FLOW:=6 TO 10 DO
             WRITE(OFL,SUMMARY[(NPK*2+YEAR+34),(FLOW+1)]:6);
          WRITE(OFL,' |',SUMMARY[(NPK*2+YEAR+34),12]:6,' |',
                        SUMMARY[(NPK*2+YEAR+34),13]:6,
                        ' (TON/COUNTRY)');
          FOR I1:=1 TO 2 DO
             WRITELN(OFL);
          LINE:=0;
          FOR LWC:=1 TO 6 DO
             BEGIN
             LINE:=LINE+1;
             NOLUS:=LI[LINE,2];
             WRITELN(OFL);
             CASE LWC OF
             1: WRITELN(OFL,'* LOW RAINFALL AREA');
             2: WRITELN(OFL,'* UNCERTAIN RAINFALL AREA');
             3: WRITELN(OFL,'* GOOD RAINFALL AREA');
             4: WRITELN(OFL,'* PROBLEM AREA');
             5: WRITELN(OFL,'* NATURALLY FLOODED AREA');
             6: WRITELN(OFL,'* IRRIGATED AREA');
             END; { CASE }
```

```
                WRITELN(OFL);
                WRITE(OFL,' TOTAL      ',
                          SUMMARY[(LWC*6+NPK*2+YEAR-8),1]:4,' |');
                FOR FLOW:=1 TO 5 DO
                   WRITE(OFL,SUMMARY[(LWC*6+NPK*2+YEAR-8),
                                (FLOW+1)]:6);
                WRITE(OFL,' |');
                FOR FLOW:=6 TO 10 DO
                   WRITE(OFL,SUMMARY[(LWC*6+NPK*2+YEAR-8),
                                (FLOW+1)]:6);
                WRITE(OFL,' |',SUMMARY[(LWC*6+NPK*2+YEAR-8),12]:6,
                          ' |',SUMMARY[(LWC*6+NPK*2+YEAR-8),13]:6);
                IF ONCE1=FALSE THEN WRITE(OFL,' (TON/LWC)');
                WRITELN(OFL);
                WRITELN(OFL);
                IF NOLUS>0 THEN
                   BEGIN
                   FOR LUS:=1 TO NOLUS DO
                      BEGIN
                      LINE:=LINE+1;
                      NOCROPS:=LI[LINE,2];
                      CASE LWC OF
                      1: WRITELN(OFL,'LR-',LUSCODE[LWC,LUS]);
                      2: WRITELN(OFL,'UR-',LUSCODE[LWC,LUS]);
                      3: WRITELN(OFL,'GR-',LUSCODE[LWC,LUS]);
                      4: WRITELN(OFL,'PR-',LUSCODE[LWC,LUS]);
                      5: WRITELN(OFL,'NF-',LUSCODE[LWC,LUS]);
                      6: WRITELN(OFL,'IR-',LUSCODE[LWC,LUS]);
                      END; {CASE}
                      WRITE(OFL,' TOTAL  ',OT[LINE,1]:7:0,' |');
                      FOR FLOW:=1 TO 5 DO
                         WRITE(OFL,
                            ROUND(OT[LINE,(NPK*11+FLOW-10)]):6);
                      WRITE(OFL,' |');
                      FOR FLOW:=6 TO 10 DO
                         WRITE(OFL,
                            ROUND(OT[LINE,(NPK*11+FLOW-10)]):6);
                      WRITE(OFL,' |       |',
                                      ROUND(OT[LINE,(11*NPK+1)]):6);
                      IF ONCE1=FALSE THEN WRITE(OFL,' (KG/HA)');
                      WRITELN(OFL);
                      WRITELN(OFL,
                ' --------------+-----------------------------',

'-+-------------------------------+-------+-------');
                      LINE:=LINE+1;
                      FOR CROP:=1 TO NOCROPS DO
                         BEGIN
                         LINE:=LINE+1;
                         IF ROUND(OT[LINE,1])>0 THEN
                            BEGIN
                            CRP4:=COPY(CRPNMS,(LI[LINE,1]*4-3),4);
                            WRITE(OFL,' ',CRP4,':    ',
                                      ROUND(OT[LINE,1]):6,' |');
                            FOR FLOW:=1 TO 5 DO
                               WRITE(OFL,
                               ROUND(OT[LINE,(NPK*11+FLOW-10)]):6);
                            WRITE(OFL,' |');
                            FOR FLOW:=6 TO 10 DO
```

```
                          WRITE(OFL,
                          ROUND(OT[LINE,(NPK*11+FLOW-10)]):6);
                       WRITE(OFL,' |           |',
                          ROUND(OT[LINE,(11*NPK+1)]):6);
                       IF ONCE1=FALSE THEN WRITE(OFL,
                          ' (KG/HA)');
                       WRITELN(OFL);
                       END; { IF AREA>0 }
                    END; {  FOR CROP   }
                 WRITELN(OFL);
                 IF ONCE1=FALSE THEN ONCE1:=TRUE;
                 END;  {  FOR LUS   }
              IF FALLOW[LWC,1]>0 THEN
                 BEGIN
                 WRITE(OFL,' FALLOW: ',FALLOW[LWC,1]:6:0,
                 ' |                                 |',
                 '                                 |');
                 CASE NPK OF
                     1,2: WRITE(OFL,'      2');
                     3  : WRITE(OFL,'      1');
                 END;    { CASE }
                 WRITE(OFL,' |');
                 IF ONCE2=FALSE THEN
                      WRITE(OFL,'       (KG/HA)');
                 WRITELN(OFL);
                 ONCE2:=TRUE;
                 END;
              WRITELN(OFL);
              END;  { IF NOLUS>0 }
           END;  {    FOR LWC   }
        WRITELN(OFL);
        END; { FOR NPK }
     END;    { FOR YEAR:=1 TO 2 }
  CLOSE(OFL);

{---------------------------------------------------------}
{ 11. Output of results                                   }
{---------------------------------------------------------}
  ASSIGN(OFL,(DIR+ON2));
  REWRITE(OFL);
  WRITELN(OFL,CNTR25);
  FOR I1:=1 TO 4 DO
     WRITELN(OFL);
  WRITELN(OFL,'Figure 3.  Nutrient balance for ',CNTR25,
          ', 1983 (in kg/ha,yr)');
  FOR I1:=1 TO 45 DO
     WRITELN(OFL);
  WRITELN(OFL,'Table 3.   LWC-distribution in ',CNTR25,
          ' (in % of total arable area)');
  WRITELN(OFL,'──────────────────────────────',
          '────────────────────────────────');
  WRITELN(OFL,'                       LR       UR    ',
          '  GR       PR        NF       IR');
  WRITELN(OFL,'                    ────────────────────',
          '──────────────────────────');
  WRITE(OFL,'1983            ');
  FOR LWC:=1 TO 6 DO
     WRITE(OFL,(SUMMARY[(LWC*6-5),1]*100/SUMMARY[37,1]):8:0);
  WRITELN(OFL);
  WRITE(OFL,'2000            ');
```

```
   FOR LWC:=1 TO 6 DO
      WRITE(OFL,(SUMMARY[(LWC*6-4),1]*100/SUMMARY[38,1]):8:0);
   WRITELN(OFL);
   WRITELN(OFL,'-------------------------',
               '-----------------------------------------------------');
   WRITELN(OFL);
   WRITELN(OFL);
   WRITELN(OFL);
   WRITELN(OFL);
   WRITELN(OFL,'Table 3.   ',CNTR25,'; nutrient balances for '
               ,'LWC s and for country total');
   WRITELN(OFL,'-----------------------------------',
               '-----------------------------------');
   WRITELN(OFL,'@@AREA|',
               '@FERT@MANU@DEPO@FIXA@SEDI@|',
               '@PROD@RESI@LEAC@GASE@EROS@|',
               '@FALL@|@TOTAL');
   WRITELN(OFL,'@@000 HA@@@@@@@@@@@@@@@@@(TON)@(KG/HA)');
   FOR LWC:=1 TO 7 DO   { 7=TOTALS }
      BEGIN
      CASE LWC OF
         1: WRITELN(OFL,'* LOW RAINFALL');
         2: WRITELN(OFL,'* UNCERTAIN RAINFALL');
         3: WRITELN(OFL,'* GOOD RAINFALL');
         4: WRITELN(OFL,'* PROBLEM AREA');
         5: WRITELN(OFL,'* NATURALLY FLOODED');
         6: WRITELN(OFL,'* IRRIGATED AREA');
         7: WRITELN(OFL,'* TOTAL ');
      END; { CASE }
      FOR NPK:=1 TO 6 DO
         BEGIN
         CASE NPK OF
            1: WRITE(OFL,'@N@1983');
            2: WRITE(OFL,'@@2000');
            3: WRITE(OFL,'@P205@1983');
            4: WRITE(OFL,'@@2000');
            5: WRITE(OFL,'@K20@1983');
            6: WRITE(OFL,'@@2000');
         END; { CASE }
         WRITE(OFL,'@',SUMMARY[(LWC*6-6+NPK),1]:0,'@|');
         FOR I1:=1 TO 5 DO
            WRITE(OFL,'@',SUMMARY[(LWC*6-6+NPK),(I1+1)]:0);
         WRITE(OFL,'@|');
         FOR I1:=6 TO 10 DO
            WRITE(OFL,'@',SUMMARY[(LWC*6-6+NPK),(I1+1)]:0);
         WRITE(OFL,'@|@',SUMMARY[(LWC*6-6+NPK),12]:0);
         WRITE(OFL,'@|@',SUMMARY[(LWC*6-6+NPK),13]:0);
         WRITE(OFL,'@',SUMMARY[(LWC*6-6+NPK),14]:0);
         WRITELN(OFL);
         END; { NPK }
      WRITELN(OFL);
      END; { LWC }
   WRITELN(OFL,'-------------------------------',
               '-----------------------------------------');
   CLOSE(OFL);
   END;  { IF MENU1  }
UNTIL ALL=FALSE;
```

```
{-------------------------------------------------------------}
{  12. Menu 2                                                 }
{-------------------------------------------------------------}
 MENU_2;
 IF MENU2=1 THEN
    CONTINUE:=TRUE
 ELSE
    CONTINUE:=FALSE;
 UNTIL CONTINUE=FALSE;
END.
```

4. EVALUATION

Two programmes were necessary to carry out this study. The first, TRANSDAT.PAS, served to bring back the huge amount of data to a data set that met the requirements for this study. This program enabled us to perform the analysis of the second program, NUTDEP.PAS, on a personal computer. NUTDEP.PAS was written during the realization of the model. It was useful to program the model during the study, because the impact of changes in the model could be judged right away.
TRANSDAT and NUTDEP were programmed in Turbo Pascal, because the main author of this study was quite familiar with it.