COMPUTER ASSISTED DIAGNOSIS

(A Case Study of Animal Production)

BY

AMINA ALHASSAN MOHAMMED B.TECH (AGRIC)

MARCH, 1994

COMPUTER ASSISTED DIAGNOSIS
(A Case Study of Animal Production)

BY

AMINA ALHASSAN MOHAMMED (B.Tech Agric)

A project in the Department of Mathematics/Statistics/ Computer Science submitted to the Post Graduate School in partial fulfilment of the requirement for the award of Post Graduate Diploma in Computer Science of the Federal University of Technology, Minna, Nigeria.

MARCH, 1994

CERTIFICATION

I certify that this work was carried out by Amina Alhassan Mohammed of the Post Graduate School, Department of Maths/Statistics/ Computer Science, F.U.T. Minna, Niger state.

AMINA ALHASSAN MOHAMMED (STUDENT)	DATE
MR. SADIQ Z. UMAR (SUPERVISOR)	DATE
DR. K. R. ADEBOYE (HEAD OF DEPARTMENT)	DATE
EXTERNAL EXAMINER	DATE

CONTENTS

CHAPTER FOUR

CHAPTER FIVE

DEDICATION

This project report is dedicated with honour nd praise to the Almighty Allah and my sweetheart Ifeanyi Okoroigwe.

ACKNOWLEDGEMENT

First of all, I give great honour and adoration to the Allah for his unfailing love and mercy over me throughout my academic struggles.

My very special thanks also go to Mallam Sadiq Umar for the attention given me during the course of the project.

I am inexpressibly indebted to my sweetheart Ifeanyi Okoroigwe who in the most of his highly financial demands has given me the maximum support and attention throughout the period of this course.

I am also indebted to the following people who in one way or the other contributed to the completion of the project. Miss Kuburat Jimoh, Margaret Sadiq, Mr. and Mrs. Agbachi, Hajiya Hauwa, Rabi Zaki, Shola Thanni, Tunde Thanni and Habiba Alhassan.

Also to my colleagues in the struggle, Adamu Umar, Abdul Ibrahim, Mallam Andrew Idris, Ben Ako and Temisan Ogisi.

ABSTRACT

This project is solely on applying Expert system to the noses, medical applications and prescriptions to livestock ases. This work is in two parts; The consultation process of l disease diagnoses, and the personal data storage of each al that has undergone such consultation.

CHAPTER ONE

1.0 INTRODUCTION

Computers and their many applications have brought with them both economic and social changes. Computers are the most significant technological developement of the century.Computer literacy involves recognising and using computer as a problem solving tool and potentials for improving the quality of life.

Winston,1979. comments that computers can do many things that require intelligence. They can solve many problems like experts,reason geometrically, solve problems in mathemaatic e.t.c Today, with the growing interest in expert system and the momentum of the Japanese fifth generation programmes, there is an increasing recognition of the scope of Intelligent Artifacts.

Expert systems are the most significant practical implementation to date of Artificial Intelligence. Expert system is used in diagnosis of disease in the medical fields, including veterinary aspects of it.

Diagnosing disease in domestic livestock can be accomplished by several means. The easiest method is visual observation of the animal for obvious external signs of infection. Disease diagnosis can also be caried out in the laboratory using biological and chemical test.

The disease underreview are highly contagious in nature and of great economic or public importance which already occur or have occured in the recent past in Nigeria.

These diseases require urgent reporting and stringent control. Example are Rinderpest, contagious Borine , Plural Pnemonia, Anthrax etc .

The causes of diseases may be classified as follows :

1. Living agents e.g Bacteria, fungi, virus.
2. Nutritional deficiencies.
3. Genetic abnormalities.
3. Upsets in metabolism.
4. Poisonous substances.

1.1 AIM AND OBJECTIVE OF CAD

The name CAD is derived from COMPUTER ASSISTED DIAGNOSIS. It is designed to provide information to all those who are interested in working with computer to diagnose livestock disease.

Its major objective is to help overcome the hard task involved in going through text books before prescribing a drug to an infected animal. With the aid of this software, this can be done easily.

1.2 HISTORY OF ARTIFICIAL INTELLIGENCE

Artificial intelligence is a new and existing technology, the idea can be found atleast as far back as the pre-Christian Greeks.

Homer was acquainted with the concept of Artificial intelligence, perhaps 800 years B.C., and the ancient historian Polybios believed that Nabis, a dictator of Sparta (about 200 B.C.), used a robot to compel rich but citizens to pay their taxes.

In 1879, Villiers de l'Isle Adam, depicts a remarkable woman animated by electricity.

In fourth century A.D. a golden Buddhist was said to sit on a carriage and tended by animated models of Taoist monks. As the carriage moved, the monks circled the buddha, variously bowing and saluting and throwing incense into a censer.

In the seventh century, boats were constructed with animated figures.

In the eigth century Chinese records depict the mechanical figure of a monk which reached out saying 'Alms ! Alms !', conveying coins from its hand into a satchel.

In 790 A.D. a wooden otter was deviced in China, which was said to catch fish.

In 890, a wooden cat, was constructed and was able to catch rats and dance tiger-flies.

In the eighteenth century, Switzerland craftsmen (such as

Pierre, Hanri-Louis and Jaquet-Droz) devised automata that could write, draw pictures and play musical instruments.

In the nineteenth century a variety of talking machines were developed. e.g Euphoria, a 'bearded Turk' exhibited in Egyptian hall in Picadilly. The device could ask and answer questions, laugh, whisper and sing.

In 1932 robots were built for London radio exhibition. These robots could speak, smoke cigars and read newspapers.

The festival plaza of expo '70 in Osaka witnessed a gigantic robot that caried flashing lights and moved its head.

Charles Babbage, born in 1792, is often represented as the " father of modern computing ". As such he is also a progenitur of artificial intelligence, though this is rarely said.

Hershel and Babbage created the royal astronomical society in the 1820 's and this was able to compile reference tables.

The tabulator designed by Dr. Herman Hollerith as an effective device for analysing the 1890 American census, was the first computering machine to use non mechanical processing means.

In 1892 William Buroughs introduced the first commercially available adding machine, but it was not till the early 1940's that it proved possible.

John Von Newmann, (1945), designed the Electronic Discrete Variable Automatic Computer (EDVAC). This leads to the stored program control incorporated into the design of an electronic digital computer.

In 1940 the emergence of Electronic, Numerical Integrator and Calculator (ENIAC), designed with 18,000 thermionic values to compute the ballistic table for guns and missiles. Mavin Minsky, and Claude Shamon worked at bell laboratory. The work was to stimulate AI developement under project MAC at the Massachusetts Institute of Technology. He is a cofounder of the MIT AI group which later became the MIT AI laboratory.

In the 70's Edward Feigenbaum, at Stanford, developed the first expert system. Also at Stanford Dendral, they used to analyse mass spectography data .

Professor Terry Winogard, also at Stanford, produced a program (called SHRDLU) which was able to manipulate simulated objects shaped like wooden blocks.

Today many large companies-IBM, Hewlett-Packard, Digital Equipment Corporation, Tektronix, Fujitsu Hitachi, etc have set up AI research laboratories and important research is been conducted at many institutes and universities.

Increasing commercial emphasis is being given to the developement of particular expert systems for specific purpose. (medical diagnosis, crop disease diagnosis, geological prospecting, electronic circuit analysis, chemical synthesis etc.).

1.1.4. EXPERT SYSTEMS

Expert systems are often regarded as representing a subclass of artificial intelligence. Expert systems are one of the key developements contributing to the international fifth-generation programme.

Addis, 1982. review that the range of expert systems represent degrees of enhancement to an information retrieval system.

An expert system can be regarded as a means of recording and assessing human competence in a particular specialist field.

Duda et al, 1980. Suggests that an expert system is capable of human like performance and can serve thereby as a replacement expert. Some of the most successful expert systems are as follows :

1. MYCIN and INTERNIST (for medical diagnosis).
2. DENDRAL and SECS (for chemical analysis).
3. PROSPECTOR (for geological prospecting).
4. GPS (general problem solver)..

4.1.5. FEATURES OF EXPERT SYSTEMS

Expert systems have defined as (quoted by d'Agapeyeff, 1983).

> ' Problem solving program that solve substantial problems generally

> conceded as been dificult and requiring expertise.
>
> They are called knowledge based because there performance depends critically on the use of facts and heuristics used by experts "

The british computer society's comittee of the specialist group on expert systems has defined an expert system as :

> ' The embodiment within a computer of a knowledge-based component from an expert skill in such a form that the machine can offer intelligent advice or take an intelligent decision about a processing function. A desirable additional characteristic, which many would regard as fundamental, is the capability of the system on demand to justify its own time of reasoning in a manner directly intelligible to the enquirer. The style adopted to attain this characteristics is rule-based programming.'

Expert systems represent a flexible approach to computer competence, drawing on specialist knowledge and exploiting various types of inference (not only deductive reasoning).

CHAPTER TWO

2.0 FEASIBILITY STUDY

Feasibility study is the process of carrying out preli minary investigation of a system or an organisation.

2.1 Study of the existing system

The existing procedures of diagnosing animal diseases are as follaws.

1. Physical examination of the animals
2. Clinical examination
3. Post-mortem test

The above method are usually carried out manually by referring to test.

In the past, when animals fall sick, there are no scientific means of examing them. All examination are based primarily on physical observations. The results from this method is quite limited. Desipte the limitation it is still practiced today even by veterinary officer. Since it give a pre-knowledge of what is wrong.

After phyiscal examination. Clinical test which either microbiological or chemical are carried out.

In the event of lossing a livestock the tendency in the past is to depatched it by burring. This practice leads to a loss of knowledge as to the case of death. With the advant of scientific

development carcass is not disposed immdiately. Rather a general post-mortem test is carried out to asserten the cause of death. This knowledge gained help in the small measure to assist the veterinary or stockman in dealing with such occurance in the furture.

2.2 PROBLEM INDENTIFICATION

Since the animal can not talk, the stockman must be conversant with the behavour of animals. Any change in normal behavour is an indication of ill-health. One of the problem associateed with physical observation of diseases in livestock is that some of the diseases have similar symptoms. As such it is usually difficult to indentify which particular disease an animal is suffering from. At times it is when the disease is at an advanced stage that the symptom normally mainfest.

CHAPTER THREE

SYSTEM ANALYSIS

System analysis is the process of gathering facts, interpreting facts and using the information to recommed improvement to a system. Listed below are the facts obtained from the invetigation.

3.1 Analysis of the investigation

A Physical observation. It is the responsibility of the stockman to know th signs of health and should develop a keen eye to prompt spot an obviously sick animal. signs of illhealth are as follows.

1· The genaral posture of the animal, it's movement and behaviour will change in case of illness. Animal standing with head down or showing undue weariness or a tendency to seperate from the herd are warning symptoms.

2· Healthy animals eat greedily. Loss of appetite and stoppage of rumination are early signs of several diseases.

3. Muzzle and nostrils of healthy animals will be moist and free from any discharges. The muzzle will be dry in animals having high temperature.

4. Variation in body temperature can be measurd by inserting a clinical thermometer into the rectum of an animal for half a minute. High temperature are usually associated with increased activity of the body in fighting with disease. The animal body temperature of different livestock are given below.

Cattle	:38.6 degree centigrade
Sheep	:38.9 degree centigrade
Goat	:39.1 degree centigrade
Pigs	:39.2 degree centigrade
Buffaloes	:36.8 - 39.4 degree cent

5. Variation in pulse rate reflects the rate at which the heart pumps blood through the body. This can be measured by placing the index or second finger on arteries where they pass near the surface of the body.

Normal pulse rate

Cattle	:50-60 beats per min
Buffaloes	:40-50 " " "
Sheep/goat	:70-80 " " "
Pig	:70-90 " " "

6. The eyes in healthy animals are bright and alert. Sunken eye with a fixed staring eyes often accompany the onset of fever. discharges from both the eyes indicate systemic ailment.

7. Dung of healthy cows should be semi solid inconsistency, rich green in colour and free from gas bubbles or blood clots.

8. Changes in quality and quantity of milk yeild is one of the early symptom in several disease. blood and clot in milk indicate mastitis.

B Post-mortem test

This test is carried out to determined the cause of the illness. It is usually carried out by cutting the death animal open to examing the organs for any abnormality or discolouration of the organs. The operation is done the veteri nary officer or by the stockman.

C Microbiological Test

It is usually carried out by taking blood sample of the death animal to the laboratory for analysis. The analysis is carried out by the lab -technologist. After which the result are sent back to the farm.

3.2 Main objectives of the system analysis

Most cases diagnosis is done by referring to text couple with experiences. However it is the intention of this project to produce a diagnosetic system, after the symptoms are known, using the computer, there by enhencing the following;

1 Reduction in time comsumed in going through text.

2 Provides on demand a ready diagnosis for the system found.

3 Creating a file for each symptom to with it's coresponding treatment.

4 Providing a comprehensive case history of system under review.

3.3 Information accessibility

The information regarding this project were obtained from the following,

1 Expert, this includes the veterinary and the stockman who manages the livestock on the farm

2 Book or textual file.

3 Direct experience of the programmer.

CHAPTER FOUR

4.0 System Design

System design is the process of planning a new system to replace or to compliment an existing system.

4.1 Aim of the system design

This system is designed to provide information to all those who are interested in working with computer on there farm to diagnose livestock diseases and also to reduce the cost involved in inviting a veterinary doctor to the farm on regular bases.

4.2 Tools required for the system design are as follows :

1: Veterinary doctor or officer.

2: Stock-man.

3: Data collected from stock-man on the various records syntoms and treatment.

4: Computer 80386 IBM (Compatible).

5: Line printer.

4.3 Cost implication of new system

1: Cost of hardware/software

Processor	80286/80386	=N= 80,000.00
RAM	1.2 MB	
Diskette drive/slot		
(a) 3.5	1.44 MB	
(b) 5 1/4	1.2 MB	
Hard disk	30 OR 40 MB	

Printer	24 pin line printer	=N= 17,000.00
Operating system	MS-DOS 4.1	=N= 5,000.00
UPS 500 Volts		=N= 5,000.00
Developement cost		=N= 35,000.00
Installation cost		=N= 10,000.00
Salary of staff involved		=N= 12,000.00
Staff training cost		=N= 10,000.00
Operating cost		
(a) Paper		=N= 2,000.00
(b) Ribbon		=N= 3,000.00
(c) Diskette		=N= 1,000.00
(d) Maintenance		=N= 600

4.4 System specification

Hardware requirement

The hardware requirement needed to run this software consists of an IBM PC or compatible with a minimum of 256 KB of memory.

Recommended hardware configuration are :

(1) An IBM PC/AT or 100% compatible computer.

(2) 256 KB of RAM minimum.

(3) An IBM BIOS or compatible and keyboard.

Software requirement

The operating system for running this system is the disk operating system, system disk are contained in the disks and are accessed through using command. Secondly, one diskette containing the DBASE III+ (Software) and all the necessary files, this diskette has to be inside the drive at all times during the running of the program.

4.5 Analysis of results

Justification of the new system

It has become important for livestock farmers to have a computer based system for diagnosing diseases for the purpose of efficiency. This system will satisfy a growing need for improved management.

This system is meant for livestock farmers, schools and universities etc. With this system accurate diagnosis can be made, hence appropriate and proper treatment and preventive measures can be applied, invariably reducing the mortality and outbreak of disease.

4.6 System requirement

The requirement for this system are as follows :

1: Increased speed.

2: Large and secured storage facilities.

3: Need for accuracy.

4: Quick accessibility of information.

5: Automation.

These features are incoporated into system to enhance the performance of the new system.

4.7 Designhned normalized data base file

(A) Symptom data

1. Symptom
2. All code

(B) Sickcure data

1. All code
2. Disease
3. Treatment
4. Control
5. Causal agent

(C) Personal data

1. Date
2. Ear tag number
3. Age
4. Sex
5. Disease
6. Treatment

4.8 Structure for symptoms.DBF

FIELD	FIELDNAME	TYPE	WIDTH	DEC
1	symptom	c	40	0
2	symptom	c	2	0

Structure for sickcure.DBF

FIELD	FIELDNAME	TYPE	WIDTH	DEC
1	all code	c	2	0
2	disease	c	15	0
3	treatment	c	40	0
4	control	c	30	0
5	causal-agent	c	20	0

Structure for personnal.DBF

FIELD	FIELDNAME	TYPE	WIDTH	DEC
1	date	D	8	2
2	eartag-number	C	6	0
3	age	N	3	2
4	sex	C	6	0
5	disease	C	15	0
6	treatment	C	30	0

4.9 Modular program structure (Procedure)

In modular structure, each process has its duty to perform effectively in the propossed system. The duties are as follows :

Main menu

This is the program that displays the main menu of the system. It consists of the following options :

1. Consultation
2. Update knowledge base
3. Delete knowledge
4. Edit knowledge
5. Exit

These options are sub-programs. They are displayed for execution.

Consultation

This section is concerned with the consultation. The existing symptoms in knowledge base are displayed for the user to select. The symptom that matches what is observed on the animal. After the selection the causal agent responsible for illness is displayed together with disease, treatment and control.

Update knowledge base

Update knowledge base consists of six options. The options are as follows :

1. Update symptoms
2. Update disease
3. Update treatment

4. Update control

5. Update causal agent

6. Exit

Any of the option can be updated, from time to time by insertion of new date.

DELETE KNOWLEDGE

This section is concerned with deletion of any particular part of the knowledge base that is not required. It consists of the following options.

1. Delete symptoms

2. Delete disease

3. Delete treatment

4. Delete control

5. Delete causal agent

6. Exit

Edit knowledge

As the name implies, it is concerned with the editing of knowledge. Edit knowledge has a menu. The menu consists of the following options :

1. Edit symptoms

2. Edit disease

3. Edit treatment

4. edit control

5. Edit cause agent

6. Exit

Personal data

As the name implies, it contains the personal data of the animal that is been diagnosed. This includes the animal type, ear tag number, age, sex, disease, treatment and finally the date.

CHAPTER FIVE

5.0 System implimentation

The final step in system developement is the system implimentation. In this step written programs are used, after which the programs are tested to ensure their correctness, The staff that will be using the system is trained, any data existing are taken from the old system and converted to the new system and lastly the new system is installed.

5.1 Programming (Written program)

The program use the files created in the system application in chapter three and knowledge base. This have been successfully written using querry language and modular structure. The program is shown in the appendix.

5.2 Installation

Installation of this system can be done by computer experts. The procedure given below can be used.
The hardware facillities required are IBM PC/AT and printer. Diskette containing the system is also required. The system can two floppy diskettes will do.
Procedure for installation

Boot your system before you install CAD.

1. insert the disk labelled "CAD" in drive A: and close the disk drive door.
2. Create a subdirectory called CAD in disk drive c:
3. Type A>COPY *.* C:>\CAD
4. Follow the instruction on the screen to enter your name and your organisation.
5. Choose the continue botton.
6. Follow the instruction on the screen to enter the directory where you want to install CAD.
7. Choose the continue button.

5.3 System testing

Every program must be tested before it can be used for production runs. Program testing determines the reliability of the program. System testing involes two kinds of activities;integretion and acceptance testing.

Acceptance testing involes planning and execution of function tests.Perfomance tests,and stress test to verify that the inplemented system satisfies its requirements.

Integration testing. Botton-up integration is the traditional strategy used to integrate the components of the software system into a functional whole. Button-up integration consists of unit testing, followed by subsystem testing, followed by testing of the entire system.

In general, if DBASE III+ files are not available on the machine to be used then you must already have had it on two floppy diskettes labelled I and II. However, if your computer already has DBASE III+ then after booting the system you will get the c prompt written thus :

C>

1. Then type

C>CD DBASE

C>DBASE

After doing this, you have completed loading DBASE III+. You should be ready to run the system.

2. Press ESCAPE key and the system takes you to the dot prompt.

3. Put the diskette containing the system in drive A and type

.Set default to A

.Do main

A message will appear on the screen as shown below

```
   COMPUTER ASSISTED DIAGNOSIS
   CASE STUDY ANIMAL PRODUCTION
            BY
AMINA ALHASSAN MOHAMMED
FOR THE AWARD OF POST GRADUATE
DIPLOMA IN COMPUTER SCIENCE

PRESS ANY KEY TO CONTINUE
```

4. After this the main menu appears and the system is ready to execute any of the chosen option/choice.

4. However, in a situation where the user has DBASE III+ on floppies(diskettes) then the following procedures should be

llowed.

ter booting the system and getting the system prompt

t the system disk I and change the drive by typing

A:

DBASE

ter loading the files from this diskettes, the computer will
k you to insert the system disk II and press the ENTER key. On
ing this, you have succeeded in loading DBASE II+ files. Press
CAPE to receive the dot prompt, then type

O CAD

he steps 4-5 above are repeated.

.4 Conversion

The change over from the old system to the new one is known
s conversion. Parallel approach is chosen for this system.
arallel approach means using both the new and the old system
ogether.

.5 system maintenance

The term "Software maintenance" is used to describe the
oftware engineering activities that occur following the delivery
f a software product to the customer.
aintenance activities involve making enhancement to
oftware product, adapting products to new environments and
odifying the software to suit the new environment.

Adaptation of software to a new environment may involve moving the software to a different machine, or for instance, modifying the software to accommodate new additional modules.

5.7 REFERENCES

1. Abdulkadir I. A. (1980)
   "Infectious diseases of livestock in Nigeria. "

2. Okoroigwe Ifeanyi Uzoma. (1991)
   "Computer assisted learning of computer fundamentals (CALFUND)."

3. G.L. Simons. (1984), "Introduction to artificial intelligence."

4. Steven L. Taiumoto. (1983), "Element of artificial intelligence."

5. McNitt J. I. (1981), "Livestock husbandry techniques."

6. Elaine Rich. (1983), "Artificial intelligence."

7. Richard Fairley. (1985), "Software engineering concept."

5.8

APPENDIX 1

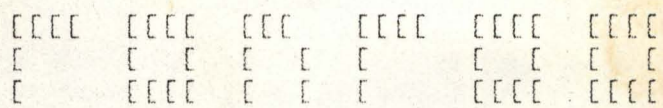
DEFINITION OF TERMS

APPEND	Adds records to the end of database.
BUG	An error usually used in relation to software problem.
CLEAR	Erase the screen.
C.B.P.P.	Contagious Bovine Pluro Pnemonia.
CASUAL AGENT	Organism that causes disease.
DISEASE	An ailment.
DELETE	Erase.
DISPLAY	Display fields of record structures.
DO	Begin execution.
DATA	Facts or information used by the system.
DATABASE	Information stored in computerised form.
EDIT	Editing of database.
EXIT	End a DO WHILE loop without halfing execution.
GO TOP/BOTTOM	Move record pointer to a specified position.
LIST	List record in current database.
KNOWLEDGE BASE	Data bank. ie. data stored in the database file.
READ	Permit data entry from get.
QUIT	Close all files of database and return to operating system.
RETURN	ENDS EXECUTION.
SEEK	Look for a particular record.
INPUT	The process of getting information into a system

OUTPUT	The process of getting information out of a system.
SYMPTOM	Signs of illness which are visible.
VACCINATION	Process of injecting materials into an animal to promote long-term immunity in the animal or, at least, the ability to tolerate a disease.

5.8 APPENDIX 1

APPEND	Adds records to the end of database.
BUG	An error usually used in relation to software problem.
CLEAR	Erase the screen.
C.B.P.P.	Contagious Bovine Pluro Pnemonia.
CASUAL AGENT	Organism that causes disease.
DISEASE	An ailment.
DELETE	Erase.
DISPLAY	Display fields of record structures.
DO	Begin execution.
DATA	Facts or information used by the system.
DATABASE	Information stored in computerised form.
EDIT	Editing of database.
EXIT	End a DO WHILE loop without halfing execution.
GO TOP/BOTTOM	Move record pointer to a specified position.
LIST	List record in current database.
KNOWLEDGE BASE	Data bank. ie. data stored in the database file.
READ	Permit data entry from get.
QUIT	Close all files of database and return to operating system.
RETURN	ENDS EXECUTION.
SEEK	Look for a particular record.
INPUT	The process of getting information into a system



Computer Assisted Diagnosis
Case Study Animal Production

Written By

Alhassan, Mohammed Amina
(PGD/MCS/042)

CADCAP Main Menu

1...Consultation
2...Update Knowledge
3...Delete Knowledge
4...Edit Knowledge
5...Personal Data
6...Exit

Enter Choice : 6

```
=[ Consultation ]=
Existing Symptoms are...
SUDDEN BODY TEMPERATURE ABOVE 40 DEGRESS    <selected>
SEVERE DEPRESSION                           <selected>
LINGERING                                   <selected>
OEDEMA                                      <selected>
PROLIFERATION OF LUNGS                      <selected>
ENLARGED BLACK SOFT SPLEEN                  <selected>
FEVER(40 TO 40.5 DEGREES)
```

Select Symptom ? [y/n] : Y

```
=[ Consultation ]=
Existing Symptoms are...
MUSCLE TREMBLING & TEETH GRINGDING.
```

```
Animal could be suffering from :
Disease       :C. B. P. P.
Possible treatments are :
Treatment     :LEUKOMYCIN
              :VACCINATION
              :ALMYCIN
        Press any key to continue...
```

[Consultation]

Existing Symptoms are...
MUSCLE TREMBLING & TEETH GRINGDING.

Animal could be suffering from :
Disease :C. B. P. P.
Causal agent is :
Causal Agent :MYCOPLASMA SPP.
:SLAUGHTER POLICIES
:DESTRUCTION OF DEAD BODIES
Press any key to continue...

[Consultation]

Existing Symptoms are...
MUSCLE TREMBLING & TEETH GRINGDING.

Animal could be suffering from :
Disease :C. B. P. P.
Possible control measures are :
Control :ERADICATION OF AFFECTED ANIMALS,
:SLAUGHTER POLICIES
:DESTRUCTION OF DEAD BODIES
Press any key to continue...

[Personal Data Modify Screen]

Record 1 of 1

Ear Tag Number : 001

Animal Type : CATTLE

Age : 2

Sex : M

Disease : C.B.P.P

Treatment : LEUKOMYCIN
: VACCINATION
: ALMYCIN

Date : 03/03/94

Correct ? [y/n] :

```
                * Program   : main.prg
* Date       : 18th Feb., 1994
* Purpose    : Main Menu

* Environmental declarations
SET STAT OFF
SET ECHO OFF
SET TALK OFF
SET BELL OFF
*****
DO WHILE .T.
   CLEAR
   @ 2,10 TO 24,70 DOUBLE
   i = 1
   DO WHILE i < 22
      @ 2+i,11 SAY REPLICATE(CHR(176),59)
      i = i + 1
   ENDDO
   STORE 6 TO Mchoice
   @ 5,25 TO 7,53 DOUBLE
   @ 6,26 SAY "      CADCAP Main Menu       "

   @ 9,25 TO 19,53 DOUBLE
   @ 10,26 CLEAR TO 18,52
   @ 11,30  SAY "1...Consultation"
   @ 12,30  SAY "2...Update Knowledge"
   @ 13,30  SAY "3...Delete Knowledge"
   @ 14,30  SAY "4...Edit Knowledge"
   @ 15,30  SAY "5...Personal Data"
   @ 16,30  SAY "6...Exit"
   @ 18,32  SAY "Enter Choice :" GET Mchoice PICT "9" RANGE 1,6
   READ

   DO CASE Mchoice
      CASE Mchoice = 1
           DO consult
           LOOP
      CASE Mchoice = 2
           DO update
           LOOP
      CASE Mchoice = 3
           DO delete
           LOOP
      CASE Mchoice = 4
           DO e
```

```
* Program  : allupdat.prg
* Date     : 17th Feb., 1994
* Purpose  : To update disease, treatment, control, causal agents
*          : knowledge base files

SELECT 1
USE symptoms
SELECT 2
USE sickcure INDEX sickcure

SELECT 1
GO TOP
DO WHILE .T.
   STORE SPACE(40) TO Msymptom, Mdisease, Mtreat1, Mtreat2, Mtreat3
   STORE SPACE(40) TO Mcontrol1, Mcontrol2, Mcontrol3, Mcausala1, Mcausal
   STORE SPACE(40) TO Mcausala3
   STORE SPACE(0) TO Mfullcode
   STORE SPACE(1) TO Mch
   GO BOTT
   STORE RECNO() TO Mlastrec
   GO TOP
   STORE "N" TO Mmore
   CLEAR
   @ 5,10 TO 14,68 DOUBLE
   @ 5,15 SAY "[ Update knowledge base ]"

   @ 7,12  SAY "Symptoms would be listed."
   @ 8,12  SAY "You are expected to select from the group the"
   @ 9,12  SAY "symptoms that correspond to your observations."
   @ 11,12 SAY "Press any key..."
   SET CONSOLE OFF
   WAIT
   SET CONSOLE ON
   @ 6,11 CLEAR TO 13,67

   @ 6,12 SAY "Existing Symptoms are..."
   GO TOP
   STORE 7 TO line
   DO WHILE .NOT. EOF()
      SELECT 1
      IF line > 13
         line = 7
         @ 22,10 SAY "Any key to see more symptoms..."
         SET CONSOLE OFF
         WAIT
         SET CONSOLE ON
         @ 22,10 SAY SPACE(40)
         @ 7,11 CLEAR TO 13,66
      ENDIF
      @ line,12 SAY symptom
      @ 22,10 SAY "Select Symptom ? [y/n] : " GET Mch PICT "@!"
      READ
      @ 22,10 SAY SPACE(40)
      IF Mch = "Y"
         @ line, 55 SAY "<selected>"
         * concatenate codes
         STORE code TO Mcode
         Mfullcode = Mfullcode + Mcode
      ELSE
         Mfullcode = Mfullcode
      ENDIF
      line = line + 1
      SKIP
```

```
ENDDO
* check Mfullcode and give error message
IF LEN(Mfullcode) = 0
   @ 22,10 SAY "Error ! No symptoms was selected..."
   @ 23,10 SAY "Press any key to continue..."
   SET CONSOLE OFF
   WAIT
   SET CONSOLE ON
   @ 22,10 CLEAR TO 23,45
   SELECT 1
   GO TOP
   @ 16,5 CLEAR TO 21,75
   @ 18,25 SAY "More Updates [y/n] ? :" GET Mmore PICT "@!"
   READ
   IF Mmore = "Y"
      LOOP
   ELSE
      EXIT
   ENDIF
ENDIF

* seek file for code and update rest
SELECT 2
SEEK Mfullcode
IF FOUND()
   @ 22,10 SAY "Such record already exist..."
   @ 23,10 SAY "Press any key to continue..."
   SET CONSOLE OFF
   WAIT
   SET CONSOLE ON
   @ 22,10 CLEAR TO 23,40
   SELECT 1
   GO TOP
   @ 16,5 CLEAR TO 21,75
   @ 18,25 SAY "More Updates [y/n] ? :" GET Mmore PICT "@!"
   READ
   IF Mmore = "Y"
      LOOP
   ELSE
      EXIT
   ENDIF
ENDIF
* not found()
@ 16,5 TO 21,75 DOUBLE
@ 17,6 CLEAR TO 20,74
@ 17,7 SAY "Disease       :" GET Mdisease PICT "@!"
@ 18,7 SAY "Treatment     :" GET Mtreat1 PICT "@!"
@ 19,7 SAY "              :" GET Mtreat2 PICT "@!"
@ 20,7 SAY "              :" GET Mtreat3 PICT "@!"
READ

@ 18,7 SAY "Control       :" GET Mcontrol1   PICT "@!"
@ 19,7 SAY "              :" GET Mcontrol2   PICT "@!"
@ 20,7 SAY "              :" GET Mcontrol3   PICT "@!"
READ

@ 18,7 SAY "Causal Agent :" GET Mcausala1   PICT "@!"
@ 19,7 SAY "             :" GET Mcausala2   PICT "@!"
@ 20,7 SAY "             :" GET Mcausala3   PICT "@!"
READ

IF Mdisease <> " "
   APPEND BLANK
```

```
      REPLACE allcode    WITH Mfullcode
      REPLACE disease    WITH Mdisease

      REPLACE treatment1 WITH Mtreat1
      REPLACE treatment2 WITH Mtreat2
      REPLACE treatment3 WITH Mtreat3

      REPLACE control1   WITH Mcontrol1
      REPLACE control2   WITH Mcontrol2
      REPLACE control3   WITH Mcontrol3

      REPLACE causala1   WITH Mcausala1
      REPLACE causala2   WITH Mcausala2
      REPLACE causala3   WITH Mcausala3

   ENDIF
   @ 16,5 CLEAR TO 21,75
   @ 18,25 SAY "More Updates [y/n] ? :" GET Mmore PICT "@!"
   READ
   IF Mmore = "Y"
      GO TOP
      LOOP
   ENDIF
   EXIT
ENDDO
CLOSE ALL
RETURN
```

```
* Program : symedit.prg
* Date    : 17th Feb., 1994
* Purpose : To edit symtoms

DO WHILE .T.
   USE symptoms
   STORE "ABCDEFGHIJKLMNOPQRSTUVWXYZ" TO Mallcodes
   STORE SPACE(40) TO Msymptom
   GO BOTT
   STORE RECNO() TO Mlastrec
   STORE SUBSTR(Mallcodes,Mlastrec,1) TO Mcode
*    IF Mcode = code  && ie. code of last record, goto next
 *       STORE SUBSTR(Mallcodes,Mlastrec + 1,1) TO Mcode
  * ENDIF
   STORE "N" TO Mmore
   CLEAR
   @ 5,10 TO 16,68 DOUBLE
   @ 14,11 TO 14,66 DOUBLE
   @ 5,15 SAY "[ Editing Symptoms ]"
   @ 6,12 SAY "Existing Symptoms are..."
   GO TOP
   STORE 7 TO line
   DO WHILE .NOT. EOF()
      STORE " " TO Medit
      IF line > 13
         line = 7
         @ 22,10 SAY "Any key to see more symptoms..."
         @ 7,11 CLEAR TO 14,66
         SET CONSOLE OFF
         WAIT
         SET CONSOLE ON
      ENDIF
      @ line,12 SAY symptom
      @ 17,12 SAY "Edit this symptom ? [y/n] :" GET Medit PICT "@!"
      READ
      @ 17,12 SAY SPACE(40)
      IF Medit = "Y"
         @ line, 45 SAY "<editing...>"
         @ 15,12 SAY "New Symptom =>" GET Msymptom PICT "@!"
         READ
      ENDIF
      IF Msymptom <> " "
         REPLACE code    WITH Mcode
         REPLACE symptom WITH Msymptom
      ENDIF
      line = line + 1
      SKIP
      @ 18,25 SAY "More Symptoms [y/n] ? :" GET Mmore PICT "@!"
      READ
      IF Mmore = "Y"
         LOOP
      ENDIF
      EXIT
   ENDDO
   EXIT
ENDDO
CLOSE ALL
RETURN
```

```
* Program : symdel.prg
* Date    : 18th Feb., 1994
* Purpose : To Delete knowledge

SELECT 1
USE symptoms
SELECT 2
USE sickcure INDEX sickcure

SELECT 1
DO WHILE .T.
   STORE SPACE(0) TO Mfullcode
   STORE SPACE(1) TO Mch
   GO BOTT
   STORE RECNO() TO Mlastrec
   STORE "N" TO Mmore
   CLEAR
   @ 5,10 TO 14,68 DOUBLE
   @ 5,15 SAY "[ Delete knowledge ]"

   @ 7,12  SAY "Symptoms would be listed."
   @ 8,12  SAY "You are expected to select from the group the"
   @ 9,12  SAY "symptoms you want deleted."
   @ 11,12 SAY "Press any key..."
   SET CONSOLE OFF
   WAIT
   SET CONSOLE ON
   @ 6,11 CLEAR TO 13,67

   @ 6,12 SAY "Existing Symptoms are..."
   GO TOP
   STORE 7 TO line
   DO WHILE .NOT. EOF()
      SELECT 1
      IF line > 13
         line = 7
         @ 22,10 SAY "Any key to see more symptoms..."
         @ 7,11 CLEAR TO 13,66
         SET CONSOLE OFF
         WAIT
         SET CONSOLE ON
      ENDIF
      @ line,12 SAY symptom
      @ 22,10 SAY "Delete Symptom ? [y/n] : " GET Mch PICT "@!"
      READ
      @ 22,10 SAY SPACE(40)
      IF Mch = "Y"
         @ line, 55 SAY "<Deleted>"
         * concatenate codes
         STORE code TO Mcode
         * check sickcure.dbf modify codes with deleted codes
          SELECT 2
          STORE 1 TO i
          DO WHILE .NOT. EOF()
             STORE allcode TO Mallcode
             DO WHILE i < 16    && 15 is the field length of 'allcode'
               STORE SUBSTR(Mallcode,i,1) TO Mcode2
               IF SUBSTR(Mallcode,i,1) <> Mcode
                 Mfullcode = Mfullcode + Mcode2
               ENDIF
               i = i + 1
             ENDDO
             REPLACE allcode WITH Mfullcode
```

```
            SKIP
          ENDDO
          GO TOP
          SELECT 1
          DELETE
*       ELSE
        *  Mfullcode = Mfullcode
       ENDIF
       line = line + 1
       SKIP
   ENDDO
   PACK

   @ 16,5 CLEAR TO 21,75
   @ 18,25 SAY "More Deletes [y/n] ? :" GET Mmore PICT "@!"
   READ
   IF Mmore = "Y"
      GO TOP
      LOOP
   ENDIF
   EXIT
ENDDO
RETURN
```

```
* Program : alledit.prg
* Date    : 20th Feb., 1994
* Purpose : TO edit diseases, controls, causal agents etc

DO WHILE .T.
   STORE " " TO Mdel
   CLEAR
   @ 5,2 TO 24,78 DOUBLE
   @ 5,15 SAY "[ Edit Knowledge ]"
   USE sickcure INDEX sickcure
   GO BOTT
   STORE RECNO() TO Mlastrec
   GO TOP
   DO WHILE .NOT. EOF()
      @ 7,5  SAY "Record " + LTRIM(STR(RECNO())) + " OF " + LTRIM(STR(Mlas
      @ 9,5  SAY "Disease :" + disease

      @ 11,5 SAY "Treatments :" + treatment1
      @ 12,5 SAY "             " + treatment2
      @ 13,5 SAY "             " + treatment3

      @ 15,5 SAY "Control    :" + control1
      @ 16,5 SAY "             " + control2
      @ 17,5 SAY "             " + control3

      @ 19,5 SAY "Causal Agent :" + causala1

      @ 22,10 SAY "Edit Knowledge ?  [y/n] :" GET Mdel PICT "@!"
      READ
      IF Mdel = "Y"
         STORE disease TO Mdisease

         STORE treatment1 TO Mtreat1
         STORE treatment2 TO Mtreat2
         STORE treatment3 TO Mtreat3

         STORE control1   TO Mcontrol1
         STORE control2   TO Mcontrol2
         STORE control3   TO Mcontrol3

         STORE causala1   TO Mcausala1

         @ 9,5  SAY "Disease :" GET Mdisease PICT "@!"

         @ 11,5 SAY "Treatments :" GET Mtreat1 PICT "@!"
         @ 12,5 SAY "             " GET Mtreat2 PICT "@!"
         @ 13,5 SAY "             " GET Mtreat3 PICT "@!"

         @ 15,5 SAY "Control    :" GET Mcontrol1  PICT "@!"
         @ 16,5 SAY "             " GET Mcontrol2  PICT "@!"
         @ 17,5 SAY "             " GET Mcontrol3  PICT "@!"

         @ 19,5 SAY "Causal Agent :" GET Mcausala1 PICT "@!"
         READ

         STORE " " TO Mconf
         @ 22,10 SAY "Entries Correct ? [y/n] :" GET Mconf PICT "@!"
         READ
         IF Mconf <> "Y"
            LOOP
         ENDIF

         REPLACE disease WITH Mdisease
```

```
            REPLACE treatment1 WITH Mtreat1
            REPLACE treatment2 WITH Mtreat2
            REPLACE treatment3 WITH Mtreat3

            REPLACE control1 WITH Mcontrol1
            REPLACE control2 WITH Mcontrol2
            REPLACE control3 WITH Mcontrol3

            REPLACE causala1 WITH Mcausala1
         ENDIF
         SKIP
      ENDDO
      PACK
      EXIT
ENDDO
CLOSE ALL
RETURN
```

```
* Program : alldel.prg
* Date    : 18th Feb., 1994
* Purpose : To delete diseases, treatments, control, etc.

DO WHILE .T.
   STORE " " TO Mdel
   CLEAR
   @ 5,2 TO 24,78 DOUBLE
   @ 5,15 SAY "[ Delete Knowledge ]"
   USE sickcure INDEX sickcure
   GO BOTT
   STORE RECNO() TO Mlastrec
   GO TOP
   DO WHILE .NOT. EOF()
      @ 7,5  SAY "Record " + LTRIM(STR(RECNO())) + " OF " + LTRIM(STR(Mla
      @ 9,5  SAY "Disease :" + disease

      @ 11,5 SAY "Treatments :" + treatment1
      @ 12,5 SAY "             " + treatment2
      @ 13,5 SAY "             " + treatment3

      @ 15,5 SAY "Control    :" + control1
      @ 16,5 SAY "             " + control2
      @ 17,5 SAY "             " + control3

      @ 19,5 SAY "Causal Agent :" + causala1

      @ 22,10 SAY "Delete Knowledge ?  [y/n] :" GET Mdel PICT "@!"
      READ
      IF Mdel = "Y"
         DELETE
      ENDIF
      SKIP
   ENDDO
   PACK
   EXIT
ENDDO
CLOSE ALL
RETURN
```

```
* Program   : consult.prg
* Date      : 20th Feb., 1994
* Purpose   : To perform consultation

SELECT 1
USE symptoms
SELECT 2
USE sickcure INDEX sickcure

SELECT 1
DO WHILE .T.
   STORE SPACE(40) TO Msymptom, Mdisease, Mtreatment1, Mtreatment2, Mtrea
   STORE SPACE(40) TO Mcontrol1, Mcontrol2, Mcontrol3, Mcausala1, Mcausal
   STORE SPACE(40) TO Mcausala3
   STORE SPACE(0) TO Mfullcode
   STORE SPACE(1) TO Mch
   GO BOTT
   STORE RECNO() TO Mlastrec
   GO TOP
   STORE "N" TO Mmore
   CLEAR
   @ 5,10 TO 14,68 DOUBLE
   @ 5,15 SAY "[ Consultation ]"

   @ 7,12  SAY "Symptoms would be listed."
   @ 8,12  SAY "You are expected to select from the group the"
   @ 9,12  SAY "symptoms that correspond to your observations"
   @ 10,12 SAY "of animal under consultation."
   @ 11,12 SAY "Press any key..."
   SET CONSOLE OFF
   WAIT
   SET CONSOLE ON
   @ 6,11 CLEAR TO 13,67

   @ 6,12 SAY "Existing Symptoms are..."
   GO TOP
   STORE 7 TO line
   DO WHILE .NOT. EOF()
   *   SELECT 1
      IF line > 13
         line = 7
         @ 22,10 SAY "Any key to see more symptoms..."
         SET CONSOLE OFF
         WAIT
         SET CONSOLE ON
         @ 7,11 CLEAR TO 13,66
         @ 22,10 SAY SPACE(40)
      ENDIF
      @ line,12 SAY symptom
      @ 22,10 SAY "Select Symptom ? [y/n] : " GET Mch PICT "@!"
      READ
      @ 22,10 SAY SPACE(40)
      IF Mch = "Y"
         @ line, 55 SAY "<selected>"
         * concatenate codes
         STORE code TO Mcode
         Mfullcode = Mfullcode + Mcode
      ELSE
         Mfullcode = Mfullcode
      ENDIF
      line = line + 1
      SKIP
   ENDDO
```

```
   * seek file for code and update rest
   SELECT 2
   SEEK Mfullcode
   IF FOUND()
      * print information
      @ 15,5 TO 23,75 DOUBLE
      @ 16,6 CLEAR TO 22,74
      @ 16,7 SAY "Animal could be suffering from :"
      @ 17,7 SAY "Disease      :" + disease

      @ 18,7 SAY "Possible treatments are :"
      @ 19,7 SAY "Treatment    :" + treatment1
      @ 20,7 SAY "             :" + treatment2
      @ 21,7 SAY "             :" + treatment3
      @ 22,15 SAY "Press any key to continue..."
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON

      @ 18,7 SAY "Possible control measures are :"
      @ 19,7 SAY "Control      :" + control1
      @ 20,7 SAY "             :" + control2
      @ 21,7 SAY "             :" + control3
      @ 22,15 SAY "Press any key to continue..."
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON

      @ 18,7 SAY "Causal agent is :                    "

      @ 19,7 SAY "Causal Agent :" + causala1
      @ 22,15 SAY "Press any key to continue..."
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON
   ELSE
      * not found()
      @ 6,11 CLEAR TO 13,67
      @ 7,12 SAY "Sorry, such code does not exist."
      @ 8,12 SAY "Please, update me !"
      @ 9,12 SAY "Use the update option of the main"
      @ 10,12 SAY "menu, select the second option to"
      @ 11,12 SAY "update my knowledge base."
      @ 13,12 SAY "Press any key to continue..."
      SET CONSOLE OFF
      WAIT
      SET CONSOLE ON
   ENDIF

   @ 15,5 CLEAR TO 23,75
   @ 18,20 SAY "Perform More Consultation [y/n] ? :" GET Mmore PICT "@!"
   READ
   IF Mmore = "Y"
      SELECT 1
      GO TOP
      LOOP
   ENDIF
   EXIT
ENDDO
CLOSE ALL
RETURN
```

```
* Program : symupdat.prg
* Date    : 17th Feb., 1994
* Purpose : To update symtoms

DO WHILE .T.
   USE symptoms
   STORE "ABCDEFGHIJKLMNOPQRSTUVWXYZ" TO Mallcodes
   STORE SPACE(40) TO Msymptom
   GO BOTT
   STORE RECNO() TO Mlastrec
   STORE SUBSTR(Mallcodes,Mlastrec,1) TO Mcode
   IF Mcode = code
      STORE SUBSTR(Mallcodes,Mlastrec+1,1) TO Mcode
   ENDIF
   STORE "N" TO Mmore
   CLEAR
   @ 5,10 TO 16,68 DOUBLE
   @ 14,11 TO 14,66 DOUBLE
   @ 5,15 SAY "[ Update Symptoms ]"
   @ 6,12 SAY "Existing Symptoms are..."
   GO TOP
   STORE 7 TO line
   DO WHILE .NOT. EOF()
      IF line > 13
         STORE 7 TO line
         @ 22,10 SAY "Any key to see more symptoms..."
         SET CONSOLE OFF
         WAIT
         SET CONSOLE ON
         @ 22,10 SAY SPACE(40)
         @ 7,11 CLEAR TO 13,66
      ENDIF
      @ line,12 SAY symptom
      line = line + 1
      SKIP
   ENDDO
   @ 15,12 SAY "New Symptom =>" GET Msymptom PICT "@!"
   READ
   IF Msymptom <> " "
      APPEND BLANK
      REPLACE code    WITH Mcode
      REPLACE symptom WITH Msymptom
   ENDIF
   @ 18,25 SAY "More Symptoms [y/n] ? :" GET Mmore PICT "@!"
   READ
   IF Mmore = "Y"
      LOOP
   ENDIF
   EXIT
ENDDO
CLOSE ALL
RETURN
```

```
* Program : person.prg
* Date    : 22nd Feb., 1994
* Purpose : To store personal data

DO WHILE .T.
   STORE SPACE(10) TO Manimal
   STORE "   "     TO Meartag
   STORE 0         TO Mage
   STORE SPACE(1)  TO Msex, Mcorr
   STORE SPACE(30) TO Mdisease, Mtreat1, Mtreat2, Mtreat3
*   STORE DATE()    TO Mrefdat
   STORE CTOD("  /  /  ") TO Mrefdat

   CLEAR
   @ 5,5 TO 23,75 DOUBLE
   @ 5,8  SAY "[ Personal Data input ]"
   @ 7,7  SAY " Ear Tag Number : " GET Meartag  PICT "999"
   @ 9,7  SAY " Animal Type    : " GET Manimal  PICT "@!"
   @ 11,7 SAY " Age            : " GET Mage     PICT "99"
   @ 13,7 SAY " Sex            : " GET Msex     PICT "@!"
   @ 15,7 SAY " Disease        : " GET Mdisease PICT "@!"
   @ 17,7 SAY " Treatment      : " GET Mtreat1  PICT "@!"
   @ 18,7 SAY "                : " GET Mtreat2  PICT "@!"
   @ 19,7 SAY "                : " GET Mtreat3  PICT "@!"
   @ 21,7 SAY " Date           : " GET Mrefdat  PICT "@!"
   READ
   IF Meartag = " "
      EXIT
   ENDIF
   @ 22,15 SAY " Correct ? [y/n] :" GET Mcorr PICT "@!"
   READ
   IF Mcorr <> "Y"
      LOOP
   ENDIF
   * replace
   USE person
   APPEND BLANK
   REPLACE animal  WITH Manimal
   REPLACE eartag  WITH Meartag
   REPLACE age     WITH Mage
   REPLACE sex     WITH Msex
   REPLACE disease WITH Mdisease
   REPLACE treatment1  WITH Mtreat1
   REPLACE treatment2  WITH Mtreat2
   REPLACE treatment3  WITH Mtreat3
   REPLACE refdat  WITH Mrefdat
ENDDO
CLOSE ALL
RETURN
```

```
* Program : persmod.prg
* Date    : 22nd Feb., 1994
* Purpose : to modify personal data

USE person INDEX person
GO BOTT
STORE RECNO() TO Mlastrec
GO TOP
DO WHILE .T.
   STORE SPACE(10) TO Manimal
   STORE "    "    TO Meartag
   STORE 0         TO Mage
   STORE SPACE(1)  TO Msex, Mcorr
   STORE SPACE(30) TO Mdisease, Mtreat1, Mtreat2, Mtreat3
   STORE DATE()    TO Mrefdat


   CLEAR
   @ 5,5 TO 23,75 DOUBLE
   @ 5,8  SAY "[ Personal Data Modify Screen ]"
   DO WHILE .NOT. EOF()
      STORE " " TO Mmod
      @ 6,35 SAY "Record " + LTRIM(STR(RECNO())) + " of " + LTRIM(STR(Mla
      @ 7,7  SAY " Ear Tag Number :  " + eartag
      @ 9,7  SAY " Animal Type    :  " + animal
      @ 11,7 SAY " Age            :  " + LTRIM(STR(age))
      @ 13,7 SAY " Sex            :  " + sex
      @ 15,7 SAY " Disease        :  " + disease
      @ 17,7 SAY " Treatment      :  " + treatment1
      @ 18,7 SAY "                :  " + treatment2
      @ 19,7 SAY "                :  " + treatment3
      @ 21,7 SAY " Date           :  " + DTOC(refdat)

      @ 22,15 SAY "Modify record ? [y/n] :" GET Mmod PICT "@!"
      READ
      IF Mmod <> "Y"
         SKIP
         LOOP
      ENDIF
      STORE animal     TO Manimal
      STORE eartag     TO Meartag
      STORE age        TO Mage
      STORE sex        TO Msex
      STORE disease    TO Mdisease
      STORE treatment1 TO Mtreat1
      STORE treatment1 TO Mtreat2
      STORE treatment1 TO Mtreat3
      STORE refdat     TO Mrefdat

      @ 7,7  SAY " Ear Tag Number :" GET Meartag     PICT "999"
      @ 9,7  SAY " Animal Type    :" GET Manimal     PICT "@!"
      @ 11,7 SAY " Age            :" GET Mage        PICT "99"
      @ 13,7 SAY " Sex            :" GET Msex        PICT "@!"
      @ 15,7 SAY " Disease        :" GET Mdisease    PICT "@!"
      @ 17,7 SAY " Treatment      :" GET Mtreat1 PICT "@!"
      @ 18,7 SAY "                :" GET Mtreat2 PICT "@!"
      @ 19,7 SAY "                :" GET Mtreat3 PICT "@!"
      @ 21,7 SAY " Date           :" GET Mrefdat     PICT "@!"
      READ
      @ 22,15 SAY SPACE(25)
      @ 22,15 SAY "Correct ? [y/n] :" GET Mcorr PICT "@!"
      READ
      IF Mcorr <> "Y"
         LOOP
      ENDIF
```

```
      * replace
      REPLACE animal  WITH Manimal
      REPLACE eartag  WITH Meartag
      REPLACE age     WITH Mage
      REPLACE sex     WITH Msex
      REPLACE disease WITH Mdisease
      REPLACE treatment1  WITH Mtreat1
      REPLACE treatment2  WITH Mtreat2
      REPLACE treatment3  WITH Mtreat3
      REPLACE refdat  WITH Mrefdat
      STORE " " TO Mann
      @ 22,15 SAY "Modify another ? [y/n] :" GET Mann PICT "@!"
      READ
      IF Mann = "Y"
         LOOP
      ENDIF
      EXIT
   ENDDO
   EXIT
ENDDO
CLOSE ALL
RETURN
```

```
* program : persdel.prg
* date    : 22nd Feb., 1994
* Purpose : to modify personal record

USE person INDEX person
GO BOTT
STORE RECNO() TO Mlastrec
GO TOP
DO WHILE .T.
   STORE SPACE(10) TO Manimal
    STORE "    "     TO Meartag
   STORE 0          TO Mage
    STORE SPACE(1)  TO Msex, Mcorr
   STORE SPACE(30) TO Mdisease, Mtreat1, Mtreat2, Mtreat3
    STORE DATE()    TO Mrefdat


   CLEAR
   @ 5,5 TO 23,75 DOUBLE
   @ 5,8  SAY "[ Personal Data Delete Screen ]"
   DO WHILE .NOT. EOF()
      STORE " " TO Mmod
      @ 6,35 SAY "Record " + LTRIM(STR(RECNO())) + " of " + LTRIM(STR(Mlast
      @ 7,7  SAY " Ear Tag Number :  " + eartag
      @ 9,7  SAY " Animal Type    :  " + animal
      @ 11,7 SAY " Age            :  " + LTRIM(STR(age))
      @ 13,7 SAY " Sex            :  " + sex
      @ 15,7 SAY " Disease        :  " + disease
      @ 17,7 SAY " Treatment      :  " + treatment1
      @ 18,7 SAY "                :  " + treatment2
      @ 19,7 SAY "                :  " + treatment3
      @ 21,7 SAY " Date           :  " + DTOC(refdat)

      @ 22,15 SAY "Delete record ? [y/n] :" GET Mmod PICT "@!"
      READ
      IF Mmod <> "Y"
         SKIP
         LOOP
      ENDIF
      @ 22,15 SAY SPACE(25)
      @ 22,15 SAY "Sure ? [y/n] :" GET Mcorr PICT "@!"
      READ
      IF Mcorr <> "Y"
         LOOP
      ENDIF
      DELETE
      PACK

      STORE " " TO Mann
      @ 22,15 SAY "Delete another ? [y/n] :" GET Mann PICT "@!"
      READ
      IF Mann = "Y"
         LOOP
      ENDIF
      EXIT
   ENDDO
   EXIT
ENDDO
CLOSE ALL
RETURN
```