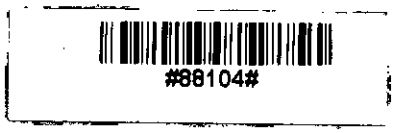
CAI FOR TEACHING A COURSE ON MATHEMATICS

BY

MD. SHAHJAHAN SHIKDER

A PROJECT REPORT SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF POSTGRADUATE DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY

DHAKA, BANGLADESH.

MARCH, 1990.

CAI FOR TEACHING A COURSE ON MATHEMATICS

A project report
by

MD. SHAHJAHAN SHIKDER

Approved as to style and contents by :-

1. Dr. Md. Quamrul Ahsan 17/4/90
Professor,
Department of Electrical and
Electronics Enginering, BUET.

Chairman

(Supervisor)

2. 17/4/90

Dr. Syed Mahabubur Rahaman
Associate Professor &
Head of the Department,
Department of Computer Science
and Engineering, BUET.

Member

3. 17/4/90

Dr. Kazi Mohiuddin Ahmed
Associate Professor,
Department of Electrical and
Electronics Engineering, BUET.

Member
(External)

ACKNOWLEDGEMENT

This project work was performed under the supervision of Dr. Quamrul Ahsan, professor, department of electrical and electronic engineering, BUET, Dhaka. I am grateful to him for his guidance, suggestion and cordial co-operation.

I also express my sincere gratitude to Dr. Syed Mahbubur Rahman, Head, department of computer science and engineering, BUET, Dhaka, for his valuable and timely co-operation and suggestion.

I am also greatly indebted to my employer Mr. H. N. Karim, President, Technohaven Co, for allowing me to attend the class and granting leave whenever required.

My sincere thanks also goes to my friends for their continuous encouragement throughout the study program.

Finally, I would like to express my gratitude to my donor, UNDP, whose scholarship support has made this study possible.

M. S. Shikder

ABSTRACT

In this project work, a package program is developed for self learning a particular subject as well as for the self evaluation of the level of understanding the subject. The package has the scope for the class teacher to keep and update the record for each student of a particular subject. This package program is interactive in nature and it is quite flexible to accommodate any type of subject. The text of the subject in the program can easily be modified and also the type as well as the number of questions in a test can also be modified. The program provides sufficient protection for the records regarding the student so that these can be reached by the student. It automatically updates the file of a student when he sits for a test and the student can not appear a test repeatedly. The program itself protects from the unauthorised use of its facilities.The core programs for this package are developed using d-BASE III + software. This package is applied to a student and also to a teacher.

TABLE OF CONTENTS.

Chapter 1

INTRODUCTION

1.1 INTRODUCTION .

The computer is a technological aid for a man . With the development of the standard of living of man the role of a computer is increasing exponentially. However, its use for education , specially in primary and secondary level has not received serious consideration outside some developed countries like USA, UK and some European countries. One of the reasons for this situation may be the cost of hardware, which is still higher compared to the existing systems and not familiarization with the Computer Aided Instruction (CAI) concepts .

Recently, the cost of computer hardware is decreasing fast. Moreover, for the last few years the need for house tutors for the secondary and higher secondary students are increasing and simultaneously the honorarium of a house tutor is also increasing rapidly. Thus the capital cost of a computer is now almost comparable with the cost of house tutors. Computers may also be used simultaneously for some other purposes.

1.2 BACKGROUND.

A large number of software are developed in different universities on Computer Aided Instructions for self learning different subjects. In what follows, some of them are discussed.

a. Chong H. Neng presented a CAI for teaching geometry in secondary school. [1] The software consists of :

1. A course comprising of learning materials with interactive facilities.

2. General student response evaluation facilities

3. A supervisory system which interacts between the student and the learning material and monitors his progress.

4. Aid for teacher in preparing the learning material as well as class management and obtaining the feedback of the performance of the students in tests and tutorials.

b. Uncharirleeporn Vongthongsre presented a CAI for teaching Mathematics in Thai language. [2] The author developed this package to learning as well as teaching Mathematics for primary school students and teachers. Lessons of this package was a part of the mathematical syllabus and this learning materials was presented in Thai language. The Author used GWBASIC to develop this package.

c. Ahmed Al-Halim designed and developed a software as well as its related hardware for learning and teaching a course. [3] The author uses Intel 8085 microprocessor for simultaneous evaluation of 64 students. Each student is equipped with a specially designed small 8-key keyboard. A system keyboard is available by which the course developer can perform the course of examination. This package can store response from students, compare with correct answer and record the evaluated grade points. The final evaluation is displayed on the monitor screen.

1.3 THESIS ORGANIZATION

There are six chapters in this Thesis. In the first chapter, an introduction to this project work, as well as its background is presented.

The second chapter presents a brief description of Computer Aided Instructions (CAI). It also presents the major uses of CAI and its salient advantages.

In the third chapter, the problem of this reachers is formulated. The objectives and scope of the reachers are also presented in this chapter. The structure form of the developed package

is also presented in this chapter. Each unit of the structure is explained using flow charts.

The related software for each unit is presented in chapter four. The function of each unit is also explained in this chapter.

The developed package is applied to a number of hypothetical students as well as teachers. The basic data and the simulation results are presented in chapter five.

The conclusions of this reachers are presented in chapter six. This chapter also presented the suggestion for future work extending this thesis.

Chapter 2

LITERATURE REVIEW

2.1 INTRODUCTION

CAI is an acronym of computer-aided (assisted or administrated) instruction. CAI is a system of individualized instruction. It provides written and visual information in a logical sequence to a student from computer. The computer serves as an audio / visual device. The student learns by reading the text material presented or by observing the graphic information displayed or by interacting with the system. [4]

This chapter pesents a brief discussion on CAI.

2.2 ADVANTAGES OF CAI.

The primary advantage of the computer over other audio / visual device is the automatic interaction and feedback that the computer can provide. Multiple paths through the course material can be taken, depending upon the individual students progress.

Another important advantage to CAI is the success factor. Because Students are confronted with exercises on their own level, they are able to do progress fast. They complete with themselves and come to see themselves as the controller of the

learning situation.

Other benefits of good CAI includes :-

1. Expanded education outreach.
2. Reduction in learning times.
3. Emphasis on the intrinsic joy of learning.
4. Deeper understanding through modeling and Simulation. [5]

2.3 TYPES OF CAI

There are several types of CAI, representing distinctions. The first distinction relates to CAI which supplements the learning situation, as opposed to that which substitutes for other modes of instruction. The former is referred to as adjunct CAI which is used to support or illustrate concepts. These concepts are then usually discussed in the regular classroom.

In contrast ,CAI materials which provide instruction of a substitute or stand-alone variety are usually of longer duration and are generally less well-known and understandable in the education world. They are referred to as primary CAI.

A second distinction refers to the simplicity-complexity level of CAI. Simplistic CAI employs easy-to-learn programming

language as well as a minimal Hardware to support the use of the program, but produces limited results i.e graphics capabilities ,large-scale calculation and the like are not components of such programs. Conversely complex CAI which permits extensive use of graphics , large-scale calculations, authoring aids etc. requires complex author language and large-scale computing capabilities to support such use. [6]

2.4 CAI COURSE IN MATHEMATICS.

The property of a computer which makes it most attractive as a teaching aid in applied mathematics is its ability to carry out complex calculations and display the results effectively and instantaneously in diagrammatic forms. The use of graphical display facilities to produce results in diagrammatic form is invaluable. A graphical display conveys qualitative information very rapidly and in a form which attracts attention. (Tawney, 1979)

2.5 METHODOLOGY IN EVALUATING CAI ON THE MICROCOMPUTER.

Evaluation as defined by the educational evaluators is the

process of conceiving, obtaining and communicating information for the guidance of educational decision making .

The approach adopted is a formalization of the informal, intuitive information - gathering methods, mainly observation and conversation, rather than as derived from the methodology of experimentation with in the natural sciences.

A major problem in the evaluation of CAI is that CAI takes time to develop and assimilate and it is only towards the end of the program that there is enough on the ground to assess its effects; it is noticeable that students who had spent only a few hours on CAI had difficulty in expressing a view on it, as did stuff who observed them, and it is impossible to test any effects with CAI which is only a minor addition to a course.

The methodology based on CAI includes :-

1. Written examinations, where results of students who learnt particular topics through CAI is compared to those who learned through traditional methods.

2. Questionnaire, typically is the most popular evaluation method, completed after each CAI exercise.

3. Interviews provide information about the problems those students encounter. Comparisons should be made between different aspects of a package and reasons given for approval or disapproval.

Chapter 3

PROBLEM FORMULATION

3.1 INTRODUCTION.

This chapter contains the definition and objectives of this project. In this chapter, it has been explained , why CAI is taken for the project work. General approach of this project are also explained in this chapter. Structure and flow diagrams of this proposed system and their explanation are also presented.

3.2 PROBLEM DEFINITION

The primary and secondary school learning of pupils can be reinforced by using tutors or after school teachers. With the availability of microcomputers, there is a scope to implementing Computer Aided Instructions (CAI) in the schools.However, an appropriate CAI package is required incorporating the local syllabus and the style of tests. As the package should replace house tutor, essentially this must incorporate the facilities of self evaluation. The facilities of updating records of each

student as well as the protection from unauthorized use must also be concluded in the package.

Therefore, the package program to be developed should include the following features :

I) The text of a subject according to the syllabus.

II) The self testing questions.

III) The self evaluating facilities.

IV) The facilities for modification and addition in the text or in the question papers by the teacher.

V) Facilities for keeping records for the students.

VI) Facilities for upkeeping a student's file when he sits for a test.

VII) Protection facilities from unauthorized use of the package and also protection from entering by a student to teachers record.

VIII) Facilities for interactive nature of learning and teaching.

3.3 OBJECTIVES OF THE STUDY

The objectives of this project is to :-

a. Develop a courseware package program to teach as well as lesson to other.

b. The effectiveness of this program would be evaluated by applying this program to few students of class six and also to some teachers of this class.

3.4 SCOPE OF THE STUDY.

The package would be developed to reinforce the concept of the students of class six regarding a particular subject. This package will help a student to learn the subject using the interactive facilities. The statistics of each student can be maintained . The course teacher can also use this package to edit the text as well as the question of the subject.

2.5 GENERAL APPROACH.

Firstly a study of class six Algebra was carried out. Areas which can be effectively presented on a computer were selected.

Discussions with the related teachers becomes useful in this selection.

The next step was to develop the corresponding courseware. The courseware is based on dBASE III$^+$.

The developed courseware was tested on pupils in a secondary school. Then the evaluation of the package was made.

SOME BASIC ASSUMPTIONS.

While developing this package the following assumptions are made :

1. Teachers have some knowledge of dBASE III$^+$.

2. Both teachers and pupils are familiar with IBM / IBM compatible personal computer or at least able to manipulate the keyboard.

3.6 STRUCTURAL FORM OF THE PACKAGE.

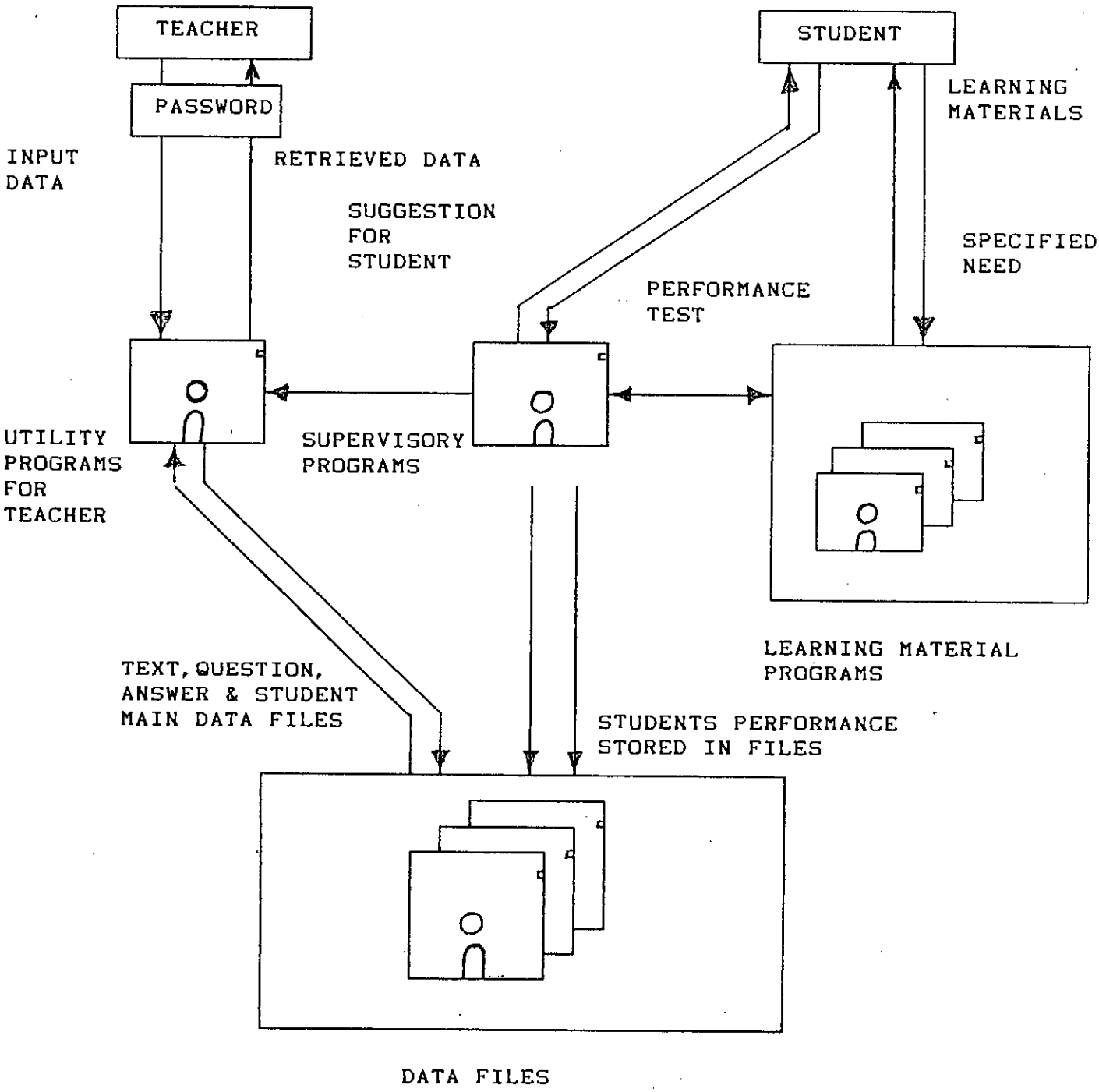


Fig - 3.01

Structure of the proposed system is shown schematically in Fig 3.01 . Basically there are two class of users; teachers and students. The structure clearly shows that different class of users do not have the same facilities to enter into each units of the package.

To access into the teacher's menu, a password is required. After entering into teacher's menu a user can edit, append, delete data from lesson file, question and it's answer file and other student files. That is a teacher can enter into the data files which consists of student's record, text and question.

From students menu one student can access into the learning material's menu and he can select any one of the three lessons according to choice. He can also sit for test, but at this time it would be checked by the supervisory program whether he is a authorized student or not.

The package consists of three main units.

a. Utility programs for teachers.

These are the programs which provides facilities to the teachers of changing i.e editing, deleting etc. of data.

b. Learning material programs.

These are the programs which display the contents of various lessons and the questions of various tests.

c. Supervisory programs.

These type of program module controls the overall situation. According to the users options these modules takes the decisions for proper work.

3.7 OPERATING PRINCIPLE OF THE PACKAGE.

At the beginning of this package it will display some screen, then automatically comes to the main menu. In the main menu there are three options. One for teachers, another for students and the last one is for exit from this package. This is shown in Fig-3.02.

When student option from main menu is selected it will display a option which includes the options for lesson, model question, test and exit to main menu.These are symbolically represented by L, S, T and E. Note that one should use a symbol

to select any one option. Say, after selection of lesson option L, it shows options for level1, level2 and level3. This is shown in Fig-3.03. Selecting any one of these option one can access to the content of the lesson. From the lesson one can go back, after finishing or with out finishing the text. From model question option one can get idea about the question type and how to answer it. From test option it will show options for test1, test2 and test3. However these test options are sequential. One student can not appear on test2 with out appearing and passing on test1. Similarly one can not appear on test3 with out appearing and passing on test2. Another important feature of this menu that a student can not sit for a particular test more than one time. These conditions for test1, test2 and test3 are shown in Fig-3.03, Fig-3.04 and Fig-3.05 respectively.

When teacher option 'T' is selected, the system will ask for password. If the given password is not valid system will give a message that given password is illegal. If password is valid then system will display a menu for teacher. There are also three options: 1) for updating the existing files 2) for retrieve data and the test one is return to main menu.

If update option is selected it will display another menu. This menu has the options for various types of files like lessons, student records, questions and answers files. This is shown in fig-3.07. Now if lesson option is selected, it will also dis-

play a menu. In this menu there are options for lesson1, lesson2, lesson3 and previous menu. It is shown in fig-3.07.

Like the lesson option, selecting option for question files or option for answer files the package display's a menu. From this menu various file can be updated. This is shown in fig-3.09.

Selecting students records option (fig 3.07), a menu is displayed. There are options for test1, test2, test3 and student main file. From various tests option records can be deleted only, no other change is allowed. Because this data comes from programs automatically when a student sit for examination. This is shown in fig-3.08. But if student file (fig 3.08) is selected, a menu is displayed from where teacher can edit, append and delete records. This is presented in fig-3.08

From teacher's menu if 'R' (retrieve option) is selected it display's a menu which gives option for individual information and overall information or for previous menu. If individual information option is selected it will ask for ID , and if ID is o.k. system will give the status of that particular student. Whether the student appeared for any examination or not, if appeared what is his marks for that test. What is his result. But if ID is not in the student main file it will give a message that ID is not available. This is menu is presented in fig-3.10.

However, in the selection of overall information process,(fig-3.10) the package will display a menu which gives the option for student main data file, test1 result, test2 result and test3 result. Now if student data file is selected it will give the options of information of all students, those appeared all tests, those passed all tests or for previous menu. This is presented in fig-3.10. According to the option the result will be in the screen.This is shown in fig-3.11.

But if test1 or test2 or test3 option is selected it will give option for information of all students appeared on that particular test, or those passed that particular test or those failed that particular test. This is in fig-3.10. According to the option result will be on the screen.It is shown in fig-3.11.

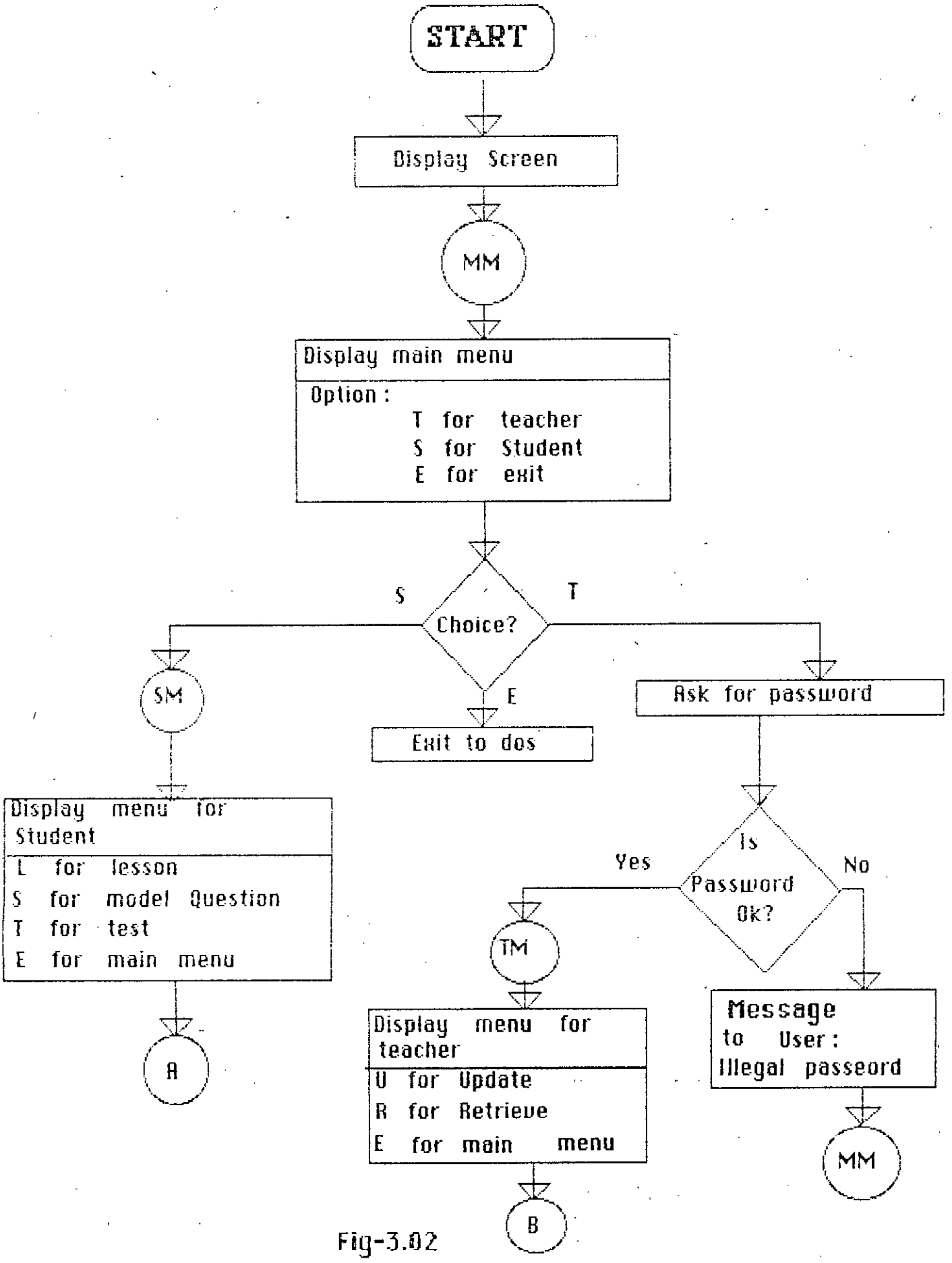

START
Display Screen
MM
Display main menu
Option :
T for teacher
S for Student
E for exit
S
T
Choice?
E
SM
Exit to dos
Ask for password
Display menu for Student
L for lesson
S for model Question
T for test
E for main menu
Is Password Ok?
Yes
No
TM
Message to User : Illegal passeord
A
Display menu for teacher
U for Update
R for Retrieve
E for main menu
MM
B

Fig-3.02

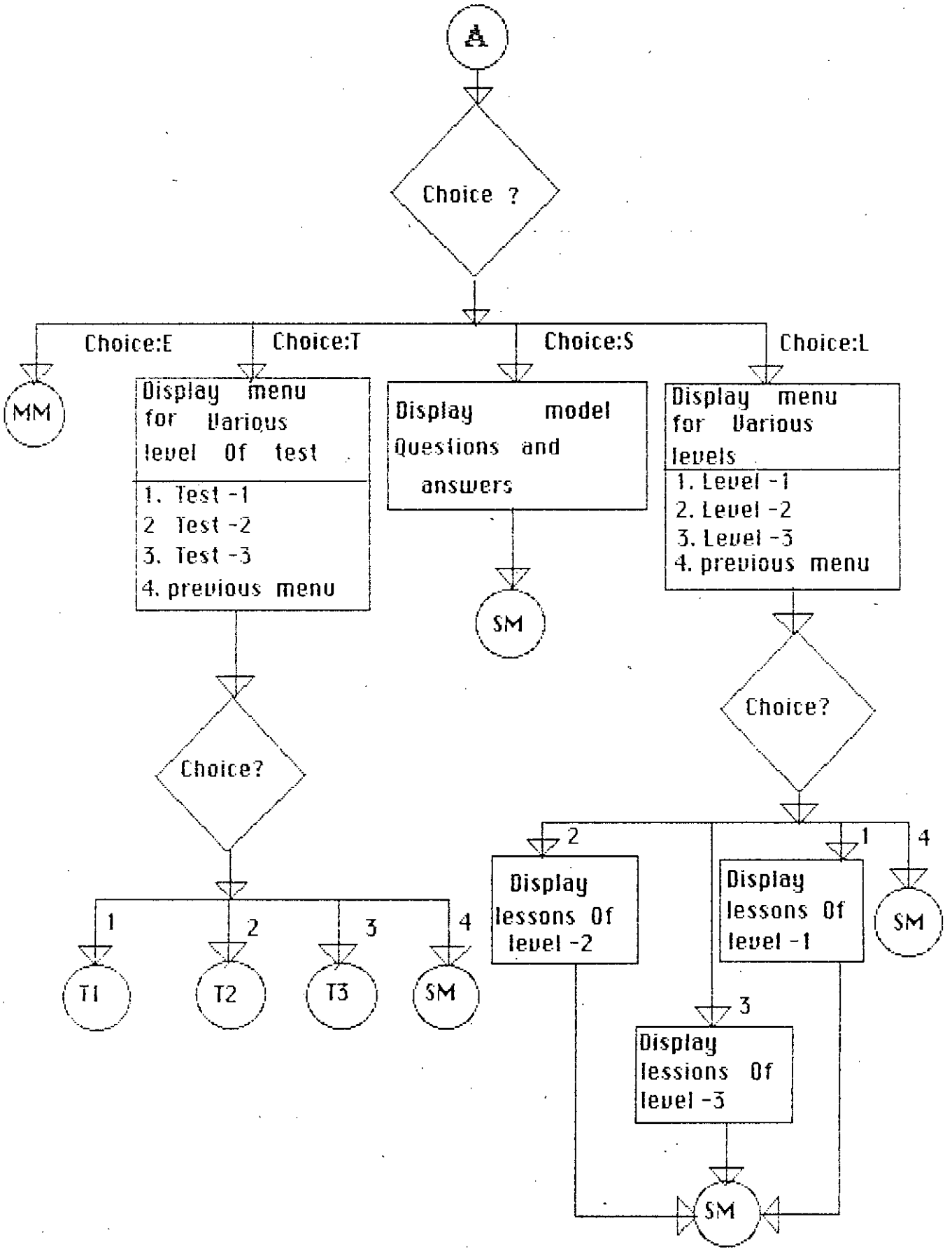

A
Choice ?
Choice:E
Choice:T
Choice:S
Choice:L
MM
Display menu for Various level Of test
1. Test -1
2 Test -2
3. Test -3
4. previous menu
Display model Questions and answers
SM
Display menu for Various levels
1. Level -1
2. Level -2
3. Level -3
4. previous menu
Choice?
Choice?
1
2
3
4
T1
T2
T3
SM
2
1
4
Display lessons Of level -2
Display lessons Of level -1
SM
3
Display lessions Of level -3
SM

Fig 3.03

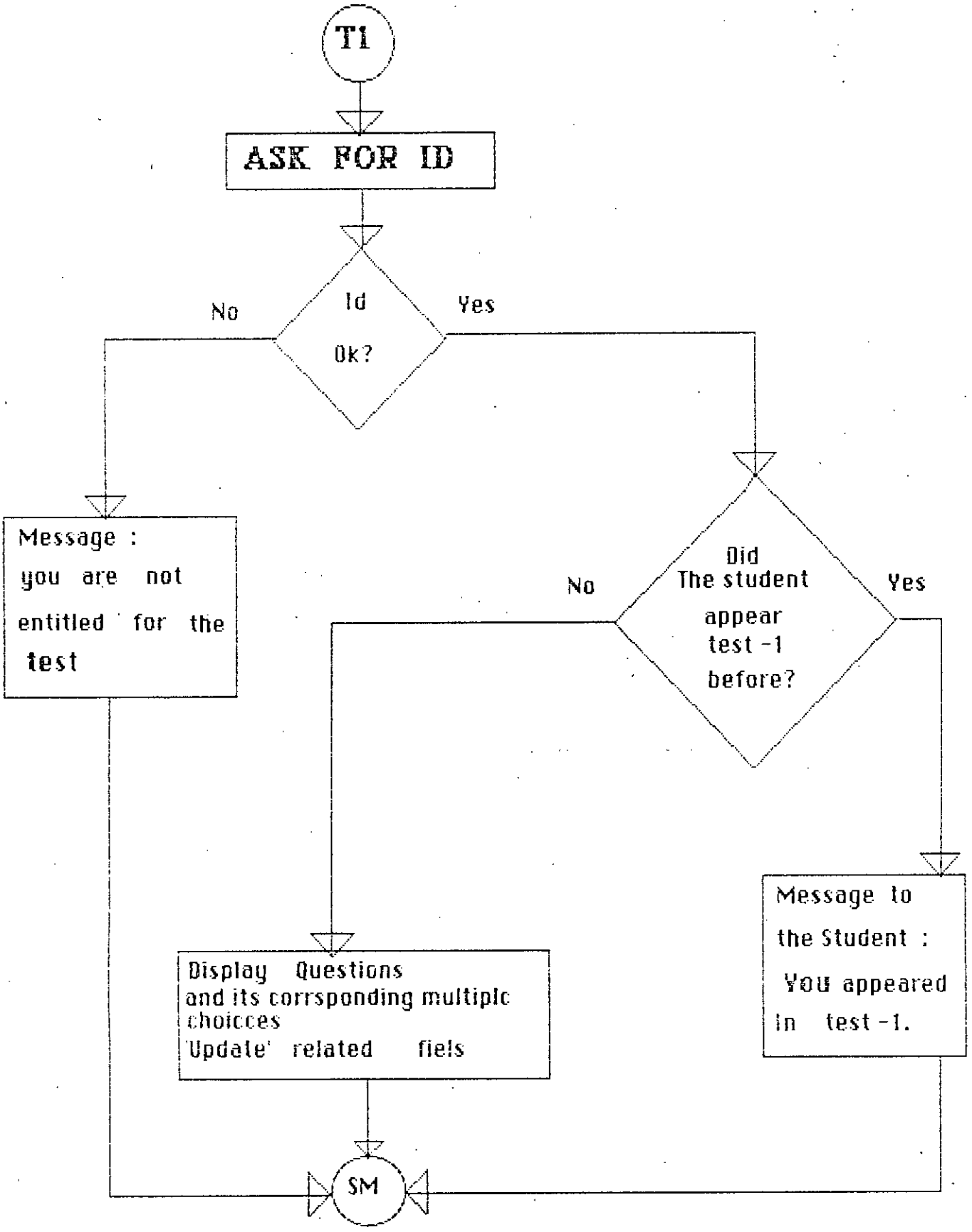

T1
ASK FOR ID
Id
Ok?
No
Yes
Message :
you are not
entitled for the
test
Did
The student
appear
test -1
before?
No
Yes
Message to
the Student :
You appeared
in test -1.
Display Questions
and its corrsponding multiple
choicces
'Update' related fiels
SM

Fig 3.04

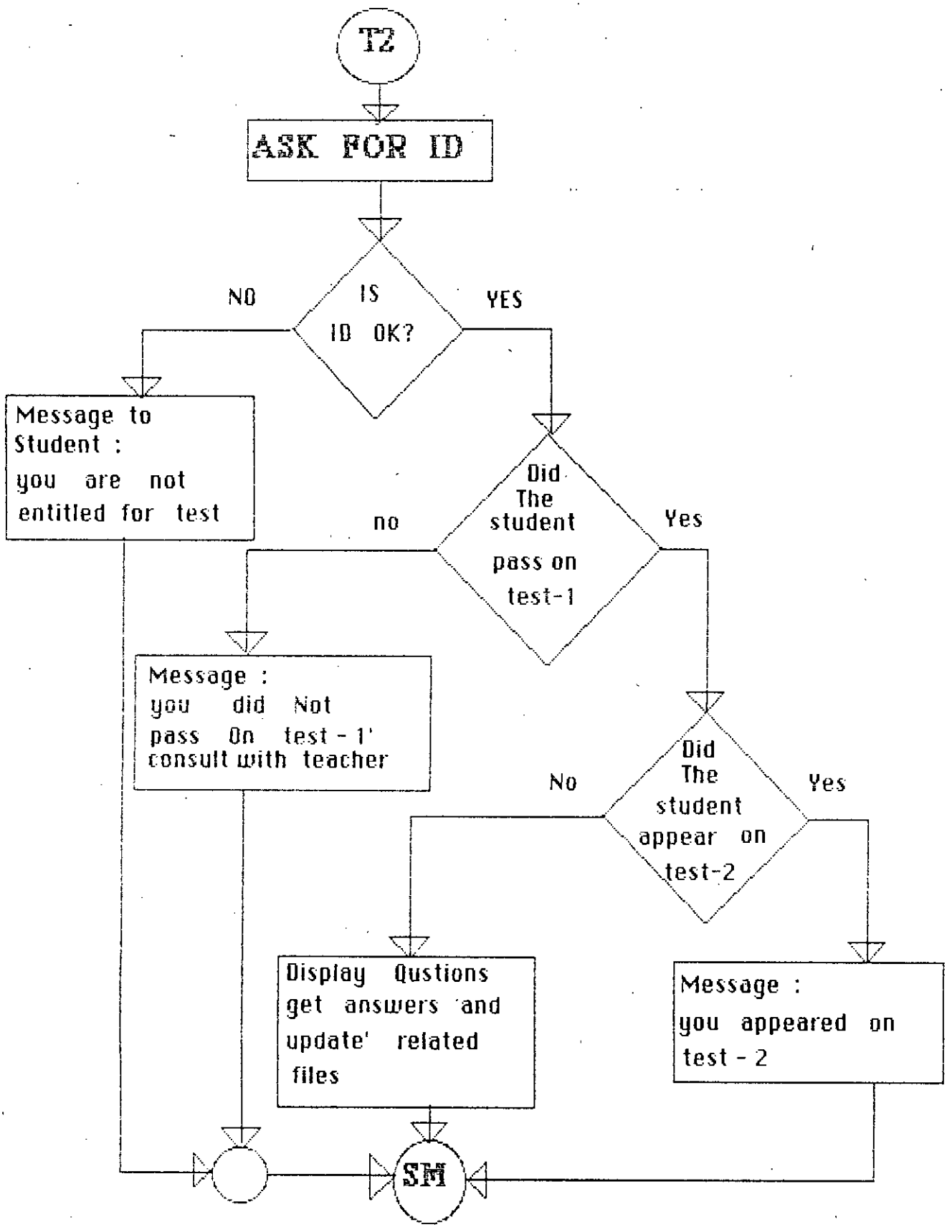

TZ
ASK FOR ID
IS ID OK?
NO
YES
Message to Student : you are not entitled for test
Did The student pass on test-1
no
Yes
Message : you did Not pass On test - 1' consult with teacher
Did The student appear on test-2
No
Yes
Display Qustions get answers and update' related files
Message : you appeared on test - 2
SM

Fig-4.05

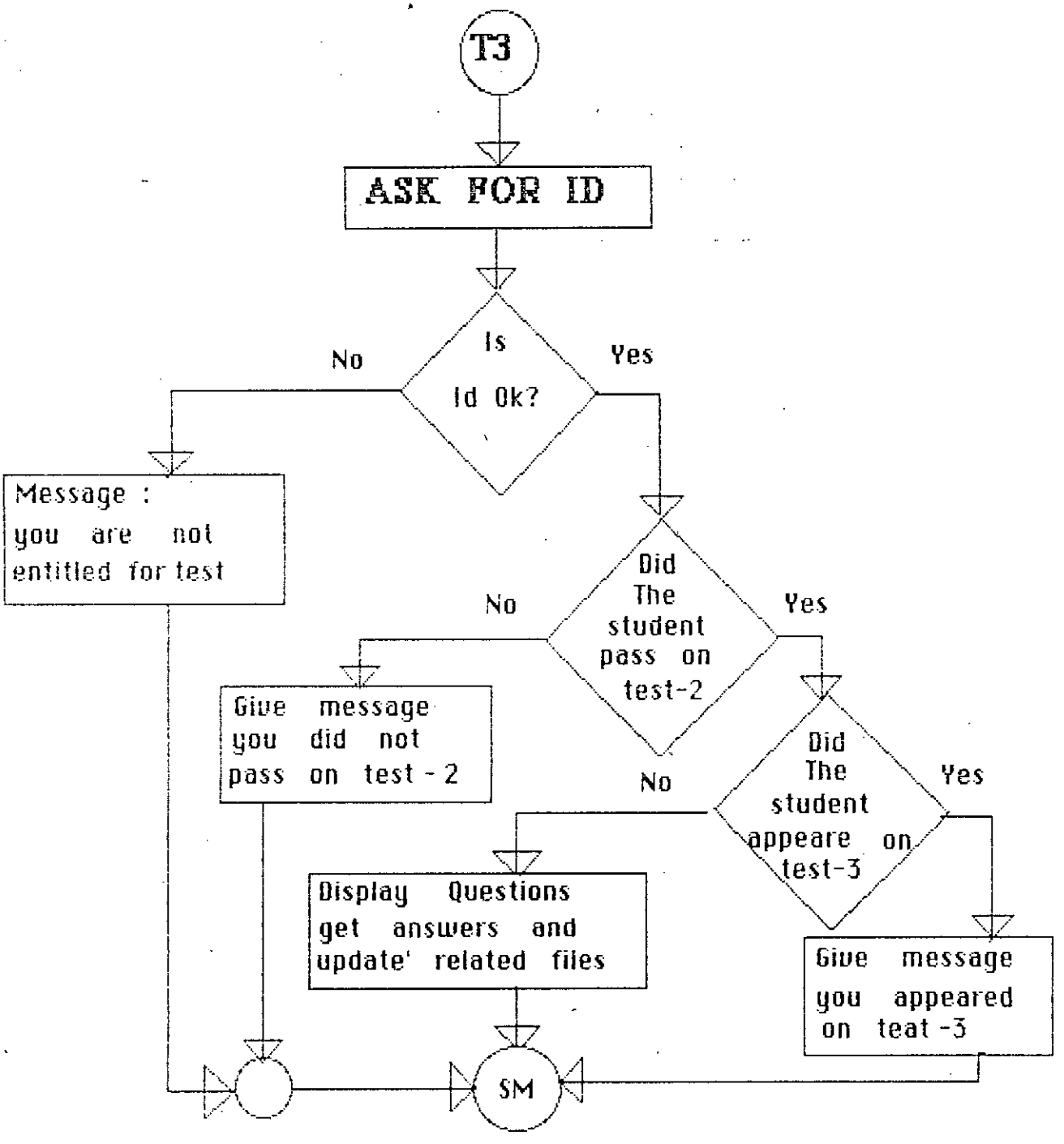

T3
ASK FOR ID
Is
Id Ok?
No
Yes
Message :
you are not
entitled for test
Did
The
student
pass on
test-2
No
Yes
Give message
you did not
pass on test - 2
Did
The
student
appeare on
test-3
No
Yes
Display Questions
get answers and
update' related files
Give message
you appeared
on teat -3
SM

Fig-4.06

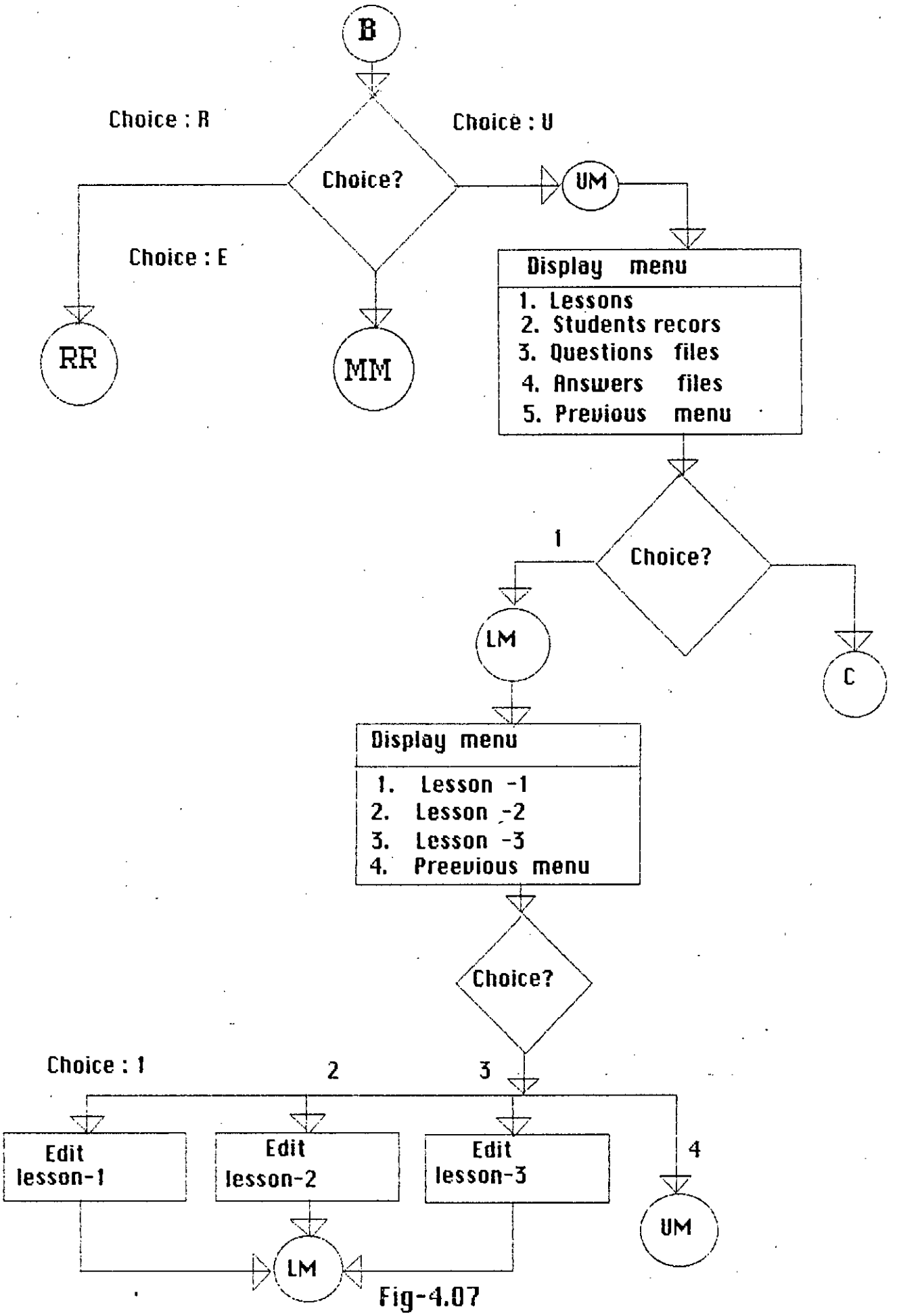

B
Choice : R
Choice : U
Choice?
UM
Choice : E
RR
MM
Display menu
1. Lessons
2. Students recors
3. Questions files
4. Answers files
5. Previous menu
1
Choice?
LM
C
Display menu
1. Lesson -1
2. Lesson -2
3. Lesson -3
4. Preevious menu
Choice?
Choice : 1
2
3
4
Edit lesson-1
Edit lesson-2
Edit lesson-3
UM
LM

Fig-4.07

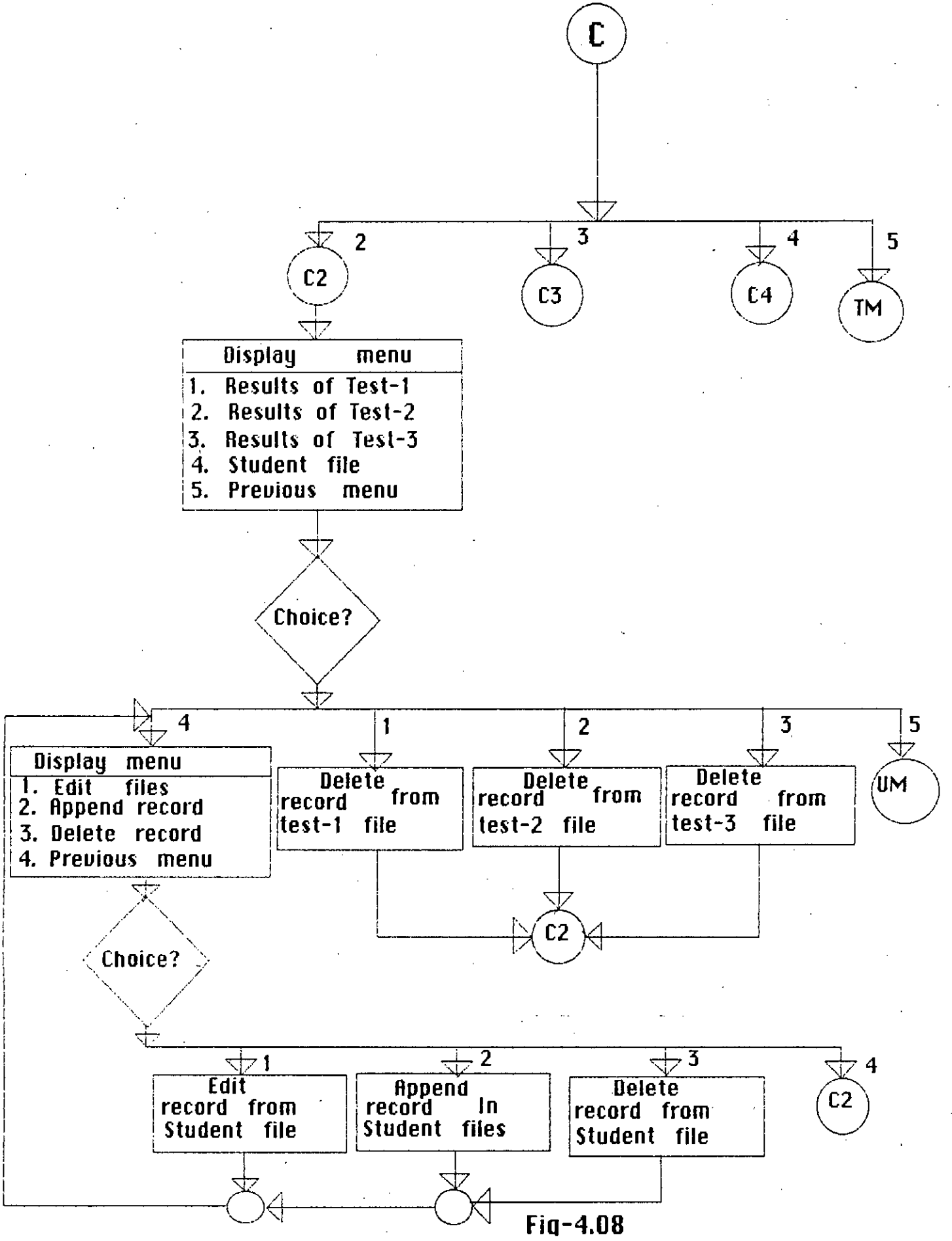

C
2
3
4
5
C2
C3
C4
TM
Display menu
1. Results of Test-1
2. Results of Test-2
3. Results of Test-3
4. Student file
5. Previous menu
Choice?
4
1
2
3
5
Display menu
1. Edit files
2. Append record
3. Delete record
4. Previous menu
Delete record from test-1 file
Delete record from test-2 file
Delete record from test-3 file
UM
C2
Choice?
1
2
3
4
Edit record from Student file
Append record In Student files
Delete record from Student file
C2

Fig-4.08

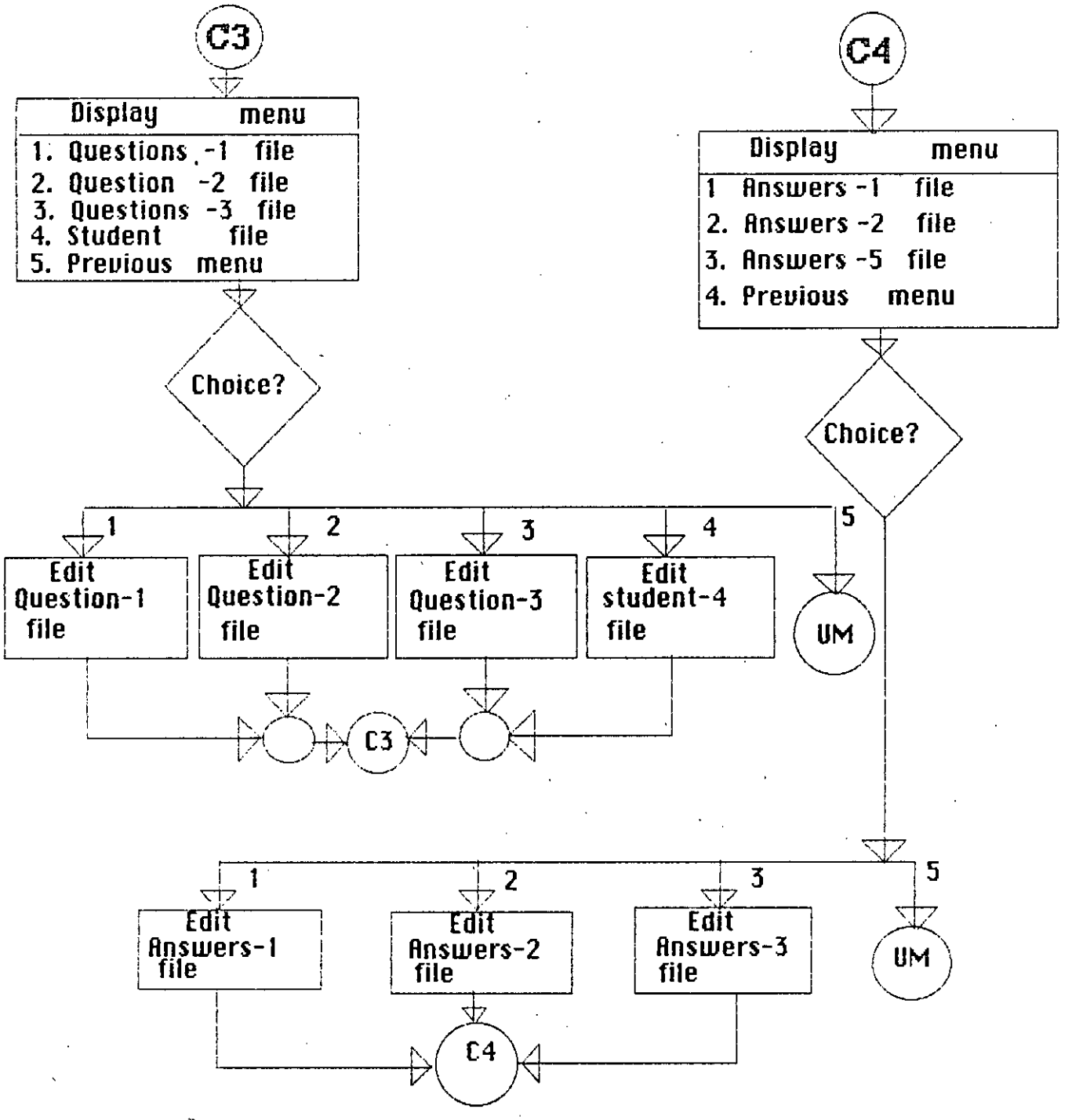

C3
Display menu
1. Questions -1 file
2. Question -2 file
3. Questions -3 file
4. Student file
5. Previous menu
Choice?
1
2
3
4
5
Edit Question-1 file
Edit Question-2 file
Edit Question-3 file
Edit student-4 file
UM
C3
C4
Display menu
1 Answers -1 file
2. Answers -2 file
3. Answers -5 file
4. Previous menu
Choice?
1
2
3
5
Edit Answers-1 file
Edit Answers-2 file
Edit Answers-3 file
UM
C4

Fig-4.09

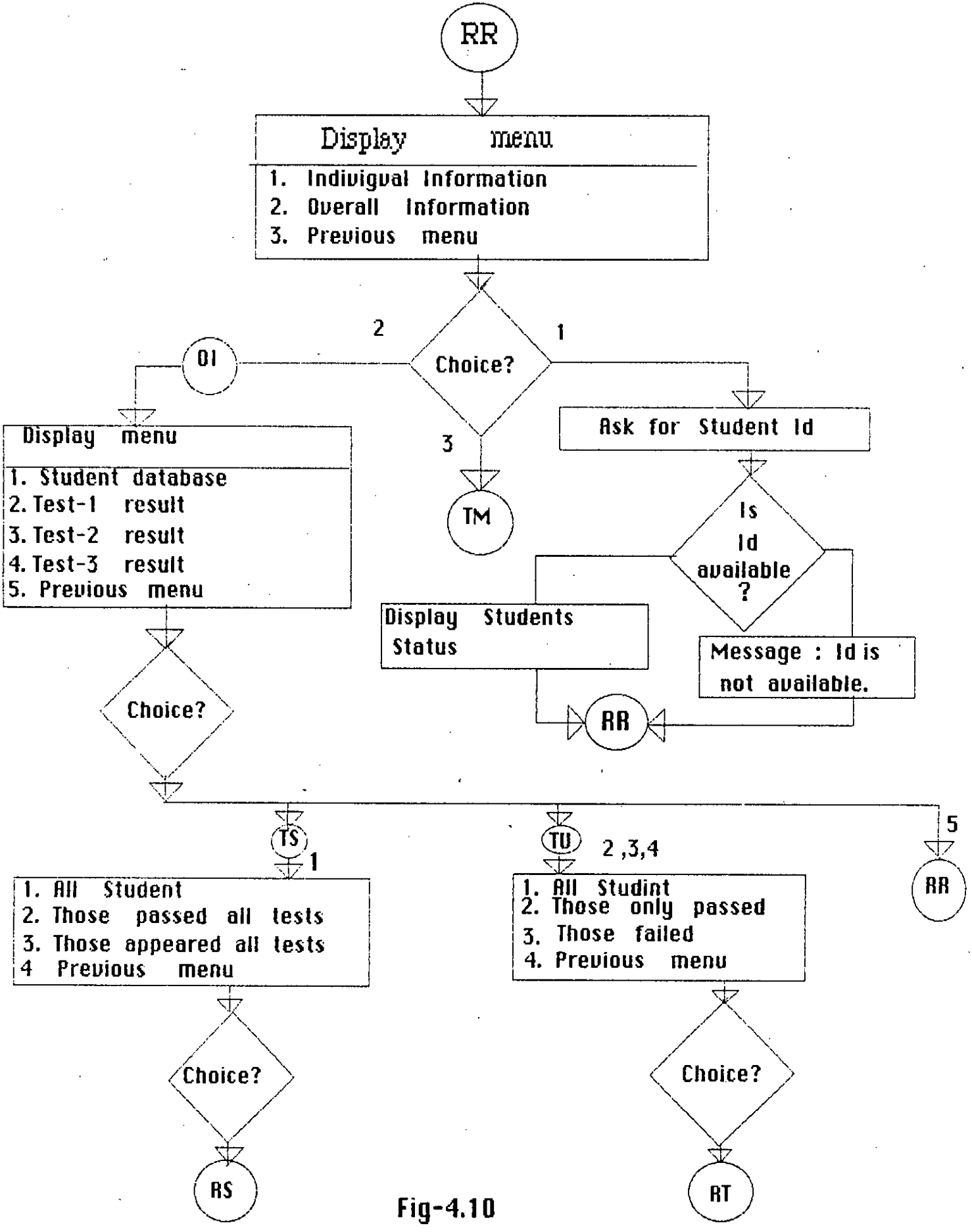

RR
Display menu
1. Indivigual Information
2. Overall Information
3. Previous menu
Choice?
2
1
3
OI
TM
Display menu
1. Student database
2. Test-1 result
3. Test-2 result
4. Test-3 result
5. Previous menu
Ask for Student Id
Is Id available ?
Display Students Status
Message : Id is not available.
RR
Choice?
TS
1
TU
2 ,3,4
5
RR
1. All Student
2. Those passed all tests
3. Those appeared all tests
4 Previous menu
1. All Studint
2. Those only passed
3. Those failed
4. Previous menu
Choice?
Choice?
RS
RT

Fig-4.10

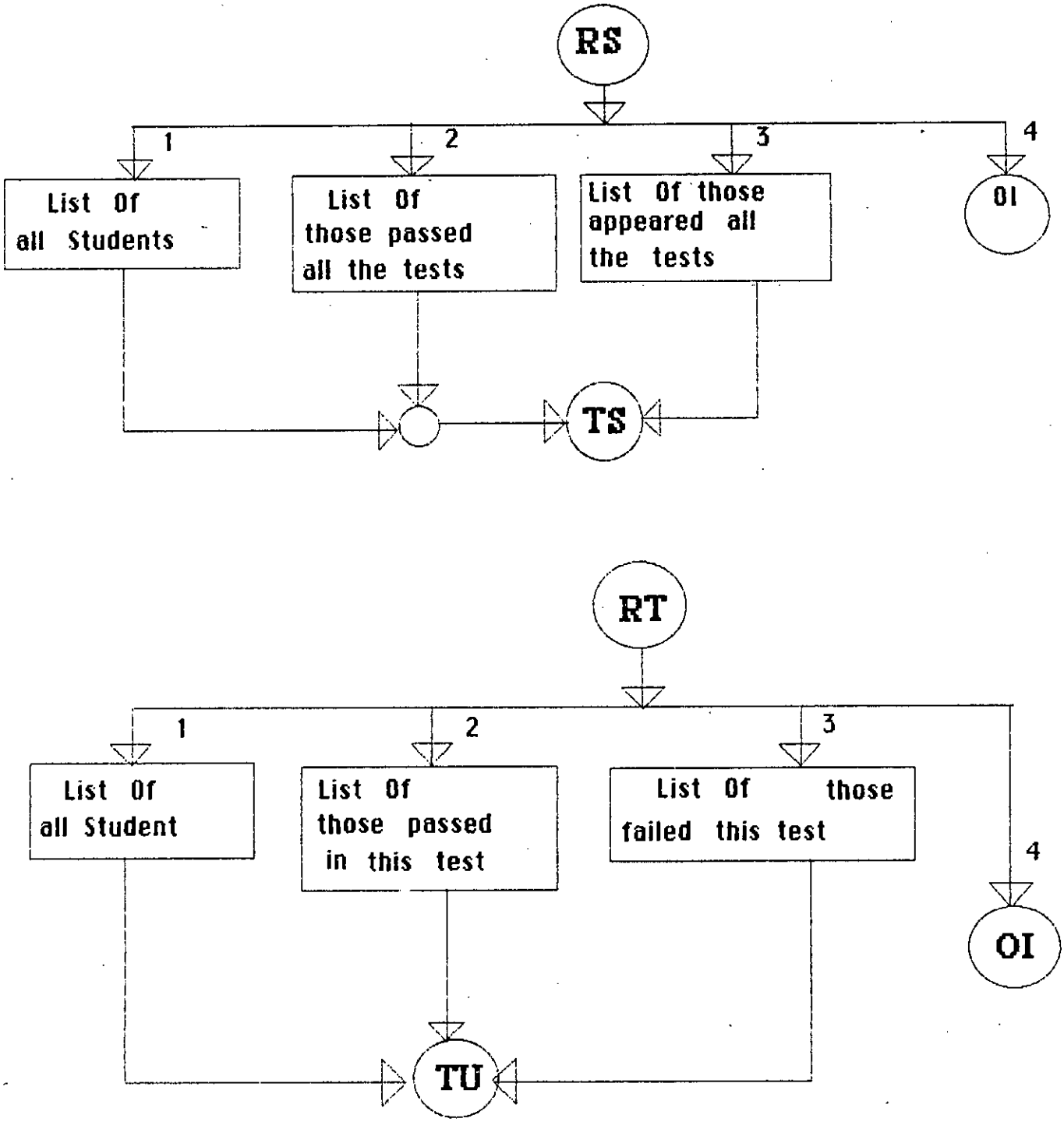

RS
1
2
3
4
List Of all Students
List Of those passed all the tests
List Of those appeared all the tests
OI
TS
RT
1
2
3
4
List Of all Student
List Of those passed in this test
List Of those failed this test
OI
TU

Fig-4.11

Chapter 4

PROGRAM DEVELOPMENT

4.1 INTRODUCTION.

The developed package has several units and the option of each unit is governed by a computer program. In this chapter the functions of these programs are described. The inter relatiocnsip among the programs is also presented in the chapter.

4.2 PROGRAM FILE RELATIONSHIP.

The inter relationship of the programs are presented in fig-4.01. Main program is MATH.PRG. It calls program CAI.PRG and CAI calls BAR.PRG and COL.PRG . These four programs altogether displays various types of screen and the main menu of this package. The main menu generates either teacher's option or student's option using TEST1.PRG and SDEN.PRG.
TEST1.PRG calls TEACH.PRG and TEACH.PRG in turn calls DRET.PRG and UPDTA1.PRG according to choice. Depending on the options UPDTAL.PRG, UPDTAS.PRG, UPDTAQ.PRG, UPDTAA.PRG are used. When UPDTAS.PRG is in use it also uses ADE.PRG . UPDTA1.PRG, UPDTAL.PRG, UPDTAS.PRG, UPDTAQ.PRG, UPDTAA.PRG, UPDTAS.PRG and ADE.PRG are used for updating files from update option. TEACH.PRG also calls DRET.PRG which again calls RETDTA.PRG and SINFOR.PRG .

RETDTA.PRG again calls SQRY1.PRG, SQRY2.PRG, SQRY3.PRG and SQRYM.PRG according to choice of user's various retrieval options.

SDENT.PRG calls ST.PRG if the selection is the siting for a test. SDENT.PRG calls LEARN.PRG if the selection by the student is to learn a lesson.

The control from ST.PRG is transfered to LOGIC1.PRG, LOGIC2.PRG or LOGIC3.PRG depending on the choice for a test. Similarly the control from LEARN1.PRG is transfered to LEARN.PRG, LEARN2.PRG or LEARN3.PRG depending on the selection of a lesson.

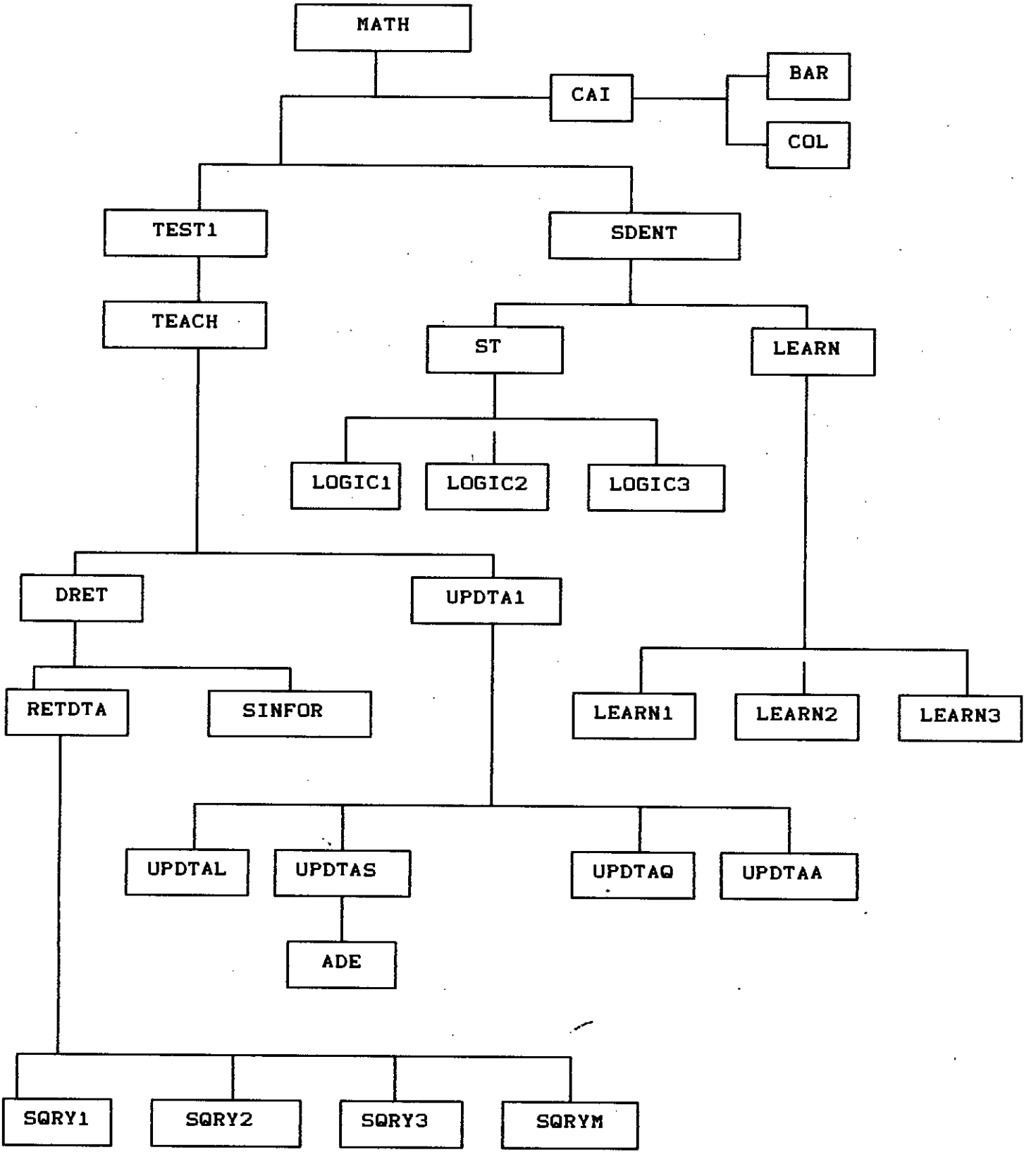


Fig - 4.01 PROGRAM RELATIONSHIP.

4.3 PROGRAM LISTING.

4.3.01 PROGRAM MATH.PRG .

* This program used for controlling the overall environment of this package and it displays the main menu of this package. In the main menu there are options for teacher, student and exit to DOS. This program calls another three programs CAI,REST1 & SDENT.

```
*
SET TALK OFF
SET HEADING OFF
SET SAFETY OFF
CLEAR ALL
SET CONSOLE ON
SET CONFIRM ON
SET PRINT OFF
SET STATUS OFF
set scoreboard off
set escape off
set bell off
*
* PROGRAM FOR SCREEN
DO CAI
*
DO WHILE .T.
CLEAR ALL
CLEAR
store ' ' to option
@ 4,10 TO 20,70 DOUBLE
@ 8,11 TO 8,69 DOUBLE
@ 6,25 SAY '      M A I N   M E N U'
@ 10,12 SAY 'OPTIONS :'
@ 12,20 SAY '   T            FOR  TEACHER '
@ 14,20 SAY '   S            FOR  STUDENT '
@ 16,20 SAY '   E            FOR  EXIT TO DOS '
@ 18,12 SAY 'ENTER YOUR OPTION :' get option
read
STORE UPPER (OPTION) TO OPTION
  DO CASE
     CASE OPTION ='T'
       DO TEST1
     CASE OPTION = 'S'
       DO SDENT
     CASE OPTION = 'E'
       CLEAR ALL
       QUIT
    OTHERWISE
       @ 23,1 SAY 'Illegal option, try again ...'
     STORE 1 TO XX
     ?? CHR (7)
```

```
      DO WHILE XX<35
         STORE XX+1 TO XX
      ENDDO
   ENDCASE
ENDDO
*EOF MATH.PRG
```

4.3.02 PROGRAM CAI.PRG.

This program is used to display two screen. One screen display rectangles, initially large then it gradually becomes smaller. After a while, the screen becomes blank and starts displaying second screen, which is also group of rectangles. But it begins with smallest one and gradually becomes larger. Then it becomes stable and starts to display very large CAI . This is written by vertical and horizontal bars. Staying few seconds it displays main menu of this package. In generating this type screen this program uses two subprograms, BAR & COL.

```
*
CLEAR ALL
CLEAR
TOP = 1
LEFT = 1
BOTTOM = 23
RIGHT = 79
DO WHILE TOP <12
   @ TOP,LEFT TO BOTTOM,RIGHT DOUBLE
   TOP = TOP + 1
   BOTTOM = BOTTOM - 1
   LEFT = LEFT + 2
   RIGHT = RIGHT - 2
ENDDO
STORE 1 TO XX
 DO WHILE XX<100
  STORE XX+1 TO XX
 ENDDO
clear
@ 12,27 say '      WELCOME  TO'
TOP=11
LEFT=21
BOTTOM=13
RIGHT=59
DO WHILE TOP >0
   @ TOP,LEFT TO BOTTOM ,RIGHT double
   TOP=TOP-1
   BOTTOM = BOTTOM + 1
   LEFT = LEFT - 2
   RIGHT = RIGHT + 2
ENDDO
 STORE 1 TO XX
```

```
DO WHILE XX < 100
  STORE XX+1 TO XX
  ENDDO
  CLEAR
TOP = 1
LEFT = 1
BOTTOM = 23
RIGHT = 79
  DO WHILE TOP < 6
  @ TOP,LEFT TO BOTTOM,RIGHT DOUBLE
  TOP = TOP + 1
  BOTTOM = BOTTOM - 1
  LEFT = LEFT + 2
  RIGHT = RIGHT - 2
  ENDDO
set procedure to bar
do bar with 8,24,220,8
do bar with 8,39,220,8
do bar with 12,39,220,8
do bar with 15,24,220,8
close procedure
set procedure to col
do col with 9,24,219,6
do col with 9,39,219,6
do col with 9,47,219,6
do col with 8,54,220,1
do col with 9,54,219,6
close procedure
XX=1
 DO WHILE XX < 200
  XX = XX + 1
 ENDDO
clear
RETURN
*eof
```

4.3.03 PROCEDURE BAR .

This procedure is used to make horizontal bar on the screen. This is used in writing very large CAI. It's function is to make horizontal bar according to the given co-ordinate and ASSCI value of the character by which bar would be made.

```
PROCEDURE BAR
PARAMETERS RR,CC,CHAR,NUMBER
  ROW = RR
  COL = CC
  TARGET = CC + NUMBER
```

```
  DO WHILE COL <= TARGET .AND. COL < 80
   @ ROW , COL SAY CHR(CHAR)
   COL = COL + 1
  ENDDO
RETURN
*END OF PROCEDURE BAR.
```

4.3.04 PROCEDURE COL FOR .

This procedure is used to make vertical column on the screen. This is also used in writing very large CAI. It's function is to make vertical bar on the screen according to given co-ordinates and ASSCI value of the character by which the column would be made.

```
PROCEDURE COL
PARAMETERS RR,CC,CHAR,NUMBER
 ROW = RR
 COL =CC
 TARGET = RR + NUMBER
  DO WHILE ROW <= TARGET .AND. ROW < 24
   @ ROW,COL SAY CHR(CHAR)
   ROW = ROW + 1
 ENDDO
RETURN
*EOF PROCEDURE COL.
```

4.3.05 PROGRAM TEST1.PRG.

This program used for password control. When user selects option teacher from main menu then this program checks weather the user is an authorized teacher or not. If user is an authorized teacher it calls another program module TEACH which display menu for teachers. Other wise it does not give permission to access teacher menu and deliver message that given password is illegal.

```
clear all
clear
store '         ' to password
@ 10,10 say 'Enter your password'
set color to w, /x
@ 10,30 get password
read
set color to w, /w
IF  password = 'BABUL'.OR. password='REAZ' .OR. password='SIR'
     clear
     @ 20,10 say 'Working...'
     do teach
```

```
ELSE
  @ 23,1 say 'Illegal password !'
  STORE 1 TO XX
  DO WHILE XX<35
    STORE XX+1 TO XX
    ENDDO
return
*eof test1
```

4.3.06 PROGRAM TEACH.PRG .

This program displays the menu for teacher. After selecting teacher option from main menu, if the password is ok then this menu is displayed. This program uses another two program updta1 and dret. When update option is selected then updta1 is called, when retrieve data option is selected from teacher menu then dret is called. Of course one can also go back to main menu from here.

```
SELECT A
DO WHILE .T.
CLEAR ALL
CLEAR
store ' ' to option
@ 4,10 TO 21,70 DOUBLE
@ 8,11 TO 8,69 DOUBLE
@ 6,25 SAY 'M E N U   F O R   T E A C H E R   '
@ 10,12 SAY 'OPTIONS :'
@ 12,20 SAY '   U            FOR  UPDATE  FILES '
@ 14,20 SAY '   R            FOR  RETRIEVE DATA'
@ 16,20 SAY '   E            FOR  EXIT TO MAIN MENU '
@ 19,12 SAY 'ENTER YOUR OPTION :' get option
read
STORE UPPER (OPTION) TO OPTION
  DO CASE
     CASE OPTION = 'U'
       DO UPDTA1
     CASE OPTION = 'R'
       DO DRET
     CASE OPTION = 'E'
       clear
       RETURN
     OTHERWISE
       @ 23,1 SAY 'Illegal option, try again ...'
     STORE 1 TO XX
     ?? CHR (7)
     DO WHILE XX<35
        STORE XX+1 TO XX
     ENDDO
     @ 22,1 CLEAR TO 23,70
  ENDCASE
ENDDO
*EOF TEACH.PRG
```

4.3.07 PROGRAM UPDTA1 .

This program is used to display options for various files, when user wants to update files from teacher menu. For updating various files it calls other four programs; updtal, updtas, updtaq & updtaa. According to options these programs are called.

```
clear all
clear
do while .t.
@ 4,20 to 16,54 double
@ 7,25 say '    1. LESSON FILES'
@ 8,25 say '    2. STUDENT FILES'
@ 9,25 say '    3. QUESTION FILES'
@ 10,25 SAY '    4. ANSWER FILES'
@ 11,25 SAY '    5. PREVIOUS MENU'
STORE ' ' TO OPTION
@    14,25 SAY '    Enter your choice ' GET OPTION
READ
  DO CASE
     CASE OPTION='1'
      DO UPDTAL
     CASE OPTION='2'
      DO UPDTAS
     CASE OPTION='3'
      DO UPDTAQ
     CASE OPTION = '4'
      DO UPDTAA
     CASE OPTION = '5'
      CLEAR
      RETURN
     OTHERWISE
      @ 23,1 SAY 'ILLEGAL OPTION'
      STORE 1 TO XX
      ?? CHR (7)
       DO WHILE XX < 50
       STORE  XX+1  TO  XX
       ENDDO
       @ 23,1 CLEAR TO 23,70
  ENDCASE
 ENDDO
*EOF UPDATE1.PRG
```

4.3.08 PROGRAM UPDTAL.PRG .

This program is used to display options for various lessons, then according to choice it gives facility of updating the particular lesson file.

```
CLEAR ALL
clear
```

```
do while .t.
@ 4,20 to 16,54 double
@ 7,26 say '    1. LESSON - 1'
@ 8,26 say '    2. LESSON - 2'
@ 9,26 say '    3. LESSON - 3 '
@ 10,26 SAY '    4. PREVIOUS MENU'
STORE ' ' TO OPTION
@    13,26 SAY '    Enter your choice ' GET OPTION
READ
  DO CASE
     CASE OPTION='1'
      USE LESSON1
      EDIT
     CASE OPTION='2'
      USE LESSON2
      EDIT
     CASE OPTION='3'
      USE LESSON3
      EDIT
     CASE OPTION = '4'
     CLEAR
     RETURN
 OTHERWISE
     @ 23,1 SAY 'ILLEGAL OPTION'
     ?? CHR (7)
 STORE 1 TO XX
  DO WHILE XX<35
   STORE XX+1 TO XX
   ENDDO
    @ 23,1 CLEAR TO 23,70
  ENDCASE
 CLEAR
 ENDDO
*EOF UPDATEL.PRG
```

4.3.09 PROGRAM UPDTAS.PRG .

The functions of this program are, firstly to display options of various files related with students data. Various test files could be updated from here depending on options. If student main database option is selected it calls program ade.

```
CLEAR ALL
clear
do while .t.
@ 4,20 to 16,54 double
@ 7,26 say '    1. TEST1  DATA'
@ 8,26 say '    2. TEST2  DATA'
@ 9,26 say '    3. TEST3  DATA '
@ 10,26 SAY '    4. STUDENT DATABASE'
@ 11,26 SAY '    5. PREVIOUS MENU'
STORE ' ' TO OPTION
@    14,26 SAY '    Enter your choice ' GET OPTION
```

```
READ
  DO CASE
     CASE OPTION='1'
      CLEAR
      STORE ' ' TO CHOICE
  @ 10,15 SAY 'Do you want to delete records ? (Y/N)' GET CHOICE
  READ
       CHOICE = UPPER (CHOICE)
       IF CHOICE = 'Y'
        CLEAR
        RN = 0
        @ 10,15 SAY 'Enter record no.'GET RN
        READ
        USE ST1 INDEX SS1
        DELETE RECORD RN
        PACK
      ELSE
        CLEAR
        RETURN
      ENDIF
      *
    CASE OPTION='2'
      CLEAR
      STORE ' ' TO CHOICE
  @ 10,15 SAY 'Do you want to delete records ? (Y/N)' GET CHOICE
  READ
       CHOICE = UPPER (CHOICE)
       IF CHOICE = 'Y'
        CLEAR
        RN = 0
        @ 10,15 SAY 'Enter record no.'GET RN
        READ
        USE ST2 INDEX SS2
        DELETE RECORD RN
        PACK
      ELSE
        CLEAR
        RETURN
      ENDIF
      *
    CASE OPTION='3'
      CLEAR
      STORE ' ' TO CHOICE
  @ 10,15 SAY 'Do you want to delete records ? (Y/N)' GET CHOICE
  READ
       CHOICE = UPPER (CHOICE)
       IF CHOICE = 'Y'
        CLEAR
        RN = 0
        @ 10,15 SAY 'Enter record no.'GET RN
        READ
        USE ST3 INDEX SS3
        DELETE RECORD RN
        PACK
```

```
      ELSE
        CLEAR
        RETURN
      ENDIF
      *
     CASE OPTION = '4'
      DO ADE
     CASE OPTION = '5'
     CLEAR
     RETURN
 OTHERWISE
     @ 23,1 SAY 'ILLEGAL OPTION'
     ?? CHR (7)
 STORE 1 TO XX
  DO WHILE XX<35
   STORE XX+1 TO XX
   ENDDO
   @ 23,1 CLEAR TO 23,70
 ENDCASE
ENDDO
*EOF UPDTAS.PRG
```

4.3.10 PROGRAM ADE.PRG .

This program is called by updtas. It's functions are to append, edit & delete records from student main database.

```
CLEAR
DO WHILE .T.
@ 4,12 TO 17,56 DOUBLE
@ 8,25 SAY '1. EDIT RECORD'
@ 9,25 SAY '2. APPEND RECORD'
@ 10,25 SAY '3. DELETE RECORD'
@ 11,25 SAY '4. PREVIOUS MENU'
STORE ' ' TO OPTION
@   15,25 SAY 'Enter your choice ' GET OPTION
READ
  DO CASE
     CASE OPTION='1'
       USE SDENTM INDEX SSS
       EDIT
    CASE OPTION = '2'
      USE SDENTM INDEX SSS
      APPEND
    CASE OPTION = '3'
        CLEAR
        RN = 0
        @ 10,15 SAY 'Enter record no.'GET RN
        READ
        USE SDENTM INDEX SSS
        COUNT TO MRN
        IF RN  <=  MRN
```

```
        DELETE RECORD RN
        PACK
        ELSE
         CLEAR
         @ 10,20 SAY ' Record number beyond the range !'
         @ 22,1 say ' '
         WAIT
         ENDIF
      *
    CASE OPTION = '4'
       CLEAR
       RETURN
    OTHERWISE
        @ 23,1 SAY ' Illegal option !'
        ?? CHR (7)
         XX=1
         DO WHILE XX < 50
         XX = XX + 1
         ENDDO
        @ 23,1 CLEAR TO 23,70
    ENDCASE
   CLEAR
   ENDDO
*EOF ADE.PRG
```

4.3.11 PROGRAM UPDTAQ.PRG .

It is used for updating various question files. Firstly it displays options for question files of several level. According to option it gives facility to update that particular question file.

```
CLEAR ALL
clear
do while .t.
@ 4,20 to 16,54 double
@ 7,26 say '    1. TEST1  QUESTIONS'
@ 8,26 say '    2. TEST2  QUESTIONS'
@ 9,26 say '    3. TEST3  QUESTIONS '
@ 10,26 SAY '    4. PREVIOUS MENU'
STORE ' ' TO OPTION
@    14,26 SAY '    Enter your choice ' GET OPTION
READ
  DO CASE
     CASE OPTION='1'
      USE QTION
      EDIT
     CASE OPTION='2'
      USE QTION1
      EDIT
```

```
      CASE OPTION='3'
       USE QTION2
       EDIT
      CASE OPTION = '4'
      CLEAR
      RETURN
  OTHERWISE
      @ 23,1 SAY 'ILLEGAL OPTION'
  STORE 1 TO XX
   ?? CHR (7)
   DO WHILE XX<35
    STORE XX+1 TO XX
    ENDDO
    @ 23,1 CLEAR TO 23,70
  ENDCASE
CLEAR
ENDDO
*EOF UPDTAQ.PRG
```

4.3.12 PROGRAM UPDTAA.PRG .

Firstly this program displays options for answer files of several level. Depending on the option it gives facility to update the answer files.

```
CLEAR ALL
clear
do while .t.
@ 4,20 to 16,54 double
@ 7,26 say '    1. TEST1  ANSWERS'
@ 8,26 say '    2. TEST2  ANSWERS'
@ 9,26 say '    3. TEST3  ANSWERS '
@ 10,26 SAY '    4. PREVIOUS MENU'
STORE ' ' TO OPTION
@    14,26 SAY '    Enter your choice ' GET OPTION
READ
  DO CASE
     CASE OPTION='1'
      USE QANS INDEX QA1
      EDIT
     CASE OPTION='2'
      USE QANS1 INDEX QA2
      EDIT
     CASE OPTION='3'
      USE QANS2 INDEX QA3
      EDIT
     CASE OPTION = '4'
     CLEAR
     RETURN
    OTHERWISE
```

```
     @ 23,1 SAY 'Illegal option, try again ...'
    STORE 1 TO XX
    ?? CHR (7)
    DO WHILE XX<35
    STORE XX+1 TO XX
    ENDDO
    @ 23,1 CLEAR TO 23,70
  ENDCASE
 CLEAR
 ENDDO
*EOF UPDTAQ.PRG
```

3.3.13 DRET.PRG .

This program simply displays a menu weather the user wants to retrieve information of individual student or all students. If individual option is selected it calls program sinfor, and program retdta is called if overall information is selected.

```
clear all
clear
do while .t.
   STORE ' ' TO CHOICE
   @ 6,14 to 17,62 double
   @ 8,28 say '1.INDIVIDUAL  INFORMATION.'
   @ 10,28 SAY '2. OVERALL  INFORMATION.'
   @ 12,28 SAY '3. PREVIOUS MENU.'
   @ 15,28 SAY 'Enter your choice' GET CHOICE
   READ
   DO CASE
      CASE CHOICE = '1'
       DO SINFOR
      CASE CHOICE = '2'
          DO RETDTA
      CASE CHOICE ='3'
         CLEAR
         RETURN
      OTHERWISE
        @ 23,1 SAY 'Illegal Choice'
        STORE 1 TO XX
        ?? CHR (7)
         DO WHILE XX<35
         STORE XX+1 TO XX
         ENDDO
        @ 23,1 CLEAR TO 23,70
      ENDCASE
   CLEAR
 ENDDO
* EOF DRET.PRG
```

4.3.14 PROGRAM SINFOR.PRG.

This is the program which informs about status of a particular student. Firstly it asks for student ID, if given ID is valid it displays name, ID, status of test1, test2 & test3. For invalid ID it simply give message ID is not found. Checking all the marks it also comments weather that student passed or failed. It also informs the grade acquired by the student.

```
clear all
clear
TM = 0
kpress = '       '
@ 8,17 TO 12,52
@ 10,20 say  'Enter students Roll No ' get kpress
read
SELECT A
use sdentm index sss
seek  kpress
If found()
CLEAR
@ 7,6 TO 17,72
@ 9,8 say  'STUDENT        STUDENT  RESULTS OF     RESULTS OF
RESULTS OF'
@ 10,8 SAY 'ROLL NO        NAME     TEST1          TEST2          TEST3'
@ 12,8 SAY kpress
@ 12,22 SAY NAME
SELECT B
USE ST1 INDEX SS1
SEEK kpress
IF FOUND()
@ 12,30 SAY MARK1
TM = MARK1
SELECT C
USE ST2 INDEX SS2
SEEK kpress
IF FOUND()
@ 12,44 SAY MARK2
TM = TM + MARK2
SELECT D
USE ST3 INDEX SS3
SEEK kpress
IF FOUND()
@ 12,58 SAY MARK3
TM = TM + MARK3
   IF MARK3 < 60
   @ 15,8 SAY 'REMARKS: FAIL'
   ELSE
     IF TM < 240
     @ 15,8 SAY 'REMARKS: PASS'
     ELSE
```

```
      @ 15,8 SAY 'REMARKS: FIRST CLASS'
      ENDIF
   ENDIF
ELSE
@ 12,59 SAY 'NOT APPEARED'
@ 15,8 SAY 'REMARKS: FAIL'
ENDIF
ELSE
@ 12,45 SAY 'NOT APPEARED'
@ 12,59 SAY 'NOT APPEARED'
@ 15,8  SAY 'REMARKS: FAIL'
ENDIF
ELSE
@ 12,31 SAY 'NOT APPEARED'
@ 12,45 SAY 'NOT APPEARED'
@ 12,59 SAY 'NOT APPEARED'
@ 15,8 SAY 'REMARKS: FAIL'
ENDIF
else
CLEAR
@ 12,10 say 'ROLL NO NOT FOUND'
endif
@ 22,1 SAY ''
WAIT
CLEAR
RETURN
*EOF SINFOR.PRG
```

4.3.15 PROGRAM RETDTA.PRG FOR .

This program simply displays menu when overall information option is selected from data retrieval menu. Thin menu gives option for various test's overall results or results of all the students. According to the choice of options it calls programs sqry1,sqry2,sqry3 & sqrym.

```
clear all
clear
do while .t.
   STORE ' ' TO CHOICE
   @ 6,14 to 20,62 double
   @ 8,28 say '1. STUDENT  DATABASE.'
   @ 10,28 SAY '2. TEST1  RESULT.'
   @ 12,28 SAY '3. TEST2  RESULT.'
   @ 14,28 SAY '4. TEST3  RESULT.'
   @ 16,28 SAY '5. PREVIOUS MENU.'
   @ 18,28 SAY 'Enter your choice' GET CHOICE
   READ
   DO CASE
      CASE CHOICE = '1'
```

```
        DO SQRYM
       CASE CHOICE = '2'
           DO SQRY1
       CASE CHOICE ='3'
          DO SQRY2
       CASE CHOICE ='4'
          DO SQRY3
       CASE CHOICE = '5'
          CLEAR
          RETURN
       OTHERWISE
         @ 23,1 SAY 'Illegal Choice'
         ?? CHR (7)
         STORE 1 TO XX
         DO WHILE XX<35
          STORE XX+1 TO XX
          ENDDO
       @ 23,1 CLEAR TO 23,70
     ENDCASE
 ENDDO
* EOF RETDTA.PRG
```

4.3.16 PROGRAM SQRY.PRG.

It displays a menu with options of all students appeared test1, those passed test1 and those failed test1. This program works when 'test1 result' option is selected from overall information menu.

```
clear all
clear
do while .t.
   STORE ' ' TO CHOICE
   @ 6,18 to 16,58 double
   @ 8,28 say '1. ALL STUDENTS .'
   @ 9,28 SAY '2. THOSE ONLY.PASS.'
   @ 10,28 SAY '3. THOSE FAIL.'
   @ 11,28 SAY '4. PREVIOUS MENU.'
   @ 14,28 SAY 'Enter your choice' GET CHOICE
   READ
   DO CASE
      CASE CHOICE = '1'
       CLEAR
       USE ST1
       LIST
       @ 22,1 SAY ' '
       WAIT
       CLEAR
     CASE CHOICE = '2'
          CLEAR
```

```
          USE ST1
          LIST FOR MARK1 >= 60
          @ 22,1 SAY ' '
          WAIT
          CLEAR
      CASE CHOICE ='3'
         CLEAR
         USE ST1
         LIST FOR MARK1 < 60
         @ 22, 1 SAY ' '
         WAIT
         CLEAR
      CASE CHOICE = '4'
         CLEAR
         RETURN
      OTHERWISE
        @ 23,1 SAY 'Illegal Choice'
        ?? CHR (7)
        STORE 1 TO XX
        DO WHILE XX<35
         STORE XX+1 TO XX
         ENDDO
      @ 23,1 CLEAR TO 23,70
     ENDCASE
 ENDDO
* EOF SQRY1.PRG
```

4.3.17 PROGRAM SQRY2.PRG .

In the case of 'test2 result' option this program becomes active. It also gives options like previous program. But it works for test2.

```
clear all
clear
do while .t.
   STORE ' ' TO CHOICE
   @ 6,18 to 16,58 double
   @ 8,28 say '1. ALL STUDENTS .'
   @ 9,28 SAY '2. THOSE ONLY PASS.'
   @ 10,28 SAY '3. THOSE FAIL.'
   @ 11,28 SAY '4. PREVIOUS MENU.'
   @ 14,28 SAY 'Enter your choice' GET CHOICE
   READ
   DO CASE
      CASE CHOICE = '1'
       CLEAR
       USE ST2
       LIST
       @ 22,1 SAY ' '
```

```
        WAIT
        CLEAR
      CASE CHOICE = '2'
           CLEAR
           USE ST2
           LIST FOR MARK2 >= 60
           @ 22,1 SAY ' '
           WAIT
           CLEAR
       CASE CHOICE ='3'
          CLEAR
          USE ST2
          LIST FOR MARK2 < 60
          @ 22,1 SAY ' '
          WAIT
          CLEAR
       CASE CHOICE = '4'
          CLEAR
          RETURN
       OTHERWISE
         @ 23,1 SAY 'Illegal Choice'
         ?? CHR (7)
         STORE 1 TO XX
         DO WHILE XX<35
          STORE XX+1 TO XX
          ENDDO
       @ 23,1 CLEAR TO 23,70
      ENDCASE
 ENDDO
* EOF SQRY2.PRG
```

4.3.18 PROGRAM SQRY3.PRG .

This program works like previous one, but it works to retrieve date for test3, when 'test3 result' option is selected.

```
clear all
clear
do while .t.
   STORE ' ' TO CHOICE
   @ 6,18 to 16,58 double
   @ 8,28 say '1. ALL STUDENTS .'
   @ 9,28 SAY '2. THOSE ONLY PASS.'
   @ 10,28 SAY '3. THOSE FAIL.'
   @ 11,28 SAY '4. PREVIOUS MENU.'
   @ 14,28 SAY 'Enter your choice' GET CHOICE
   READ
   DO CASE
      CASE CHOICE = '1'
       CLEAR
```

```
        USE ST3
        LIST
        @ 22,1 SAY ' '
        WAIT
        CLEAR
      CASE CHOICE = '2'
           CLEAR
           USE ST3
           LIST FOR MARK3 >= 60
           @ 22,1 SAY ' '
           WAIT
           CLEAR
       CASE CHOICE ='3'
          CLEAR
          USE ST3
          LIST FOR MARK3 < 60
          @ 22,1 SAY ' '
          WAIT
          CLEAR
       CASE CHOICE = '4'
          CLEAR
          RETURN
       OTHERWISE
         @ 23,1 SAY 'Illegal Choice'
         ?? CHR (7)
         STORE 1 TO XX
         DO WHILE XX<35
          STORE XX+1 TO XX
          ENDDO
       @ 23,1 CLEAR TO 23,70
      ENDCASE
 ENDDO
* EOF SQRY3.PRG
```

4.3.19 PROGRAM SQRYM.PRG .

The function of this program is to display options. User can get information about all students, those passed all tests, appeared all tests and list of students those are authorized.

```
clear all
clear
do while .t.
   STORE ' ' TO CHOICE
   @ 6,18 to 16,58 double
   @ 8,28 say '1. ALL STUDENTS .'
   @ 9,28 SAY '2. THOSE PASS ALL TESTS.'
   @ 10,28 SAY '3. THOSE APPEAR ALL TESTS.'
   @ 11,28 SAY '4. PREVIOUS MENU.'
   @ 14,28 SAY 'Enter your choice' GET CHOICE
```

```
    READ
    DO CASE
       CASE CHOICE = '1'
        CLEAR
        USE SDENTM
        LIST
        @ 22,1 SAY ' '
        WAIT
        CLEAR
      CASE CHOICE = '2'
           CLEAR
           USE ST3
           LIST FOR MARK3 >= 60
           @ 22,1 SAY ' '
           WAIT
           CLEAR
       CASE CHOICE ='3'
          CLEAR
          USE ST3
          LIST
          @ 22, 1 SAY ' '
          WAIT
          CLEAR
       CASE CHOICE = '4'
          CLEAR
          RETURN
       OTHERWISE
         @ 23,1 SAY 'Illegal Choice'
         ?? CHR (7)
         STORE 1 TO XX
           DO WHILE XX < 50
          STORE XX+1 TO XX
          ENDDO
       @ 23,1 CLEAR TO 23,70
      ENDCASE
 ENDDO
* EOF SQRYM.PRG
```

4.3.20 PROGRAM SDENT.PRG.

This program becomes active when option 'student' is selected from main menu. It's function is to display various options for students. It displays options for concept learning, sample questions and tests. When concept learning option is selected it calls program learn1 and program st is called if test option is selected. One can also go back to main menu from this student menu.

```
CLEAR ALL
SELECT A
DO WHILE .T.
```

```
CLEAR
store ' ' to option
@ 4,10 TO 22,70 DOUBLE
@ 8,11 TO 8,69 DOUBLE
@ 6,25 SAY 'M E N U   F O R   S T U D E N T   '
@ 10,12 SAY 'OPTIONS :'
@ 12,20 SAY '   L            FOR  CONCEPT LEARNING '
@ 14,20 SAY '   S            FOR  SAMPLE QUESTION '
@ 16,20 SAY '   T            FOR  TEST'
@ 18,20 SAY '   E            FOR  EXIT TO MAIN MENU '
@ 20,12 SAY 'ENTER YOUR OPTION :' get option
read
STORE UPPER (OPTION) TO OPTION
  DO CASE
     CASE OPTION ='L'
       DO  LEARN1
     CASE OPTION = 'S'
       CLEAR
       USE SSAM
       SET MEMOWIDTH TO 65
       LIST OFF SAMPLE
       wait
       clear
     CASE OPTION = 'T'
       DO  ST
     CASE OPTION = 'E'
       RETURN
     OTHERWISE
     @ 23,1 SAY 'Illegal option, try again ...'
     STORE 1 TO XX
     ?? CHR (7)
   DO WHILE XX<35
        STORE XX+1 TO XX
     ENDDO
   @ 23,1 CLEAR TO 23, 50
  ENDCASE
ENDDO
*EOF SDENT.PRG
```

4.3.21 PROGRAM LEARN1.PRG .

The function of this program is simply display the menu for various level of lessons. It calls program learn for level-1 option, learn2 for level-2 and learn3 for level3.

```
clear all
clear
do while .t.
@ 6,20 to 20,52 double
```

```
@ 10,26 say '    1. LEVEL - 1'
@ 11,26 say '    2. LEVEL - 2'
@ 12,26 say '    3. LEVEL - 3'
@ 13,26 say '    4. PREVIOUS MENU'
STORE ' ' TO OPTION
@    16,24 SAY 'Enter your choice ' GET OPTION
READ
  DO CASE
     CASE OPTION='1'
       DO LEARN
     CASE OPTION='2'
       DO LEARN2
     CASE OPTION='3'
       DO LEARN3
     CASE OPTION = '4'
      CLEAR
      RETURN
     OTHERWISE
     @ 23,1 SAY 'ILLEGAL OPTION'
     ?? CHR (7)
 STORE 1 TO XX
  DO WHILE  XX < 50
   STORE XX+1 TO XX
   ENDDO
    @ 23,1 CLEAR TO 23,70
 ENDCASE
 enddo
* EOF LEARN1.PRG
```

4.3.22 PROGRAM LEARN.PRG .

The function of this program is to display contents of level-1 learning materials. This program also checks weather the user wants to continue learning or go back to previous menu. Pressing proper key user can move around the texts.

```
clear all
clear
SELECT A
USE LESSON1
SET MEMOWIDTH TO 65
store 65 to RR
LIST OFF CONTENT for code= CHR(RR)
kpress=0
DT = 0
DO WHILE  DT < 10
do while  kpress=0
   kpress = INKEY()
enddo
 DO CASE
```

```
   CASE kpress=18
   IF RR > 65
   store RR -1 to RR
   CLEAR
   LIST OFF CONTENT FOR CODE = CHR(RR)
   ELSE
     CLEAR
     RETURN
   ENDIF
   CASE kpress=3
      If RR < 71
       store  RR + 1 to RR
       CLEAR
       LIST OFF CONTENT FOR CODE = CHR(RR)
       ELSE
        CLEAR
        RETURN
      ENDIF
 OTHERWISE
   @ 23,1 say 'Please press proper key'
   ?? CHR (7)
    XX=1
    DO WHILE XX<50
     XX = XX+1
    ENDDO
  @ 23,1 CLEAR TO 23,70
 ENDCASE
 DT = DT + 1
kpress=0
ENDDO
* EOF LEARN.PRG
```

4.3.23 PROGRAM LEARN2.PRG .

The function of this program is to control the learning materials of level-2 when user selects option level-2 from lesson menu. Here user can also move around the texts. He can also go back to previous menu from anywhere of the text.

```
clear all
clear
SELECT A
USE LESSON2
SET MEMOWIDTH TO 65
store 65 to RR
LIST OFF CONTENT for code= CHR(RR)
kpress=0
DT = 1
DO WHILE DT < 10
do while kpress = 0
```

```
   kpress = INKEY()
enddo
 DO CASE
   CASE kpress=18
   IF RR > 65
   store RR -1 to RR
   CLEAR
   LIST OFF CONTENT FOR CODE = CHR(RR)
   ELSE
     CLEAR
     RETURN
   ENDIF
   CASE kpress=3
        IF  RR < 71
        store  RR + 1 to RR
        CLEAR
        LIST OFF CONTENT FOR CODE = CHR(RR)
        ELSE
         CLEAR
         RETURN
       ENDIF
   CASE kpress=6
        CLEAR
        RETURN
   OTHERWISE
   @ 23,1 say 'Please press proper key'
   ?? CHR (7)
   XX = 1
     DO WHILE XX <50
       XX = XX+1
     ENDDO
  @ 23,1 CLEAR TO 23,70
 ENDCASE
DT = DT + 1
kpress = 0
ENDDO
*EOF LEARN2.PRG
```

4.3.24 PROGRAM LEARN3.PRG .

This programs works like previous one. it becomes active when option level-3 is selected from lesson menu.

```
clear all
clear
SELECT A
USE LESSON3
SET MEMOWIDTH TO 65
store 65 to RR
LIST OFF CONTENT for code= CHR(RR)
```

```
kpress=0
DT = 1
DO WHILE DT < 25
do while kpress = 0
   kpress = INKEY()
enddo
 DO CASE
   CASE kpress=18
   IF RR > 65
   store RR -1 to RR
   CLEAR
   LIST OFF CONTENT FOR CODE = CHR(RR)
   ELSE
     RETURN
   ENDIF
   CASE kpress=3
      IF RR < 84
       store  RR + 1 to RR
       CLEAR
       LIST OFF CONTENT FOR CODE = CHR(RR)
      ELSE
       CLEAR
       RETURN
      ENDIF
   CASE kpress=6
       CLEAR
       RETURN
   OTHERWISE
   @ 24,1 say 'Please press proper key ...'
   ?? CHR (7)
    XX=1
      DO WHILE XX < 50
       XX = XX + 1
     ENDDO
  @ 24,1 CLEAR TO 24,40
 ENDCASE
 DT = DT + 1
 kpress = 0
ENDDO
*EOF LEARN3.PRG
```

4.3.25 PROGRAM ST.PRG FOR .

This is a simple program only to display options of various level of tests. It calls program logic for test1, logic2 for test2 and logic3 for test3. User can also go back to previous menu from here.

```
clear all
clear
DO WHILE .T.
```

```
@ 6,20 to 18,52 double
@ 10,26 say '    1. TEST1'
@ 11,26 say '    2. TEST2'
@ 12,26 say '    3. TEST3'
@ 13,26 SAY '    4. PREVIOUS MENU'
STORE ' ' TO OPTION
@   15,24 SAY 'Enter your choice ' GET OPTION
READ
  DO CASE
     CASE OPTION='1'
         DO LOGIC
     CASE OPTION='2'
         DO LOGIC2
     CASE OPTION='3'
         DO LOGIC3
     CASE OPTION ='4'
         CLEAR
         RETURN
     OTHERWISE
     @ 23,1 SAY 'ILLEGAL OPTION !'
     ?? CHR (7)
 STORE 1 TO XX
  DO WHILE XX<35
   STORE XX+1 TO XX
   ENDDO
   @ 23,1 CLEAR TO 23,70
 ENDDO
*EOF ST.PRG
```

4.3.26 *PROGRAM LOGIC.PRG .

When a student selects option test1 from test menu, at that time this program becomes active. Firstly it checks weather the student is authorized or not. IF student is not authorized system deliver a message. Otherwise it checks weather student already appeared for test1 or not. If appeared it again gives message to contact with teacher. If not, then it displays some message about test1 and questions of test1. After getting the answer for each question system also informs weather it is correct or not. At the end of examination it also display total marks acquired by student. Then it update all the related files.

```
clear all
clear
kpress = '       '
@ 10,10 say 'Enter your Roll No' get kpress
read
SELECT A
use sdentm index sss
seek  kpress
```

```
If found()
 CLEAR
 SELECT B
  USE ST1 INDEX SS1
  SEEK kpress
  IF FOUND()
 @ 5,10 say 'You already appeared in the examination ! '
 @ 22,1 SAY ' '
WAIT
  ELSE
   @ 5,22 SAY '         Welcome to Test1 !'
   @ 10,15 SAY 'Full marks of this test is 100. There are  5'
   @ 11,15 say 'questions in this test. You have to answer all'
   @ 12,15 say 'the questions. Each question carry 20 marks.The'
   @ 13,15 say 'PASS mark is 60. You have to pass this test for'
   @ 14,15 say 'appearing the next test.'
   @ 22,1 SAY ' '
   WAIT
*This part for display questions & get answers.
RR = 1
MARK = 0
AA = 1
DO WHILE  AA <= 5
CLEAR
SELECT c
USE QTION
set memowidth to 65
list off QUESTION for QNO = RR
QA = SPACE(1)
@ 15,5 say 'Enter choice. ' get QA
read
QA = upper(QA)
   use QANS INDEX QA1
   SEEK RR
     IF FOUND()
   IF QANS = QA  FOR QNO = RR
   MARK = MARK + 20
   @ 17,5 SAY 'Your answer is OK'
   ELSE
   @ 17,5 SAY 'Your answer is not OK'
   ENDIF
RR = RR + 1
AA = AA + 1
@ 22,1 SAY ' '
WAIT
ENDIF
ENDDO
clear
@ 5,8 say 'You have got          in this test.'
@ 5,23 say  str(MARK,3)
@ 22,1 say ' '
wait
*end of questions & answers.
* This part for adding records.
```

```
close databases
select B
use st1
append blank
goto bottom
replace ID with kpress
replace MARK1 with MARK
set index to SS1
reindex
clear all
* END OF ADDING RECORD
  ENDIF
 else
   CLEAR
   @ 15,5 say 'Sorry, you are not entitled for examination !'
   @ 22,1 SAY '  '
   WAIT
endif
*EOF LOGIC.PRG
```

4.3.27 *PROGRAM LOGIC2.PRG .

This program becomes active if option test2 is selected. It checks weather student authorized or not. If not, if gives message otherwise it checks weather the student appeared for test1 or not. If appeared, system checks weather he/she passed test1 or not. If he/she does not pass test1 it gives message to student for contacting with teacher. If he/she passes on test1, this program checks weather he/she appeared for test2 or not. If appeared it gives message to student. If not, it displays information about test2 and questions of test2. It also informs weather given answer is correct or not. At the end of test it shows marks and update all related files.

```
clear all
clear
kpress = '       '
@ 10,10 say 'Enter your Roll No' get kpress
read
SELECT A
use sdentm index sss
seek  kpress
If found()
 CLEAR
 SELECT B
 USE ST1 index SS1
     SEEK kpress
     IF FOUND()
         IF MARK1 >= 50 FOR ID = kpress
```

```
              SELECT C
              USE ST2 INDEX SS2
              SEEK kpress
              IF FOUND()
               CLEAR
@ 15,10 SAY ' You already appeared in the examination !'
@ 22,1 say ' '
              WAIT
              ELSE
               CLEAR
               @ 5,22 say 'Welcome to TEST2 '
               @ 8,10 SAY ' '
              TEXT
          Full marks of this test is 100. There are five questions
          in this test. Each question carry 20 marks. You have to
          answer all the questions. The  pass marks is 60. You must
          pass this test for appearing next test.
             ENDTEXT
              @ 22,1 SAY ' '
              WAIT
   *This part for display questions & get answers.
RR = 1
MARK = 0
AA = 1
DO WHILE  AA <= 5
CLEAR
SELECT D
USE QTION1
set memowidth to 65
list off QUESTION for QNO = RR
QA = SPACE(1)
@ 15,5 say 'Enter choice. ' get QA
read
QA = upper(QA)
   use QANS1 INDEX QA2
   SEEK RR
     IF FOUND()
   IF ANS1 = QA  FOR QNO = RR
   MARK = MARK + 20
   @ 17,5 SAY 'Your answer is OK'
   ELSE
   @ 17,5 SAY 'Your answer is not OK'
   ENDIF
RR = RR + 1
AA = AA + 1
@ 22,1 SAY ' '
WAIT
ENDIF
ENDDO
clear
@ 5,8 say 'You have got          in this test.'
@ 5,23 say str( MARK,3)
@ 22,1 say ' '
wait
```

88104

```
* END QUESTIONS
*This is the  program for adding record in st2 from test2.
close databases
select b
use st2
append blank
goto bottom
replace ID with kpress
replace MARK2 with MARK
set index to SS2
reindex
clear all
*EOF
            ENDIF
         ELSE
         @ 10,5 SAY 'You did not pass on TEST1, Please inform  to
your teacher.'
         @ 22,1 SAY ' '
         WAIT
         ENDIF
     ELSE
       CLEAR
       @ 16,5 SAY 'You did not appear on TEST1'
       WAIT
     ENDIF
 else
   CLEAR
   @ 15,5 say 'Sorry, you are not no entitled for examination !'
  @ 22,1 SAY  ' '
  WAIT
endif
*EOF LOGIC2.PRG
```

4.3.28 PROGRAM LOGIC3.PRG .

This program becomes active when option test3 is selected. It does all the works done by previous program. It additionally checks weather the user passed test2 or not. If passed on test2 and does not appear for test3 it gives permission for test3 and do same works as previous program.

```
clear all
clear
kpress = '       '
@ 10,10 say 'Enter your ID NO' get kpress
read
SELECT A
use sdentm index sss
seek  kpress
If found()
```

```
CLEAR
SELECT B
USE ST1 index ss1
    SEEK kpress
    IF FOUND()
        IF MARK1 >= 50 FOR ID = kpress
           SELECT C
           USE ST2 index ss2
           SEEK kpress
           IF FOUND()
              IF MARK2 >= 50 FOR ID = kpress
               SELECT D
               USE ST3 INDEX SS3
               SEEK kpress
                IF FOUND()
  @ 15,10 SAY 'You already appeared in the examination !'
  @ 22,1  SAY ' '
  WAIT
                ELSE
                @ 5,22 say 'Welcome to TEST3 '
                @ 10,15 SAY ' '
                TEXT
       This is the last test. If you pass in this test,  you will pass
       in this subject. The pass marks for this test is also 60.
       There are also 5 questions in this test, you have to answer all
       the questions.
                ENDTEXT

                @ 22,1 SAY ' '
                WAIT
       *This part for display questions & get answers.
RR = 1
MARK = 0
AA = 1
DO WHILE  AA <= 5
CLEAR
SELECT E
USE QTION2
set memowidth to 65
list off QUESTION for QNO = RR
QA = SPACE(1)
@ 15,5 say 'Enter choice. ' get QA
read
QA = upper(QA)
   use QANS2 index QA2
   SEEK RR
     IF FOUND()
   IF ANS2 = QA  FOR QNO = RR
   MARK = MARK + 20
   @ 17,5 SAY 'Your answer is OK'
   ELSE
   @ 17,5 SAY 'Your answer is NOT OK'
   ENDIF
RR = RR + 1
```

```
AA = AA + 1
@ 22,1 SAY ' '
WAIT
ENDIF
ENDDO
clear
@ 5,8 say 'You have got              in this test.'
@ 5, 23 SAY STR( MARK,3)
@ 22,1 say ' '
wait
*end of question & answer.
*This is the  program for adding record in st3 from test3.
close databases
select b
use st3
append blank
goto bottom
replace ID with kpress
replace MARK3 with MARK
set index to SS3
reindex
close all
*EOF adding record.
               ENDIF
                ELSE
                @  5,5  SAY  'You did not pass  on  TEST2,  Please
inform  your teacher.'
                @ 22,1 SAY ' '
                WAIT
                ENDIF
            ELSE
            @ 5,10 SAY 'You did not appear on TEST2'
            @ 22,1 SAY ' '
            WAIT
            ENDIF
          ELSE
          @  5,5 SAY 'You did not pass on TEST1, Please inform  to
your teacher.'
          @ 22,1 SAY ' '
          WAIT
          ENDIF
     ELSE
       CLEAR
       @ 5,5 SAY 'You did not appear on TEST1'
       @ 22,1 SAY  ' '
       WAIT
     ENDIF
 else
   CLEAR
   @ 15,5 say 'Sorry, you are not no entitled for examination !'
   @ 22,1 SAY  ' '
   WAIT
endif
*EOF LOGIC3.PRG
```

Chapter 5

RESULTS

5.1 INTRODUCTION

This chapter describes the basic data required for this package. The package uses this data and the results are presented in this chapter. During the application of this package, various situation are considered and these are simulated through the developed package. This chapter presents the corresponding results.

5.2 BASIC DATA.

This package requires three main types of data :

I) Students identification and his authorized teachers.

II) Text of the subject.

III) Model question and answers and test questions and their answers.

The above data files are named as students and teachers file, lesson file and question and answer file respectively.

5.2.1 Students and teachers file.

This file includes students name, ID no (Roll no) ID No

and a list of teachers names. The students as well as the teachers of this file will be the authorized users of this package. The authorization is implemented through password. That is a password is attached to each user. Only providing this password the user can enter into this CAI environment, otherwise not. Recall that in any case the student can not enter into the record files and some other facilities reserved for teachers.

5.2.2 Lesson file.

There are three level of lessons. Each lesson contains a sequential text, which helps the students to learn the subject in systematic fashion. These texts are Algebra of class six. In lesson-1, the use of literal symbols, comparison of literal symbols with numeric numbers, odd and even numbers, use of signs and related examples are presented.

Lesson-2 presents the use of brackets, simple addition and subtraction, simple multiplication, calculating the numeric value from a expression. All these conceptions are exemplified in this lesson.

In lesson-3 the explanation of directed numbers, addition and subtraction of directed numbers and multiplication of directed numbers are presented. Suitable examples of all of these options are also given in this lesson.

5.2.3 Question and Answer file.

There are three levels of test option in this package. Each level corresponds to a lesson. That is, for each lesson there is a set of questions stored in a question data file. Again for each question of each level there is only one correct answer. These answers are stored in the answer file.

5.3 APPLICATION.

This package is applied to some hypothetical students and teachers. That is the package is run considering that a student or a teacher is sitting in front of the keyboard. The package is used for various conditions.

At first, a student is considered and the package is run for

learning lessons, observing model questions and answer and appearing different level of tests.

Then, a teacher as an user is considered. It is also considered that the contents of various lesson files, question and its answer files are modified and edited by that teacher. Considering that a new student is admitted the appropriate edition of student's record file is made by the teacher.

5.4 RESULTS.

In this section, the results corresponding to different conditions of previous sections are presented.

First of all, a student is considered to sit in front of the computer. After typing CAI the main menu of the package displayed; it is presented in fig-5.01. Then pressing 'S' for student option, the student menu came on the screen. It in shown in fig-5.02. Pressing 'L' menu for various lessons came on the screen. Then pressing '1' the first lesson displayed on the screen and pressing key 'End' the student exited from lesson. It is shown in fig 5.03, 5.04, 5.05 respectively. After pressing '4'

student menu again came on the screen.

Selecting 'S' for sample questions from student menu, some sample questions with their answers came on the screen. After pressing a key student menu displayed on the screen. This is shown in fig 5.06, 5.07 & 5.08 respectively.

Then option 'T' for test was selected from student menu. In test1, first of all the package asked for an ID. An ID was given. However it was not in the student's record file. The system gave a message 'Sorry, you are not entitled for examination !'. This results are presented sequentially in Tables 5.09, 5.10 and 5.11.

Secondly another ID was given. The package delivered message 'You already appeared in the examination !'. The package gave such message because there was a record for test1 with that ID. This results are presented in the Tables 5.12, 5.13 and 5.14.

Thirdly an ID was given, which was not in the test1 records file. The package gave a message 'Welcome to TEST1' and some information about test1. After pressing a key package displayed questions one by one. After answering it also displayed whether the given answer is correct or not. There was five questions in test1. At the end of test1 it also total marks acquired by the student. all these results are presented in Tables 5.15, 5.16, 5.17, 5.18 and 5.19.

From teacher point of view, the package asked for password. After typing a password the package was displayed teacher menu. This is shown in table 5.21 and 5.22. Selecting option 'R' from teacher menu the data retrieve menu came on the screen. It is in table 5.23. From this data retrieve menu individual information of a student was observed before and after appearing for a test. It is shown in table 5.25 and 5.26. From data retrieve menu overall information was also observed before and after appearing for a test of a particular student. This is presented in tables 5.27, 5.28, 5.29, 5.30 and 5.31.

Table 5.01 Main Menu.

[In the starting of this package the main menu is displayed.]

```
                    M A I N     M E N U

OPTIONS :

              T        FOR  TEACHER

              S        FOR  STUDENT

              E        FOR  EXIT TO DOS


              Enter your option    S
```

Table 5.02 Student Menu.

[The selection of option S, Student menu displayed.]

```
                S T U D E N T      M E N U

OPTIONS :

                L       FOR  CONCEPT LEARNING

                S       FOR  SAMPLE QUESTION

                T       FOR  TEST

                E       FOR  MAIN MENU

                Enter your option     L
```

Table 5.03 Lessons Menu.

[The selection of option L, Lessons Menu was displayed.]

```
        LESSONS     MENU

TYPE :

        1    FOR  LEVEL - 1
        2    FOR  LEVEL - 2
        3    FOR  LEVEL - 3
        4    FOR  PREVIOUS MENU

        Enter your option    1
```

Table 5.04 Sample of Level1 text.
[After selecting option 1, text of level1 displayed.]

LEVEL - 1

INTRODUCTION.

1. THE USE OF LITERAL SYMBOLS. In Arithmetic we deal with numbers like 1, 2, 3,...1/2, 3/7,... etc. Here each number has a fixed definite value.

In Algebra, in addition to the use of these numbers, it is found convenient to use letters like a, b, c, ...,x, y, z to represent any numbers. The following examples illustrate this use;

Ex 1. A man's present age is 30.
His age 3 years hence will be 30 + 3 ;
" " n " " " " 30 + n.

The use of n in the last case generalizes the statement about his age.

Press Page Up to go back, Page Dn for next page,
End for lessons menu.

Table 5.05 Lessons Menu.
[Pressing key 'End' Lessons Menu was again displayed.]

LESSONS MENU

TYPE :

1 FOR LEVEL - 1
2 FOR LEVEL - 2
3 FOR LEVEL - 3
4 FOR PREVIOUS MENU

Enter your option 4

Table 5.06 Lesson menu.
[The selection of option 4, Student menu displayed.]

```
          S T U D E N T      M E N U

OPTIONS :

            L       FOR  CONCEPT LEARNING

            S       FOR  SAMPLE QUESTION

            T       FOR  TEST

            E       FOR  MAIN MENU

            Enter your option     S
```

Table 5.07 Sample Questions and their Answers.

[Again selecting S from student menu, sample questions and their answers were displayed.]

```
                    This is sample questions.

1.  A man's present age is 30 years. What was his age   x   years
ago ?

a)  30x     b)   30 - x     c)   x - 30     d)  x + 30

     Correct answer is b.  So you have to type letter b.

2.  If   x + 4 = 12       then   x = ?

a)   4       b)   12       c)   8        d)   16

     Correct answer is c. So you have to type letter c.

Press any key to continue...
```

Table 5.08 Student Menu.

[After pressing key 'Spacebar' student menu displayed again.]

```
                S T U D E N T      M E N U

OPTIONS :

                L       FOR  CONCEPT LEARNING

                S       FOR  SAMPLE QUESTION

                T       FOR  TEST

                E       FOR  MAIN MENU

                Enter your option      T
```

Table 5.09 Tests Menu.

[Selecting T from student menu, tests menu was displayed.]

```
                TESTS   MENU

TYPE :

     1   FOR  TEST1
     2   FOR  TEST2
     3   FOR  TEST3
     4   FOR  PREVIOUS MENU

     Enter your option   1
```

Table 5.10
[After typing 1, from tests menu, ID was asked.]

```
Enter your Roll No
```

Table 5.11

[Typing a Roll No 435 it gave message :-]

```
Sorry, You are not entitled for examination !


Press any key to continue...
```

Table 5.12

[After pressing key 'space bar' tests menu came on screen.]

```
        TESTS   MENU

TYPE :

     1   FOR  TEST1
     2   FOR  TEST2
     3   FOR  TEST3
     4   FOR  PREVIOUS MENU

     Enter your option   1
```

Table 5.13

[Again typing 1, package asked for Roll No]

```
Enter your Roll No.
```

Table 5.14

[Entering a roll no 8702f, package gave message]

```
You already appeared in the examination.


Press any key to continue...
```

Table 5.15

[Then pressing key 'space bar' tests menu came on screen.]

```
        TESTS   MENU

TYPE :

    1   FOR  TEST1
    2   FOR  TEST2
    3   FOR  TEST3
    4   FOR  PREVIOUS MENU

    Enter your option   1
```

Table 5.16

[Again selecting 1, system asked for roll no.]

Enter your Roll No

Table 5.17

[Entering a roll no 8804f, package gave the message :-]

Welcome to test1 !

Full marks of this test is 100. There are 5 questions in this test. You have to answer all these questions. Each question carry 20 marks. The PASS mark is 60. You must pass this test for appearing the next test.

Press many key to continue...

Table 5.18 Questions and answers.

[Pressing key 'spacebar' package displayed questions sequentially.]

QUESTION-1
****

A man's present age is 40 years. What was his age n years ago ?

a) 40n b) 40 - n c) n - 40 d) None of this.

Enter choice.

Letter **b** was typed, then system displayed message

Your answer is OK.

Press any key to continue...
[After pressing key 'space bar' it displayed second question.]

QUESTION-2
****

A rod is x feet long. If y feet is cut from one end and z feet from the other, what length is left ?

a) x - y + z b) x + y - z c) x - y - z d) x - yz

Enter choice.

Letter a was typed. system gave message

Your answer is not OK.

Press any key to continue...

[After pressing 'space bar'it displayed third question.]

QUESTION-3
****

The numbers 2, 4, 6, 8, 10, ... are called even numbers. What is the general even number ?

a) 2n b) 2 - n c) n - 2 d) 2n + 1

Enter choice.

Letter a was typed, system gave message

Your answer is OK.

Press any key to continue...

[After pressing key 'space bar' it displayed fourth question.]

QUESTION-4
****

The numbers 1, 3, 5, 7, 9, ... are called odd numbers. What is the general odd number ?

a) 2n b) n c) 2n + 1 d) n + 1 e) n - 1

Enter choice.

Letter c was typed, it gave message

Your answer is OK.

Press any key to continue...

[Pressing key 'space bar' system displayed last question.]

QUESTION-5
****

Find the value of 9a - 5, when a = 2.

a) 87 b) 13 c) 6 d) None of this.

Enter choice.

Letter b was typed, then it displayed message

Your answer in OK.

Press any key to continue...

Table 5.19

[Pressing key 'space bar' it displayed message]

```
You have got  80  in this test.
```

Press any key to continue...

[Pressing key 'space bar' it displayed student menu. And here was the end of test1.]

Table 5.20 Main Menu.

[Data was retrieved from main menu selecting teacher option.]

```
                    M A I N     M E N U

OPTIONS :

                T       FOR  TEACHER

                S       FOR  STUDENT

                E       FOR  EXIT TO DOS


                Enter your option    T
```

Table 5.21
[Selecting the option T, package asked for password.]

```
      Enter your password.
```

Table 5.22 Teacher Menu.
[After typing 'SIR' teacher menu was displayed.]

```
        M E N U    F O R    T E A C H E R

OPTIONS :

            U        FOR  UPDATE FILES

            R        FOR  RETRIEVE DATA

            E        FOR  EXIT TO MAIN MENU


            Enter your option     R
```

Table 5.23 Data Retrieve Menu.

[Selecting R from teacher menu, this menu came on the screen.]

```
              DATA RETRIEVE  MENU

TYPE :

     1   FOR  INDIVIDUAL INFORMATION
     2   FOR  OVERALL INFORMATION
     3   FOR  PREVIOUS MENU

     Enter your option   1
```

Table 5.24

[After selecting 1, package asked for student's Roll No.]

```
Enter students Roll No
```

[Roll No 8804f was entered.]

Table 5.25

[Before test1, result was]

STUDENT ROLL NO	STUDENT NAME	RESULTS OF TEST1	RESULTS OF TEST2	RESULTS OF TEST3
8804f	Shikder	NOT APPEARED	NOT APPEARED	NOT APPEARED
REMARKS: FAILED				

Table 5.26

[After test1, it displayed.]

STUDENT ROLL NO	STUDENT NAME	RESULTS OF TEST1	RESULTS OF TEST2	RESULTS OF TEST3
8804f	Shikder	80	NOT APPEARED	NOT APPEARED
REMARKS: FAILED				

Press any key to continue...

Table 5.27 Data retrieve menu.

[From data retrieve menu, option 2 was selected.]

```
                DATA RETRIEVE  MENU

TYPE :

     1   FOR  INDIVIDUAL INFORMATION
     2   FOR  OVERALL INFORMATION
     3   FOR  PREVIOUS MENU

     Enter your option   2
```

Table 5.28 Data retrieve menu1

[Then this menu came, after selecting option 2.]

```
                DATA RETRIEVE  MENU1

TYPE :

     1   FOR  STUDENT DATABASE
     2   FOR  TEST1 RESULT
     3   FOR  TEST2 RESULT
     4   FOR  TEST3 RESULT
     5   FOR  PREVIOUS MENU

     Enter your option   2
```

Table 5.29

[Selecting option 2 from Data Retrieve menu1 ,table 4.29 came on the screen. Option 1 was selected from this menu.]

```
TYPE :

    1   FOR  ALL STUDENTS
    2   FOR  THOSE ONLY PASSED
    3   FOR  THOSE FAILED
    4   FOR  PREVIOUS MENU

    Enter your option   1
```

Table 5.30

[Before Roll 8804f appearing test1 it gave information.]

ROLL NO	MARKS
8802p	60
8817f	40
8910f	80

Table 5.31

[After Roll No 8804f appearing test1 it gave information.]

ROLL NO	MARKS
8802p	60
8817f	40
8910f	80
8804f	80

Press any key to continue...

[Pressing key 'space bar' teacher menu came again on the screen.]

Chapter 6

CONCLUSION AND SUGGESTION

6.1 CONCLUSION.

A package has been developed for learning as well as teaching a subject through computer. It is completely menu driven. This package contents the syllabus of Algebra of class six. The package is used considering students and teachers as users for different conditions of learning and teaching. The results are intuitive. Replacing the contents of the lessons it can be used for any level of students.

6.2 SUGGESTION.

This package had not been implemented in a realistic system. This can be implemented in some schools to get the real effects of this package. Another extension of this package may be the implementation of Bangla text. In that environment students could be able to learn any subject through Bangla language. They could also be able to sit for examination in Bangla.

REFERENCES

[1] Chong H. Neng ,'An adjunct CAI for teaching geometry in sec ondary schools', AIT thesis, 1983.

[2] Uncharirleeporn Vongthongsre, 'Thai CAI for teaching mathematics', AIT thesis,1984.

[3] Ahmed Al-Halim , 'Microprocessor based student performance evaluation system', BUET thesis, 1986.

[4] Frenzel.L , 'The personnel computer -last chance for CAI.', Byte, McGraw-Hill Vol.5 No-7, pp, 86-96, July '80.

[5] Evans.S,'Interface Age', Sept.' 78 .

[6] Chambers J.A. & Sprecher J.R, 'Computer Assisted Instruction Current Trends & Critical Issues.', Communica tion of the ACM, Vol. 23 No 6, pp, 332-342,June 1980.