~~Hong Kong Certificate of Education Examination 2009~~
Computer and Information Technology Paper 3
(Coursework)

~~Module A~~
Algorithm and Programming

Multiple-Choice Analysis Report

Candidate ID No:

Content

Chapter 3 Design of Solution

Charter 4 Implementation

Charter5 Testing and Evaluation

Chapter 6 Conclusion & Discussion

Chapter 7 Reference and Acknowledgement

Appendix

Chapter1 Introduction

1.1 Background

Situation

An inter-school mathematics competition has jointly hold by tens secondary schools. All the questions are in the form of multiple-choice. After participants finished the questions, answer sheet will be read and converted into a text file using an Optical mark Recognition (OMR) and a programmer is going to work for the competition. One of the jobs for the programmer is to develop a computer program for which can read the converted answers, check all the answers, court the marks for each participants and produce a detailed analysis report for the competition.

Difficulties without the use of a program

Without using IT, we have to court the mark for the participants one-by-one and it is very inconvenience and time-consuming. Also, mistakes will easily occur if people do the work since checking multiple-choice is very difficult and confusing. If there is any mistake, the competition will become unfair and affect the results of the competition. Also, compare with checking by people, using a computer program can be faster

---

Benefits of using a program

Therefore, it would be judicious to use a computer program for the competition. Apart form effective or accurate, using a computer program can show the results in a clear way. User can check what they want to know easily. Also, the program can be reused in other multiple choice checking. It can also help the user to produce the analysis report easily.

1.2 Objectives

Situation of use and target user

By using this computer program, user can check the answer for the participant in an easier way. After input all the answers done by participants and also the correct answers, the computer will help the user to check all the answer with a very high speed. The target user of this program is the organizer, participants. For participants, the program is surely for them to check the results. But for the organizers, they can use the result to design different awards for the participants, for example, the best results for school, the best results for participant or over all champion, etc. Also this program can be used for all the schools. After the competition, the program can still run for different competition or homework, which is in multiple-choice form so that

User requirement and scope of project

Users just require basis knowledge of computer; they should know how to run this program and typing a few words. It is really easy so everyone is suitable for everyone to use it. Moreover, the user should provide a text file that includes all the answer of participants and another text that contenting the correct answers. Then the program will run and give out the results.

Purpose of the program

Apart form check all the answer for the participants, the program are used to produce a detailed analysis report for the mathematics competition.

The analysis report included:

- The total number of participants
- The total number of participating schools
- The results for the participants
- The results for the participating schools
- The percentage of the right answers for each questions
- The winners results
- Find percentage for one participants
- Find average for one school

1.3 Project Plan

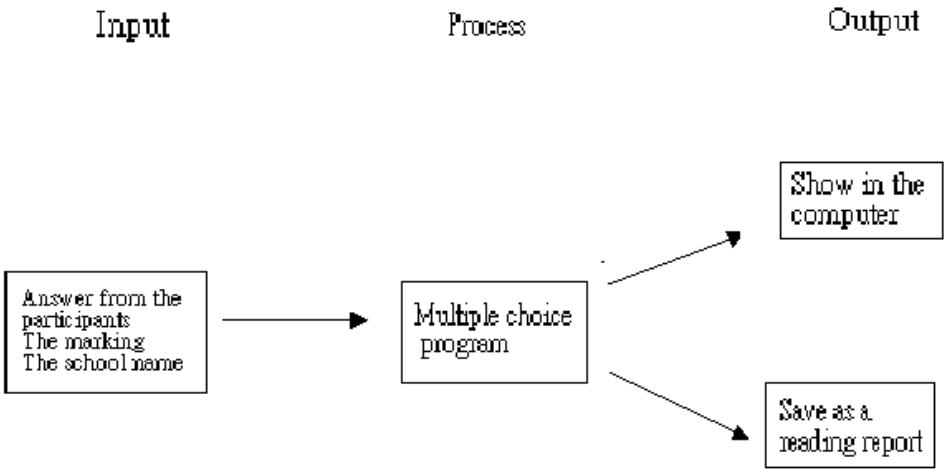
Analysis

In this multiple-choice competition, I plan to solve it in the following way. Firstly, for the questions, I design that there are several questions in the competition. For each questions, there are about two to five choices. If the participants can get the answers correct, they will get one mark. The participants who get the higher will have the higher opportunity to win the competition.

For the competition's awards, I design that there will be two awards. One is for the good performance participants. Another is for the good performance schools. And the number of participants and participating schools that may get the awards will be designed in the next charter.

In the report text file, I design that all the results of the competition should be included,
For example, 'the total number of participants', 'The total number of participating schools', ' The results for the participants', 'The results for the participating schools'……etc.

For the input, output and processing, I will solve it follow the under format:



Implementation

In this program, program language such as writeln, readln, repeat until , if then else or array of..Etc, will be used. Also, different procedures will be used. Although there may be lots of problems occur, if we solve it one-by-one patiently, it is not difficult for us to do a program that can use for this competition.

Testing and Evaluation

Debug is very important part for a successful program. For testing if there is any bug in the program, we can input some wrong input into it and check that if the program can run. For example we can input a wrong text file name or change the pattern of the text file. If the program cannot run, we can solve the error as quick as possible.

1.4 Conclusion and Discussion

The rough planning for the program is showed before. For any more planning or correction will be changed in the other charters. For further development, for example, the text file format, the format of questions, the regulation of the competition will be changed later.

Chapter 2 Analysis

2.1 Competition Regulations

participants and schools

The mathematics competition is jointly organized by ten secondary schools. There are overall thirty participants in these participating schools. Each school can choose how many students join this mathematics competition. In the competition, there is only one section that called section A with only one round. In this section, there are overall twenty multiple-choice questions with same level. And all the participants will answer the same set of questions. All the answer done by participants will be read and conversed into a text file using Optical Mark Reorganization (OMR) system.

Format of the answers

For the twenty multiple-choice questions, each question has five choices. They are A, B, C, D and E. Participants can choose one of the letters as his answers and write it on the answer sheet. After checking all the answer, if the participants can answer the questions correct, he will get one mark. On the other hand, no mark will be given for a wrong answer or keep the answer as a blank.

Awards

In this competition, there will be two awards. The first one is 'the individual award' for participant. The first three best perform participants will get this awards. Another one is 'the school award'. The award is set for the participating school. It is similar to the 'the individual award'. The first three participating schools, which have the highest average mark, will get these awards.

2.2 Data Collection

After the mathematics competition, all the answer done by participants will be read and conversed into a text file using Optical Mark Reorganization (OMR) system. Also a correct answer text file will also be made and store in the program..

Number of files

There will be four text file for my program. They are the participating school name text file, the text file name of participants, the answers text file for the participants and also the model answer text file. And the content of this text file will be showed below:

Text file (1): Participants name

This file will include all the name of the participants. Each line will only show one student name and it is followed the alphabetical order. Also the school number for them is following this order form one to thirty.

Text file (2): School name

This file will include the name of the participating schools. The other of school name is random. But the school number will follow this way form one to ten.

Text file (3): Answer

In this text file, apart form the participants, the student number, the school number where they come from with be showed. A space bar is used between each of them so that it is much more easy for program to read the data.
For example:

0001 01 ACBACABCAECBECABECAB

Text file (4): Marking

It is just a one line text consist of the model answer.

2.3 Inputs, Output and Process

For a program, there should be input, output and processing. And there are also these things for m program. And they will be described below.

The input data

For the input data, they are the four text files. They are ‘name.txt’, ‘schname.txt’, ‘ans.txt’ and ‘marking.txt’. Each of them will consist of different important data for the competition. The ‘name.txt’ content the name of the participants. The ‘schname.txt’ content the name of the participating schools. The ‘ans.txt’ content the answers, school number and also the student number. And lastly, the ‘marking.txt’ content the model answers.

The way of data input

The input data will be input by keyboard. It is because using keyboard to input data is the most convenience and effective way to input data.

The data which input by the keyboard included all the information that needed in the program. It means that the answers for each participants, which consist of six hundred, answers, the name of the participants which consist of thirty participants, the name of the participating schools which consist of ten schools. And also, the model answer should be input. It means that the twenty correct answers should be input.

The input data should be saved as a text file. It is because it is usually easy for a program to read a simple text file.

For the input data, plenty of text file is needed, One text file is for the name of the participants, one text file is for the name of the participating schools, and another text file is needed for the model answers. And lastly, one more text file is needed for the answers, which are done by the participants. Therefore, there will be four text file input

---

The output data

Apart from the basic acquired output data:
The total number of participants,
Total number of participating schools,
Total number of participant(s) from each participating school,
Winners of individual awards and school awards.

My program output data have added some more. These outputs can help the users to control the competition. They are:

- The total number of participants
- The total number of participating schools
- The results for the participants
- The results for the participating schools
- The percentage of the right answers for each questions
- The graph of the results for each questions
- The graph of the results for each participants
- The rank for the participants
- The rank for the participating schools
- The winners results
- Find percentage for one participants
- Find average for one school

Design

Overall plan or structure of the solution with indication of its major parts

0. Main body
    0.1 Get data
    0.2 Ask the user to type the text files
    0.3 Choose the case
        0.3.1. If choice is '1', show the number of participants and schools
        0.3.2 .If choice is '2'; show the results for the participants
        0.3.3. If choice is '3', show the results for the participating schools
        0.3.4. If choice is '4', show the percentage of the right answers for each question
        0.3.7. If choice is '7', show the rank for the participants
        0.3.8. If choice is '8', show the rank for the participating schools
        0.3.9. If choice is '9', show the winner's results
1. Gets data
    1.1. Open the file
    1.2. Copy the information
    1.3. Close the text file
2. Processing
    2.1. Procedure 'counting' to count the participants and schools
    2.2. Procedure 'checking' to check the answers
    2.3. Procedure 'quespercent' to count the percent for each question
    2.4. Procedure 'peopercnet' to count the percent for each participant
    2.5. Procedure 'studsort' to sort the student
    2.6 Procedure 'schsort' to sort the school
3. Output the program

Output format

The output data will print on the program. In the program, users can choose what information they want. There is a menu for the program. They can choose a number to display the results. For example, press <1> will show how many participants and how many participating school is in the mathematics competition.

Also, the report will be saved as a text file at the end of using this program.

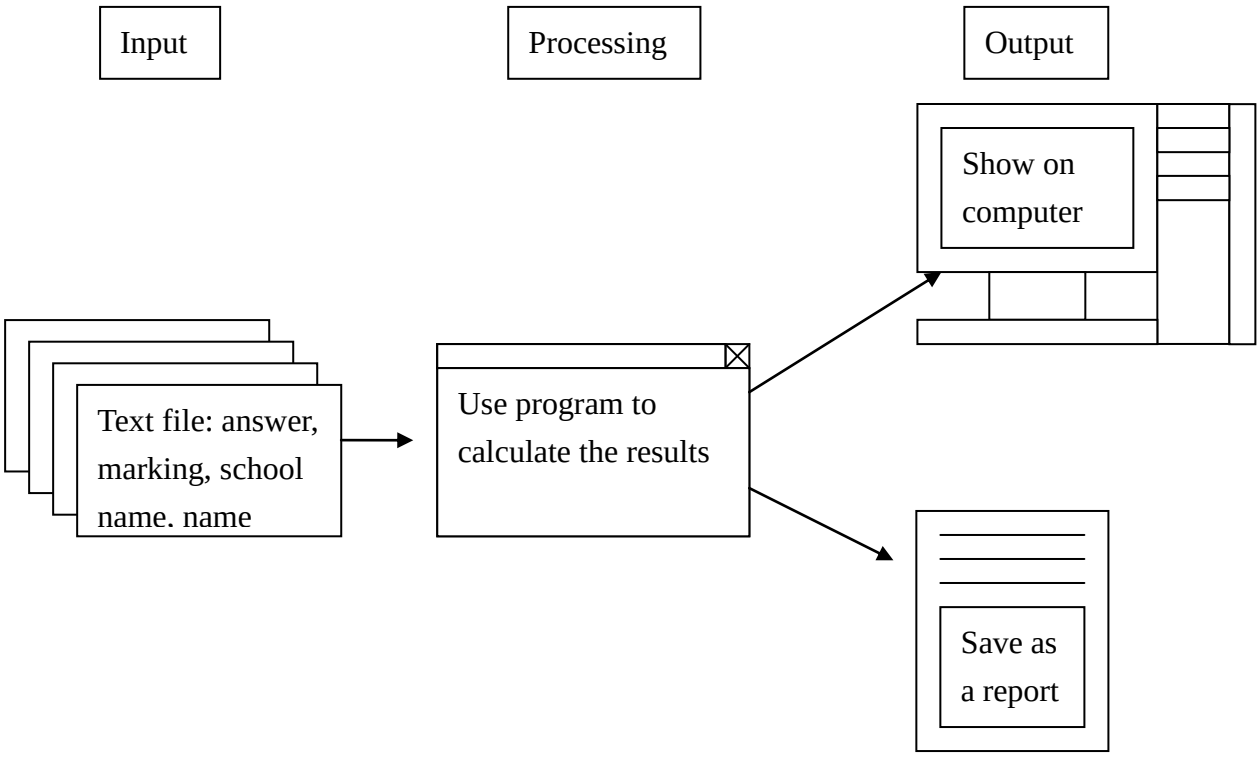
Reason for the output format

I think choosing by the users own is the best way to display the data. Because there is a lot of information in the mathematics competition. If all the output data is show at a time, it is much more badly than doing nothing. Oppositely, choosing what they want is a clear way to show the information.
Besides, the saved report is for the participants to bring away so that they can check the results anytime

---

Input format

The four text files, ‘name.txt’, ‘schname.txt’, ‘ans.txt’ and ‘marking.txt’ can be not a fixed name. The user can give the name for the file by them. After opening the program, the program will ask them to input the file name. The input file name should be the same as the name they given with a ‘ .txt ’ at the ending.



2.4 Choice of IT tool

Alternative approaches or IT resources

Apart from Pascal, there are still many different kinds of IT resources that can be used or writing the program, such as, flash, Asp. Net, html and java. And they consist of different advantages and disadvantage. Here is some for them:

Flash

It contains animation that is more attractable
It provides sound effects display pictures
It is more complicated to make
The program size is a bit large
Not everyone can stand for the format

Asp.net

It has debugging system
It provides you different kinds of functions, such as summit bar and calendar
However, it is also complicated to make
It occupies too many memory of the hard disk
It is difficult to find the bug
It is uses to make more serious program

Java

It is compatible for both IE and Netscape
Display format is more abundant
However, it is too slow since it uses virtual machine
The compiler is not efficient
Method overrides are not explicit which create a lot of bugs
It occupies too many memory of the hard disk
It contains too many packages and library so the scope is too wide

---

HTML (Hyper Text Markup Language)

Using a simple text editor can create it
You can easily edit HTML files using a WYSWYG (what you see is what you get) editor like FrontPage and Dream weaver, instead of writing your markup tags in a plain text file.
It has either htm or html file extension
However, it is difficult to find out the bugs

Pascal

It is plain and not suitable for serious programming
The program is east to install
The display format is simple and clear
It does not need to occupy a large hard disk memory of computer
The size of the product produces is small
It is a simple language for beginner to learn
It is suitable only for small, self-contained programs
It can check for invalid values and operations, by compile-and run-time checking

---

Array approaches

If we do not use array, the variable used can only hold one value at a time. It is too troublesome to correct the answers on-by-one. Therefore, array approaches is more efficient and the program code will be clearer to identify.

Procedure approaches

Each procedure represents a subprogram. Using procedures, a program with small modules is easier to follow and modify. We can concentrate on one small branch at a time and it I easier to debug. Moreover, it can be used in other programs.

Criteria of the situation

The Pascal programming is used in producing the results. Since comparing with the other IT resources, Pascal programming is an easier method to solve the problem. Although the product of Pascal is a bit simple and plain, it is clean and east to observe. Moreover, checking the answer does not need a too complicated programming. Besides, Pascal is easy to install and do not require too much memory of the computer and the product produce is also small in size. The program is also simple to use. Therefore, it is the most suitable IT sources for solve this problem.

2.5 Conclusion of Study

In conclusion, I will divide the program into three part, input, output and processing. There are four files needed for the output, the name, school name, answer and marking. For output, the data will be show in two ways. They will show on computer and also save as a text file report.

Besides, there are many way to solve the program. We can solve it by different IT tools. Different IT tools consist of their own advantages and disadvantage. But both of them can use to solve the program

Chapter 3 Design of Solution

3.1 Brief Description

Problem divides

To solve this problem, it is a good idea for me to divide the problem into different parts. Then I can solve them by different procedure. And lastly arrange the order of the procedure to solve the problem.

According to charter two, I have divided twelve outputs for users to choose. Therefore, I also divided the problem into similar parts:

- The total number of participants
- The total number of participating schools
- The results for the participants
- The results for the participating schools
- The percentage of the right answers for each questions
- The graph of the results for each questions
- The graph of the results for each participants
- The rank for the participants
- The rank for the participating schools
- The winners results
- Find percentage for one participants
- Find average for one school

We can just find out what is the problem I will encourage follow the above planning. After that, the problem will be divided and solved one-by-one successfully.

Planning of the program

I have designed to divide the problem into five main parts.

They are: 1. Read data file
2. Check the answers
3. Find the winner
4. Display the results
5. Save the report

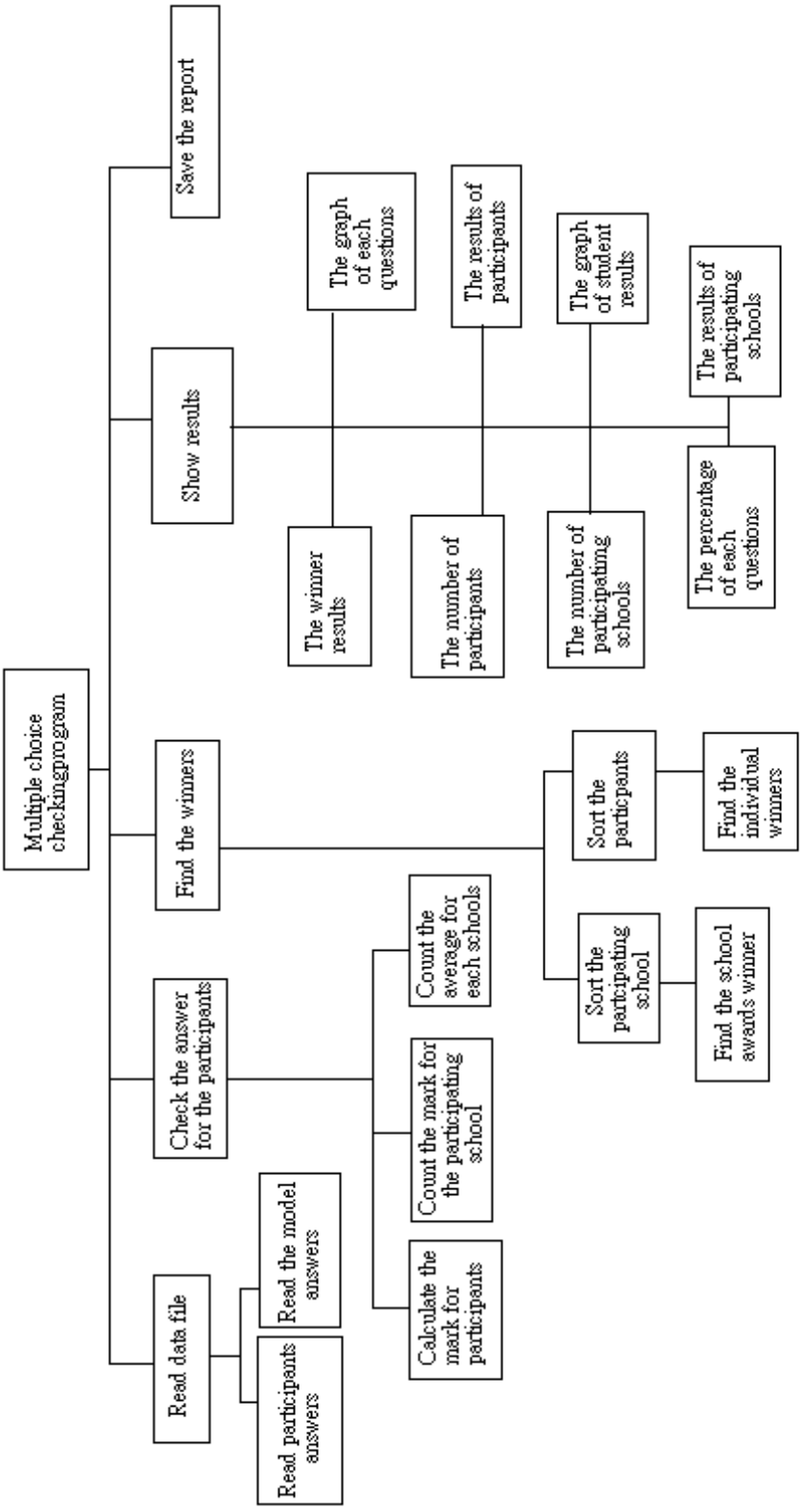
In these five sub-problems, each of them consists of about two to five procedure. And I will plan for it in the coming charter. There are many small sub-problems.

Block diagram

It is really inconvenience for me to show all of the sub-problem by using word only because it is not clear and difficult for me to understand the main point of different sub-problem.

In order to solve this problem, I am going to use a block diagram to show what the problems that I will encounter in this program. Using a block diagram is really convenience for me. It shows all the things clear. It is good for my planning on this program. And it is also good for me on the time doing debug because I can find out the relation between different procedure so that the time spent on debug will be therefore decrease,

The block diagram



3.2 Input data file format

As mentioned in charter before, the input for the program, I have designed it into four text files. They are the ‘name.txt’, ‘schname.txt’, ‘answer.txt’ and the last one ‘marking.txt’. Each of them consists of different information for the program.

Texts file (1): Participants name

This file will include all the name of the participants. Each line will only show one student name and it is followed the alphabetical order. Also the school number for them is following this order form one to thirty.

Name.txt

```
CHAN CHUN HEI
CHAN CHUNG WAI
CHAN CHUNG KIN
CHAN HOU WAI
CHAN KA HEI
CHAN KI TSUN
CHAN ON CHUN
CHAN PAK HANG
CHAN HO YIN
CHO YAN PING
CHU KIN FUNG
```

Texts file (2): School name

This file will include the name of the participating schools. The other of school name is random. But the school number will follow this way form one to ten.

School.txt

```
CCYCCC School
CCEEEME School
CDDDDMY School
CSWCSS School
ABCSSCC School
CCBAT School
FUYCE School
MYAAAS School
SSCCCC School
WYCCE School
```

Texts file (3): Answer

In this text file, apart form the participants, the student number, the school number where they come from with be showed. A space bar is used between each of them so that it is much more easy for program to read the data.

Student number (4 characters)	School number (2 characters)	Answer (20 characters)
0001	01	ABCECBECABDCC...
0004	02	ACECEDAACEDDC...

Answer.txt

```
0001 01 CABADAAAADEBBDDEBBCE
0002 01 CCBEDABCCDDEBBCABECB
0003 01 EBABADABCCACBABCABCC
0004 01 CBEDABCCDAEBBBCABDCE
0005 02 CCBEDABCACDBEBBCABEB
0006 02 CABEEDBCCADDEBBCABDC
0007 02 CCBEDABCCCDEBBCABECE
0008 03 CCBEDCBCCCDDBBBCABDD
0009 04 CCBEDCDEBCDADBBCABEC
0010 05 CCBEDABCCDDEBBBABAAA
```

Text file (4): Marking

It is just a one line text consist of the model answer.

Marking.txt

```
CCBEDABCCDDEBBCABECE
```

Input screen on running the program

On this page, the program will ask the user to type in the name of different text file. The program will ask the users to enter the input text file names one by one. The order is name of participants, name of participating schools, answer of the participants and lastly The marking answers. Here is the example:

```
'=========================================================
Please enter the file name of the student's names: _
```

After the users have typed in the name of the participant's name, the program will ask users to type in the others names one by one.

```
'=========================================================
Please enter the file name of the student's names: name.txt
Please enter the file name of the schools: school.txt
Please enter the file name of the students answer: answer.txt
Please enter the file name of the model answers: _
```

Output report format

In the output report, as in mentioned, there are two kinds of displays. The outputs will show on the computer and also save as a text file. They consist of different format.

Output shown on screen

After inputting the correct text file names, the program will go to a menu. Before going to the menu, there will be a welcome to the user.

On the menu, there will be:

1. A title
2. The number of participants and schools
3. The results for the participants
4. The results for the participating schools
5. The percentage of the right answers for each questions
6. The graph of the results for each question
7. The graph of the results for each participant
8. The rank for the participants
9. The rank for the participating schools
10. The winner's results
11. Searching a result for one participants
12. Searching a result for one participating school

Output results in Analysis Report file

The analysis report will be saved at the end of using this program. The program will ask the user to type in the name of the report. Then, the program will help the user to save it!

Inside the text report, all the results should be showed as mentioned in charter two with a clear handing. For example, a title < Report of Math Competition>.

In the report, it will consist:

1. A title
2. The number of participants and schools
3. The results for the participants
4. The results for the participating schools
5. The percentage of the right answers for each questions
6. The graph of the results for each question
7. The graph of the results for each participant
8. The rank for the participants
9. The rank for the participating schools
10. The winner's results

Besides, the Analysis Report file will be saving as a text file. Since text file is easier to handle by the programmer and also for the program.

Here is one of the examples.

<<Report of Math Competition>>

****

1. The number of participants and schools

The numbers of participating schools are 10

The numbers of participants are 30

==========

2. The results for the participants

Name	Correct answers	Percentage
CHAN CHUN HEI	9	45.0%
CHAN CHUNG WAI	19	95.0%
CHAN CHUNG KIN	3	15.0%

....

… (Different reports)

…

And the second runner up school is CCEEEME School and he gets 16 marks.

****

Charter 4 Implementation

4.1 Data structure

In the program, I have used a lot of variable for the counting. And most of the variables are array of format. Here is the variable I used,

---

Constant used

```
program MultipleChoiceProg;
 uses crt;
 const max = 200;
       maxsch = 30;
       numofquestions = 20;
```

Firstly, I have designed three constant defines.
Max: = 200 , Maxsch := 30 and Numofquestions := 20
Max means the maximum number of participant; Maxsch means the maximum number of the school when Numofquestions means the maximum number of questions.

Variables used

studnum : array [1..max] of integer
It means the student number of the participants. Each of the studnum[i] consists of one student number followed the order of the 'name.txt' text file.

schnum : array [1..max] of integer;
It means the school number of the participating school. Each of the schnum[i] consist of one school name followed the order of the 'school.txt' text file.

studans : array [1..max] of string[20];
It means the answers for the participants

name : array[1..max] of string[20];
It means the name of the participants

schname : array[1..maxsch] of string[20];
It means the name of the participating school

studcount : array [1..30] of integer;
It mean the student get how many marks

averagesort : array[1..max] of real;
It means the school average used for sorting

schnamesort : array[1..maxsch] of string;
It means the name of the school used for sorting

schmark : array[1..maxsch] of integer;
It means the total mark the school get

quecount : array[1..20] of integer;
It means each questions how many student correct

studsch : array[1..max] of integer;
It means the schools have how many participants.

average : array[1..maxsch] of real;
It means the school average.

quepercent : array[1..20] of real;
It means the percentage of answer correct for each questions

peopercent : array[1..30] of real;
It means the percentage of answer correct for each participants

sortstud:array[1..max] of integer;
It is the mark for each student used for sorting

sortname:array[1..max] of string;
It is the name of participant used for sorting

ans,iname,ischool,ianswer,imark: string;
They are the input file name

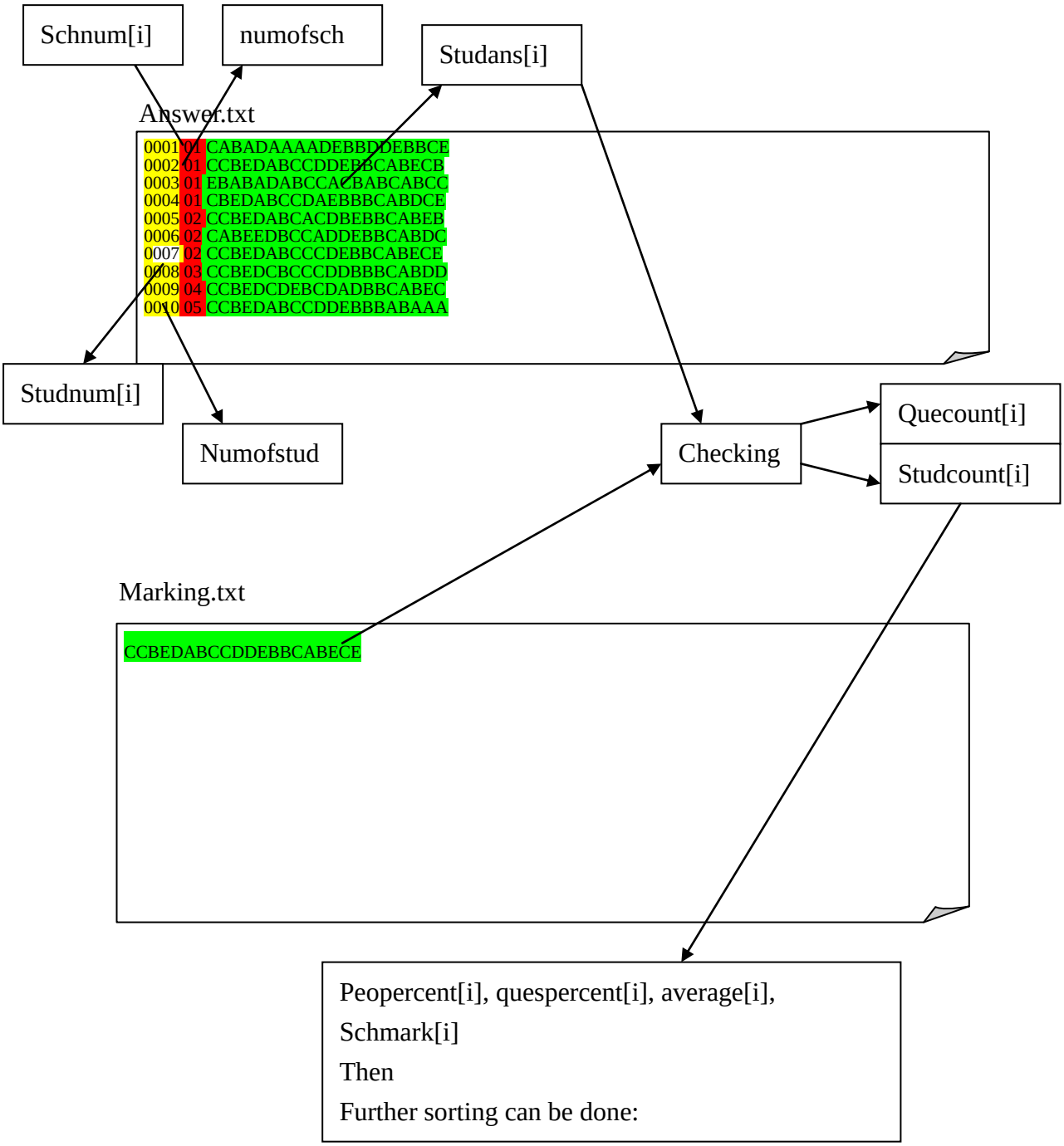
numofstud : integer;
It means the number of student

numofsch : integer;
it means the number of school

enternum : integer;
It means the number entered by the users in the menu

This variable can be very useful in my program. Most of them is use to save the input data.

The graph show how the variable is used



4.2 Procedures in the Program

There are a lot of procedures used in the program. A big program is therefore divided into sub-program. And it is much easier for me to solve the program.

1.0 procedure inputfile;

1.1 Ask user to enter the text file name

```
writeln('======================================================================');
write('Please enter the file name of the students names: ');
readln(iname);
write('Please enter the file name of the schools: ');
readln(ischool);
write('Please enter the file name of the students answer: ');
readln(ianswer);
write('Please enter the file name of the model answers: ');
```

1.2 Open the file

1.3 Read the data

```
 assign(schfile,ischool);
 reset(schfile);
 i := 0;
 while not eof(schfile) do
  begin
   i := i + 1;
   readln(schfile,schname[i]);
  end;
 assign(namefile,iname);
 reset(namefile);
 i := 0;
 while not eof(namefile) do
  begin
   i := i + 1;
   readln(namefile,name[i]);
  end;
 assign(studfile,ianswer);
 reset(studfile);
 i := 0;
 while not eof(studfile) do
  begin
   i := i + 1;
   readln(studfile,studnum[i],schnum[i],dummy,studans[i]);
  end;
 assign(anskey,imark);
 reset(anskey);
 while not eof(anskey) do
  read(anskey,ans);
 clrscr;
end;
```

1.4 Close the file

2.0 procedure participant;

2.1 Count the number of participant

2.2 Count the number of participating schools

```
for i := 1 to max do
 begin
  if (studnum[i] > 0)and (studnum[i] <> studnum[i+1]) and (studnum[1+1] <> 0 ) then
   numofstud:=numofstud+1;
  if (schnum[i] > 0) and (schnum[i] <> schnum[i-1])then
   numofsch := numofsch + 1;
 end;
```

3.0 procedure peoplechecking;

3.1 Copy the student answer

3.2 Copy the marking answer

3.3 Check the different between them and add mark

```
for n := 1 to numofstud do
 begin
  for i := 1 to numofquestions do
   begin
    modelans := copy(ans,i,1);
    studentans := copy(studans[n],i,1);
    if modelans = studentans then
     studcount[n] := studcount[n] + 1;
   end;
 end;
```

4.0 procedure questionchecking;

4.1 Copy the student answer

4.2 Copy the marking answer

4.3 Check the different between them and add mark

```
begin
 for q := 1 to 30 do
  begin
   for p := 1 to 20 do
    begin
       studentans := copy(studans[q],p,1);
       modelans := copy(ans,p,1);
       if (modelans = studentans) then
        quecount[p] := quecount[p] + 1;
    end;
  end;
```

4.4 Count the percentage for each question

```
for i := 1 to 20 do
 begin
  quepercent[i] := (quecount[i] / 30) * 100;
 end;
```

5.0 procedure peoplepercent;

5.1 Count the percentage for each participant

```
for i := 1 to 30 do
 peopercent[i] := (studcount[i]/numofquestions)*100;
```

6.0 procedure school_average;

6.1 Counting the number of participants from the schools
6.2 Counting the marks of the participants from that school get
6.3 Count the total mark the school gets
6.5 Count the average of the school

```
for i := 1 to numofsch do
 for n := 1 to max do
  if schnum[n]=i then
   begin
    schmark[i]:=schmark[i]+studcount[n];
    studsch[i]:=studsch[i]+1;
   end;
for i := 1 to numofsch do
 average[i]:=schmark[i]/studsch[i];
```

7.0 procedure peoplesorting;

7.1 Rewrite the student mark into a variable
7.2 Rewrite the student name into a variable

```
for i := 1 to numofstud do
 begin
  sortstud[i] := studcount[i];
  sortname[i] := name[i];
 end;
```

7.3 Rearrange the rank of them by using bubbles sort

```
for n := 1 to numofstud-1 do
 for i := 1 to numofstud-n do
  begin
   if sortstud[i] < sortstud[i+1] then
    begin
     biggermark := sortstud[i];
     sortstud[i] := sortstud[i+1];
     sortstud[i+1] := biggermark;
     biggername := sortname[i];
     sortname[i] := sortname[i+1];
     sortname[i+1] := biggername
    end;
   end;
```

8.0 procedure schoolsorting;

8.1 Rewrite the school average into a variable

8.2 Rewrite the school name into a variable

```
for i := 1 to numofsch do
 begin
  averagesort[i] := average[i];
  schnamesort[i] := schname[i];
 end;
```

8.3 Rearrange the rank of them by using bubbles sort

```
for n := 1 to numofsch-1 do
 for i := 1 to numofsch-n do
  begin
   if averagesort[i] < averagesort[i+1] then
    begin
     smallermark := averagesort[i];
     averagesort[i] := averagesort[i+1];
     averagesort[i+1] := smallermark;
     smallername := schnamesort[i];
     schnamesort[i] := schnamesort[i+1];
     schnamesort[i+1] := smallername
    end;
  end;
```

9.0 procedure savereport;

9.1 Ask the users to type in the file name

```
writeln('Enter the file name of the Report file : ');
writeln('(Please enter the file name within four words and with a .txt at the ending)');
writeln('(e.g. abcd.txt)');
readln(reportfilename);
```

9.2 Read all the results into that file

9.3 Tell the user that the file is saved

```
close(outfile);
writeln;
writeln('File save!');
writeln('Thank you for using this program');
writeln('Please press <enter> to leave');
readln;
```

9.4 End the program

10.0 procedure mainbody;

10.1 Call the procedure inputfile
10.2 Display the menu
10.3 Ask the user to choose one of them

```
write('Please enter a number<1-11> or any other number to quit : ');
delay(25);
readln(k);
```

10.3.1. If choice is '1', show the number of participants and schools
10.3.1.1 Press <enter> back to menu
10.3.2 .If choice is '2'; show the results for the participants
10.3.2.1 Press <enter> back to menu
10.3.3. If choice is '3', show the results for the participating schools
10.3.3.1 Press <enter> back to menu
10.3.4. If choice is '4', show the percentage of the right answers for
10.3.4.1 Press <enter> back to menu
10.3.5. If choice is '5', show the graph of the participant's results
10.3.5.1 Press <enter> back to menu
10.3.6. If choice is '6', show the graph of the questions
10.3.6.1 Press <enter> back to menu
10.3.7. If choice is '7', show the rank for the participants
10.3.7.1 Press <enter> back to menu
10.3.8. If choice is '8', show the rank for the participating schools
10.3.8.1 Press <enter> back to menu
10.3.9. If choice is '9', show the winner's results
10.3.9.1 Press <enter> back to menu
10.3.10. If choice is '10', search for a participant
10.3.10.1 Press <enter> back to menu
10.3.11. If choice is '11', search for a school
10.3.11.1 Press <enter> back to menu

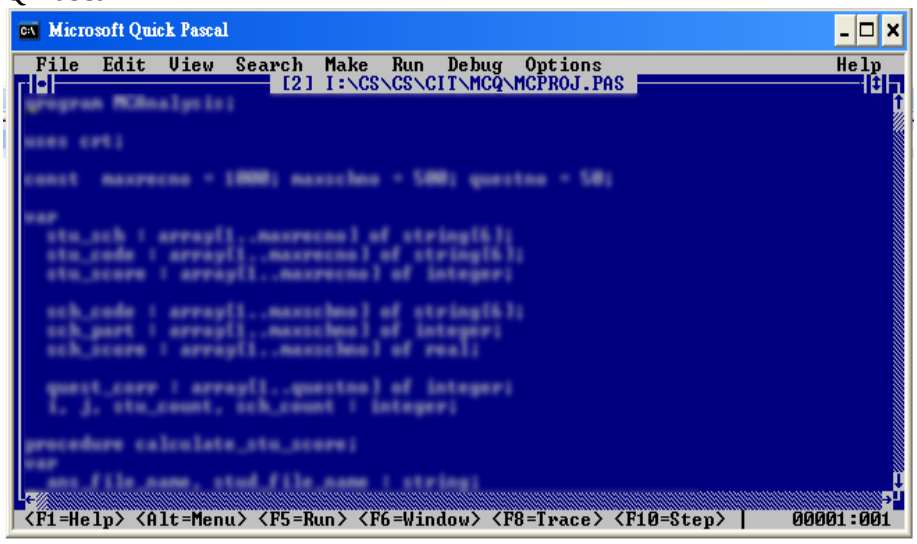
11.0 main body

11.1 Show a title
11.2 Call the procedure participant;
11.3 Call the procedure peoplechecking;
11.4 Call the procedure questionchecking;
11.5 Call the procedure peoplepercent;
11.6 Call the procedure school_average;
11.7 Call the procedure peoplesorting;
11.8 Call the procedure schoolsorting;
11.9 Call the procedure mainbody;
11.9. Call the procedure savereport;
11.10 End.

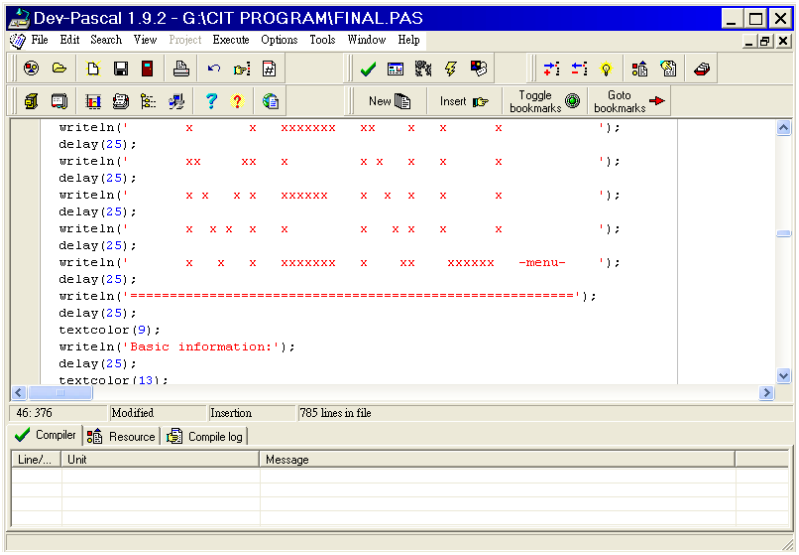
4.3 programs coding

In this program, I have used the Pascal to solve this program. The mainly type of Pascal I used is the Q-Pascal and Dev-Pascal. Because Pascal are easy for me to control and it is a very useful program.

Q-Pascal



Dev-Pascal

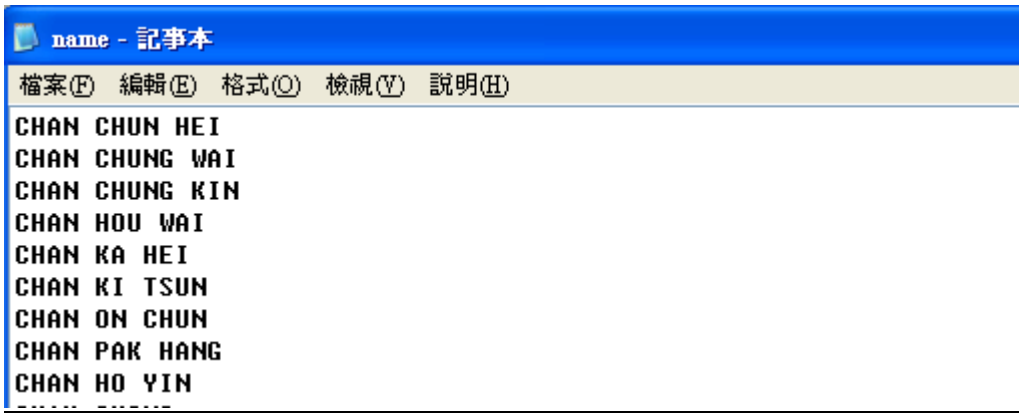


For the program coding, please refer to the Appendix 1 for the complete program code.

4.4 Program Execution

As mentioned in charter two and four, there are four text file in my program.

Text file (1): Participants name



This file will include all the name of the participants. Each line will only show one student name and it is followed the alphabetical order. Also the school number for them is following this order form one to thirty.

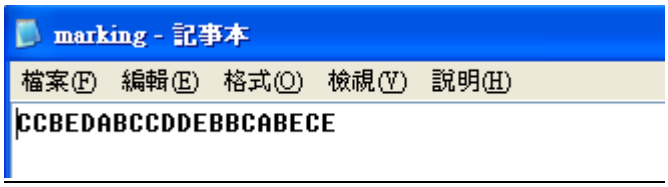
Text file (2): School name

This file will include the name of the participating schools. The other of school name is random. But the school number will follow this way form one to ten.

Text file (3): Answer

In this text file, apart form the participants, the student number, the school number where they come from with be showed. A space bar is used between each of them so that it is much easier for program to read the data.

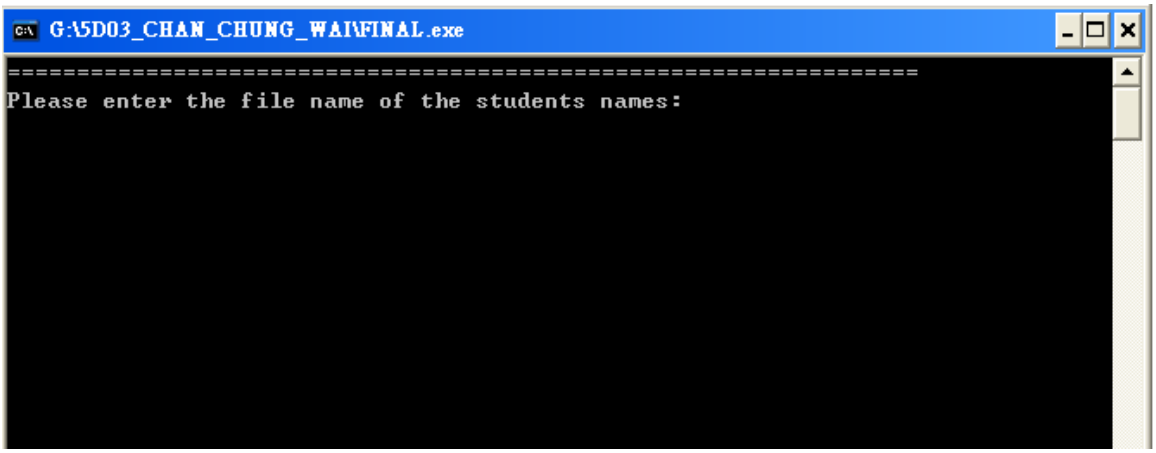
Text file (4): Marking



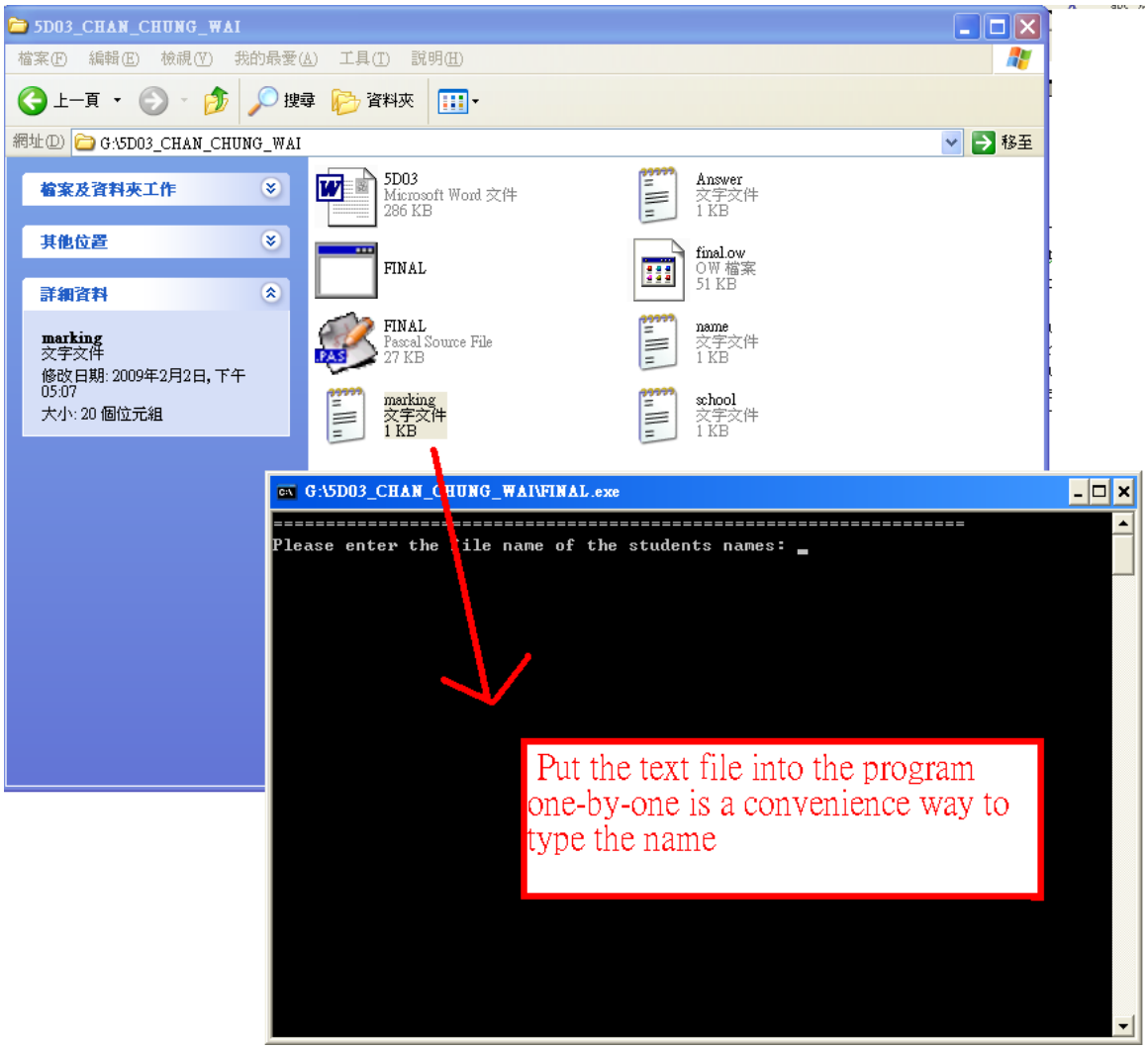
It is just a one line text consist of the model answer.

Use of program

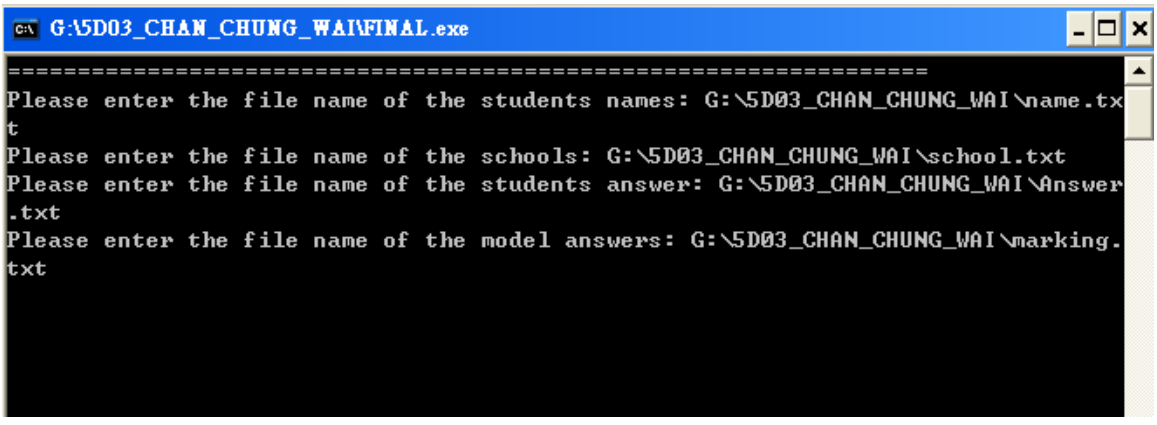
For the program, click the “Final” to open the program. Then the program will ask to type in the name of the text file one-by-one. The type in word should be included the full name, for example, ‘F: /ABC/Name ’



The way to input the file name:

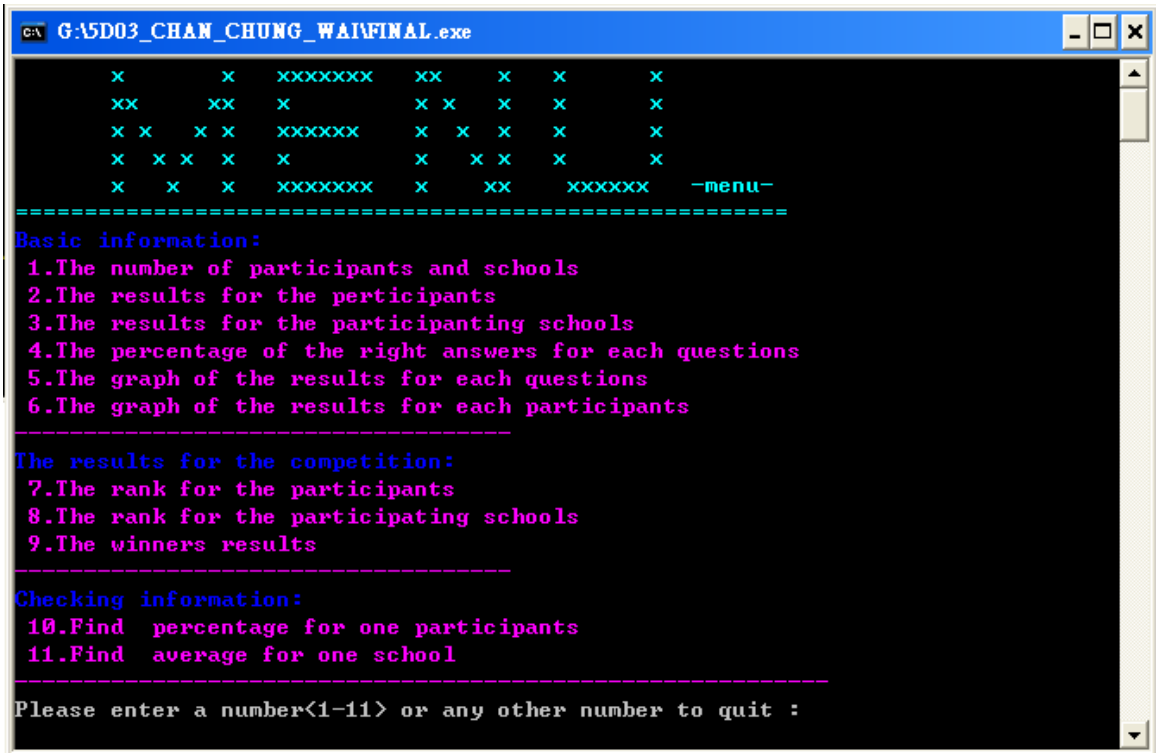


If the name of the textfile is typed by user one-by-one, there is a risk of typing wrong word and the program will therefore shut down. The more convenience and effective way to input is put the text file into the program. Then a full name will be typed in the program.



Then a cover of the program will be displayed. The user can then press <enter> go to the menu (procedure mainbody) of the program.

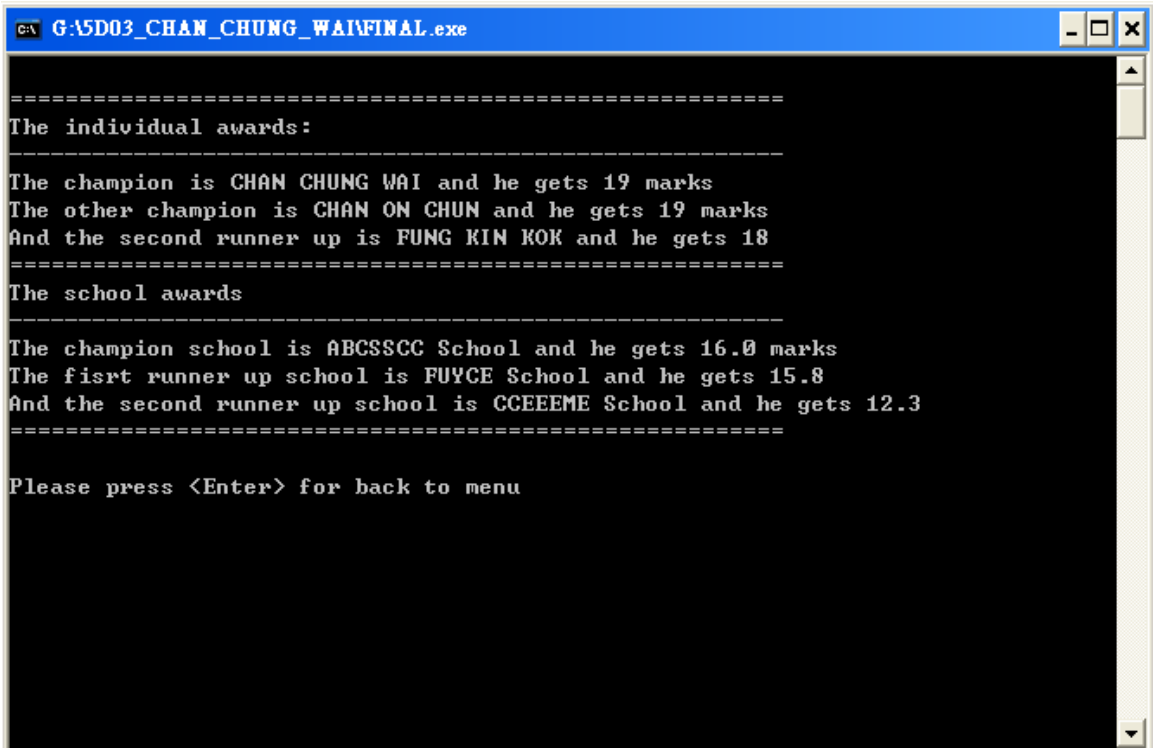
After pressing <enter>……



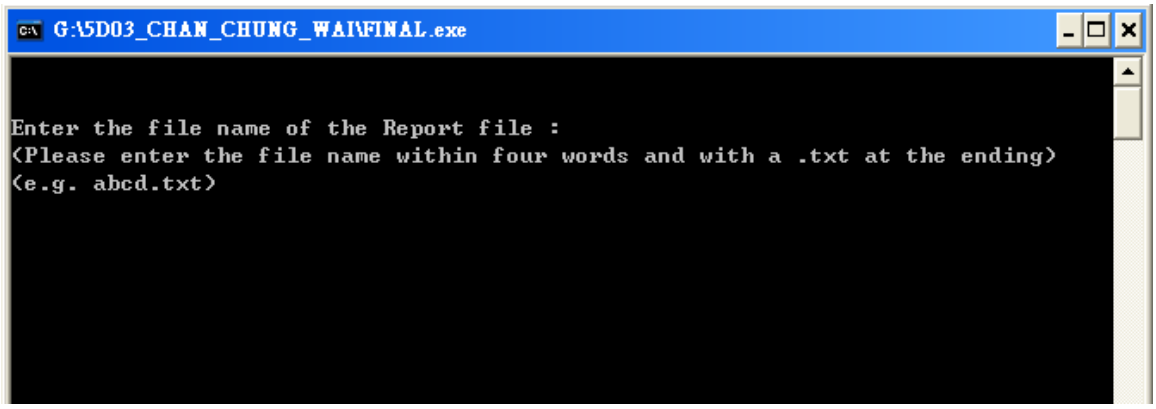
A menu will be displayed, user can then choose one of the choices(1-11) by himself.

Then, the user can choose one of them.

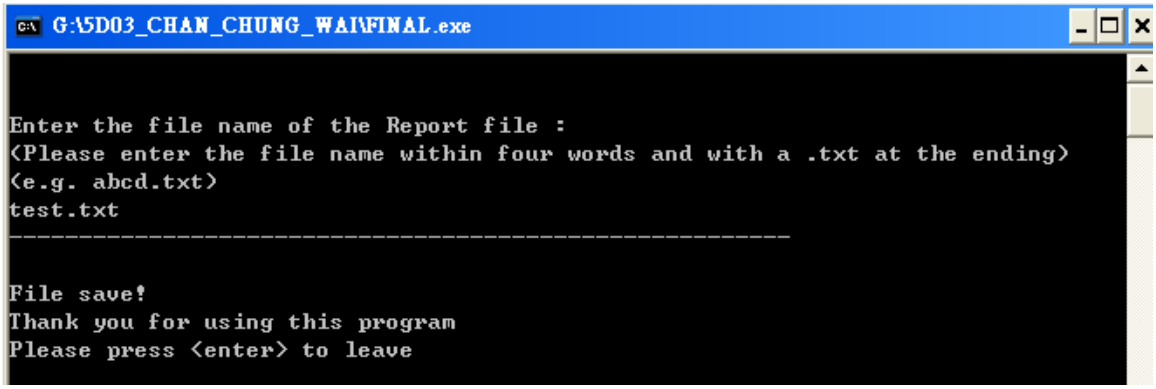
For example, if I press <9>, the winner results. The screen will show the competition results.



After the user has received all the results, he can press <enter> for back to menu. Besides, the way to quit is pressing any number except one to eleven in the menu. Then the program will help the user to save a report. Before that the program will ask the user to type in the file name.

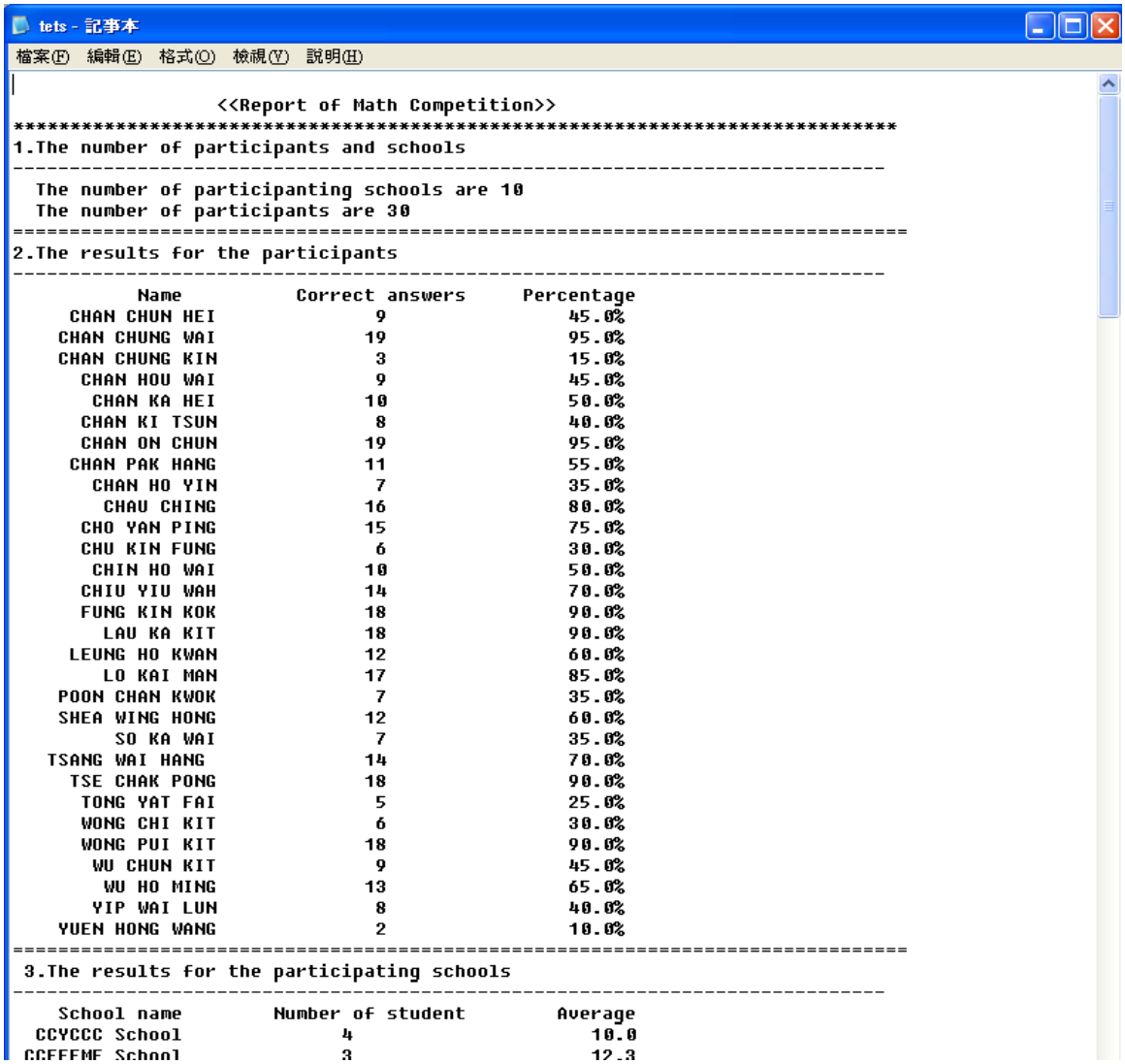


After that, the report will be saved!



The report file will then save in the file which the program is.

User can open it.



Charter5 Testing and Evaluation

5.1Testing for the syntax error

Menu choose	Results	After press <enter>	Solution
1.The number of participants and schools	Results showed successfully	Back to menu	
2.The results for the participants	Results showed successfully	Back to menu	
3.The results for the participating schools	Results showed successfully	Back to menu	
4.The percentage of the right answers for each questions	Results showed successfully	Back to menu	
5.The graph of the results for each questions	Results showed successfully	Back to menu	
6.The graph of the results for each participants	Results showed successfully	Back to menu	
7.The rank for the participants	**Shut down**	Back to menu	Change the For p:=1 to numofquestions To For p:=1 to numofstud
8.The rank for the participating schools	Results showed successfully	Back to menu	
9.The winners results	Results showed successfully	**Shut down**	Add ‘mainbody;’ so that the program can back to menu
10.Find percentage for one	Checking results showed	Can type ‘99’ to leave	

participants	successfully		
11.Find average for one school	Checking results showed successfully	Can type '99' to leave	

5.2 Testing for the logical error

Menu choose	Results	Solution
1.The number of participants and schools	No logical error occur after checking	
2.The results for the participants	No logical error occur after checking	

3.The results for the participating schools	**The last ten results are all '0' but not real average**	Change
		for i := 1 to numofquestions do
		to
		for i := 1 to numofsch do

4.The percentage of the right answers for each questions	No logical error occur after checking	
5.The graph of the results for each questions	**A strange graph is showed**	Using for…loop instead of repeat…until
6.The graph of the results for each participants	**A strange graph is showed**	Using for…loop instead of repeat…until
7.The rank for the participants	No logical error occur after checking	
8.The rank for the participating schools	**All the results are '0'**	A ';' is typed so that the program cannot gice a right resign
9.The winners results	No logical error occur after checking	
10.Find percentage for one participants	No logical error occur after checking	
11.Find average for one school	No logical error occur after checking	

Problem for menu

Although there are seem no problem in the menu, but if I type in a English or symbol, e.g. $%^@#$. The program will shut down. It is still a big problem. Maybe the way to solve is asked the users to only type number.

5.3 Testing the analysis report

Input name with no '.txt'

After entering a number of 0 (any number other than one to eleven), the program go to the save report page. Then I type in the text file name, 'test' without '. txt'. As results, a file saved but it is not a text file that cannot be opened.

Input name in Numbet.txt

After entering '123.txt', a text file with a name '123' showed in the file. Although the file can save, it is a good idea for the users to enter an English name.

Input name in English.txt

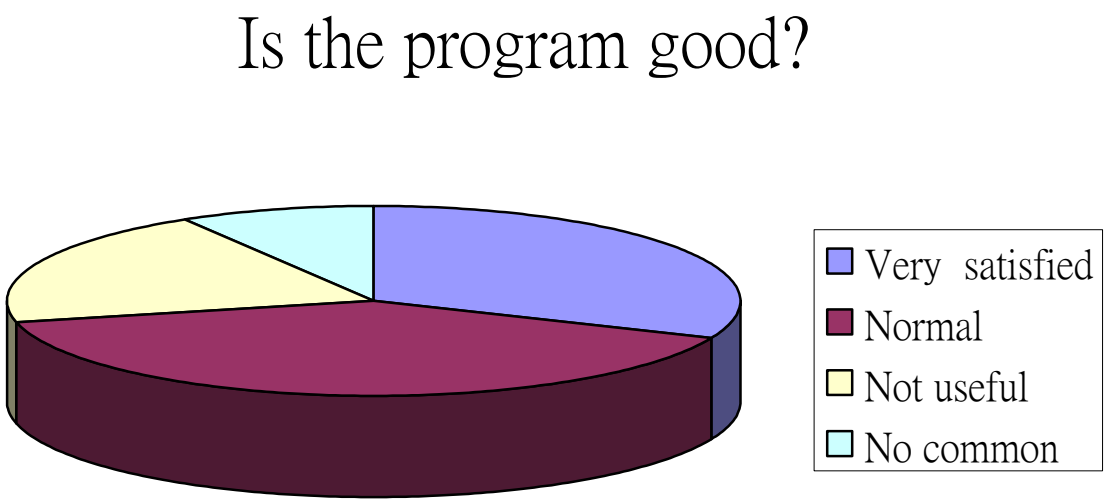
After entering 'test.txt', a text file with a name 'test' showed in the file.

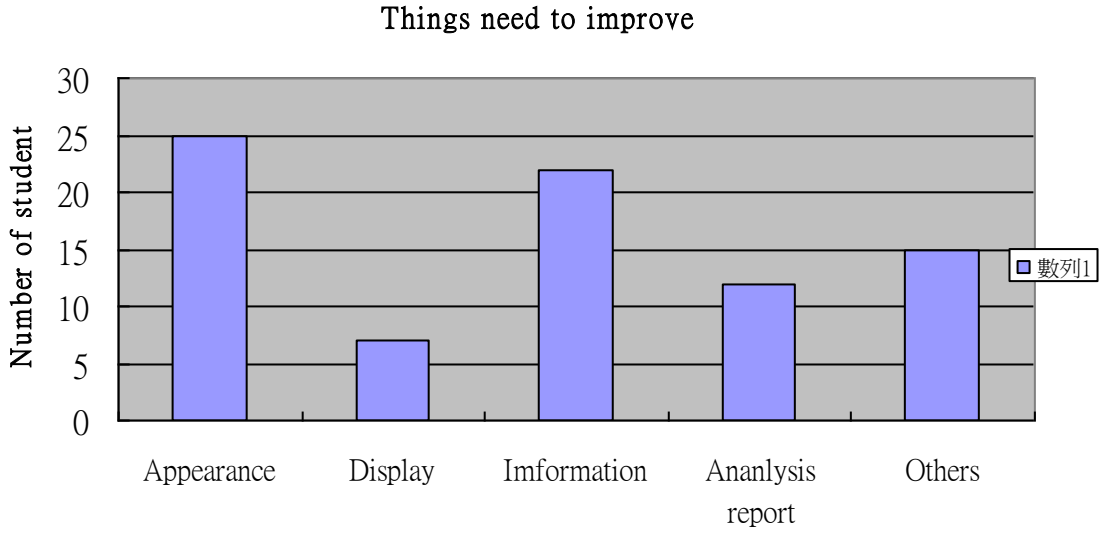
Error

I have tested save file for several time, there still some time have a error. The reason still cannot be found out yet. After I have typed in the file name, the file name can save but the program will back to choice'10. search a student 'information'. It is also a serious program that cannot be solved. It is still very difficult for me to find out which part having bug.

5.4 Replications

After the entire thing have been done and debug, I have give my project for classmates to try. I collect their opinion and do a char to show the results of their feeling about this program.





5.5 Analysis of the results for the questionnaire

In the questionnaire, i found that most of the classmate thought that my program is a normal one. And about fifteen percent of the student thought it is a satisfaction one. By the way, there are still a few of them thought it is not useful. For the thing need to improve, they said that the appearance and the information need to improve.

5.6 Correctness or errors based on the testing

After a several time of debug and correct, no error occurs during the testing of the program. Compile error does not exist since all the codes have been debugged and the program can be compiled successfully. It is also free of run time error since the program can run smoothly and will not terminate when there is invalid input. Finally, the program display results as same as expectation. Thus, there is no logical error.

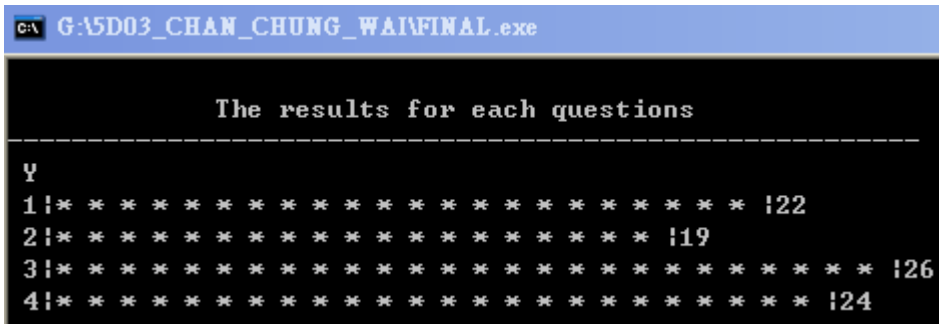
Chapter 6 Conclusion & Discussion

6.1 Pros and cons of my Program

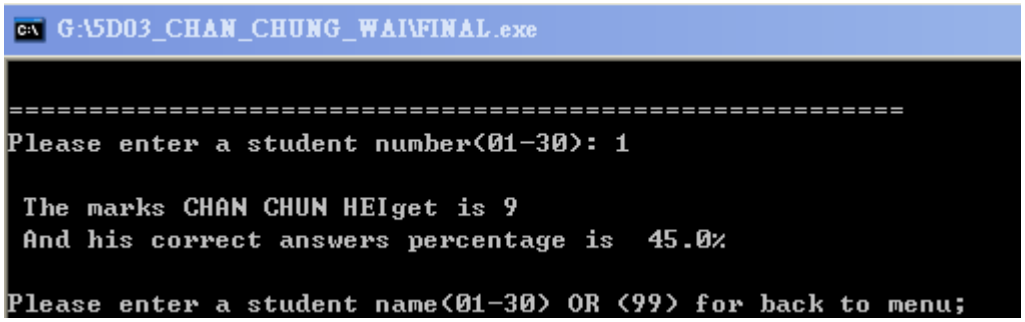
Pros

For advantages, my program's appearance is good so that all the information, guideline, results, etc… are showed clearly. Apart from the color change and the delay, I think the menu of my program is good. The entire choices for users to choose are showed clearly. After they have typed the number of the choices (No.1-11), the information they want will be show without the menu show. After that, they can press <enter> to back to the menu without the information stay.

For the choices, apart from the required one I have add some more. Firstly, the program can show the graph for the questions and the participants. Compare with the results showed in word, using a diagram is better. Using a diagram to show the results is much more attractive. Also, the results are showed more clearly. It is easier for users to compare the results.



Besides, there is a searching procedure to search the information for a participating school or participants. It is difficult and inconvenience for users to find out the information they want in a list of text. Also, it is very time-consuming but using this searching procedure, users can know what they want easily with a high speed.



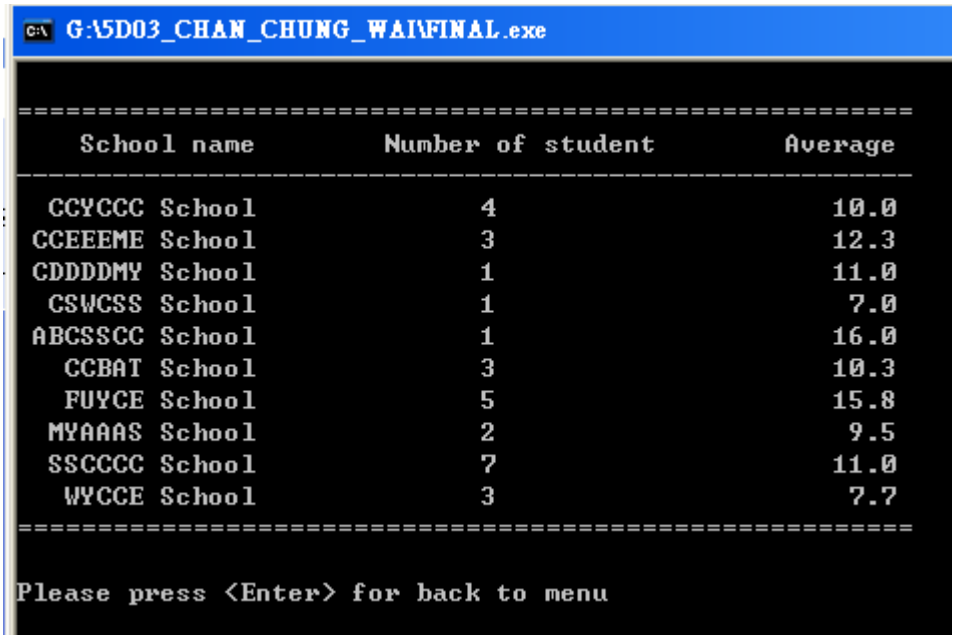
Last but not least, after users have used this program before they quit. The program will help them to save a text file for them. Inside the text file, all the information for this competition is showed. Then they can bring the information away and check it in their own computer.

Cons

There are plenty of advantages for this program, but there are still some disadvantages.

Firstly, the text files for this program are too many. In this program, four-text file are required. They are name, school, answer and marking. Compared with other program, my program needs much more text files. Sadly, the things inside the four text files are the same with the others. It means that my procedure on read file is longer than others and the time spend will be much more

For the information, one of the choices, ‘The results for the participants ’, is showed not clear because the report showed without the school name of the participants. There is also the same problem as ‘The results for the participants’ in the ‘The rank for he participants’ that no school name is showed. And for the choice ‘The results for the participating schools’, the results showed without the participants for that school. Therefore, the displaying format of my program seemed that is not good enough. The users cannot get a full data.



There is also a problem for the searching procedure of the participants and participating schools. Although the system can help the users to search the information they want, they can only type the number of the student or the school. If they only know the name for the one they should for without the number, they cannot get what they want.

Lastly, the award is also settled not well enough. In my idea, there are only three winners for the competition.. The winners can be one champion, first runner up and second runner up. For the other case, they can be two champions with a second runner up. Moreover, it can also be three champions. But there is still a logical error that has not been solved. If there are more than three champions or more than one second runner up, the first three participants and fast three participating school according the alphabetical order and the forth one cannot be showed.

6.2 Future Improvement.

In this program, I think the most important thing that needs to improve is the format of displaying information. The choice ‘the results for the participants ‘and ‘the rank for he participants’ should consist of the name of the school that the participants come form. It means adding a ‘School comes from’ on the line
between the ‘student name’ and ‘mark’ in the ‘results for the participants’. And add the same thing between the ‘student names’ and ‘mark’ in the ‘rank in the participants’. By adding that, the display of information will become more perfect.

For the problem for the searching procedure of the participants and participating schools, we are improving the project by allowing the users to type the participant’s name. At the same time, users can also type the school name since it is also a working way to search the information. But for the school name, there are three choices foe the user to type, the school number, the school name short form and the full name of the school. In my opinion, it is difficult for the programmer to change the program that allowing users to type this three format. And I think allow users to enter school number and short form of the school name is enough.

Moreover, I think the problem of school awards is a bit difficult to solve. Since the competition regulation has limit the number of winners, the program should follow the same way to find out the winners. It means that even there are more than three champion or more than two second runner up, there can be only three participants win the awards. For improvement, I think the questions in the competition can be change. Each question can get different marks. For example, from questions one to ten, participants can get one mark for each correct answer. But for questions eleven to twenty, participants can get to marks for each correct answer. So that it can decrease the probability of participants getting same marks.

6.3 Self-Reflection

For doing a Pascal program, I think the most important thing is that don't prefer that writing a program is very easy. I think most of the student just like me always starts writing a program code immediately after they receive the topic. They do the things without any planning; just do what they think at that time. However, this will just make us to spend more time on the project and make it less efficient. On the other hands, if you want to become faster and clear on expressing the idea, you should first write a flow chart to show that what data you want to get, how to process and output. Also, these can also prevent logical error produces.

On solving problems, I think it is the most difficult things for making a program. I think the time spending on debug is more than making a program. Since it is really difficult for me to find out a logical error and solve it. It is really a annoyed problem. But after doing this program, I know more about how to solve such a problem. Firstly, you should first think clearly what is the aim for this procedure/program and where is the problem come from. The more you know about the program, the easier for you to control and debug. Moreover, you should deal with the problems patiently. Do not think that you can solve it within a very short time. If you are really frustrates, you can ask teachers' or classmates' opinion or brain storming more. Do not hurry in finish the project. Last but not least, take a rest is also important. If you feel tired, you should take some rest to relax. When you come back to work, maybe you will fond out how to solve this problem!

Beside, doing a project can give us a chance to learn more. And also test ourselves if we have learned in the lesson. Although it is very hard to do a project on our own, when it is finished, you will feel very satisfied with what you have done. Moreover, we should not think that doing a project is a boring or a dull job. If you think is this way, you cannot do a good project and you will not gain anything from the project. Therefore, we should adopt a positive attitude towards this project.

'Practice makes perfect'. After finishing this project, I found that the skill of using Pascal has improved a lot. It shows that doing exercise is the best way for you to improve but not only read books. If I have a chance to a project again, I think I will be very heppy!

Chapter 7 Reference and Acknowledgement

From Internet websites

1. http://www.useful.com.hk/~abc
2. http://www.courseware.ust.hk/cantonese/pasframe.htm
3. http://hk.geocities.com/jack22_17/
4. http://www.heungto.edu.hk/~chau/filesharing.htm
5. http://hk.knowledge.yahoo.com
6. http://sheepdogsoftware.co.uk/pltut.htm
7. http://www.geocities.com/hotdogcom/ptutor/paslist.html
8. http://library.thinkquest.org/27297/

From Book

Appendix

1. Program coding

```
Appendicesprogram
 MultipleChoiceProg;

 uses crt;
 const max = 200;
      maxsch = 30;
      numofquestions = 20;
 var studnum : array [1..max] of integer;
     schnum : array [1..max] of integer;
     studans : array [1..max] of  string[20];
     name : array[1..max] of string[20];
     schname : array[1..maxsch] of string[20];
     studcount : array [1..30] of integer;
     averagesort : array[1..max] of real;
     schnamesort : array[1..maxsch] of string;
     schmark : array[1..maxsch] of integer;
     quecount : array[1..20] of integer;
     studsch : array[1..max] of integer;
     average : array[1..maxsch] of real;
     quepercent : array[1..20] of real;
     peopercent : array[1..30] of real;
     sortstud:array[1..max] of integer;
     sortname:array[1..max] of string;
     ans,iname,ischool,ianswer,imark: string;
     i,k,numofstud,numofsch,enternum : integer;
     dummy : char;

procedure inputfile;
 var namefile : text;
     studfile : text;
     anskey : text;
     schfile : text;
 begin

writeln('==============================================
================');
  write('Please enter the file name of the students names: ');
  readln(iname);
  write('Please enter the file name of the schools: ');
  readln(ischool);
  write('Please enter the file name of the students answer: ');
  readln(ianswer);
  write('Please enter the file name of the model answers: ');
  readln(imark);
  assign(schfile,ischool);
  reset(schfile);
```

```
  i := 0;
  while not eof(schfile) do
   begin
    i := i + 1;
    readln(schfile,schname[i]);
   end;
  assign(namefile,iname);
  reset(namefile);
  i := 0;
  while not eof(namefile) do
   begin
    i := i + 1;
    readln(namefile,name[i]);
   end;
  assign(studfile,ianswer);
  reset(studfile);
  i := 0;
  while not eof(studfile) do
   begin
    i := i + 1;
    readln(studfile,studnum[i],schnum[i],dummy,studans[i]);
   end;
  assign(anskey,imark);
  reset(anskey);
  while not eof(anskey) do
   read(anskey,ans);
  clrscr;
 end;

procedure participant;
 var i :integer;
 begin
  for i := 1 to max do
   begin
    if (studnum[i] > 0)and (studnum[i] <> studnum[i+1]) and
(studnum[1+1] <> 0 ) then
     numofstud:=numofstud+1;
    if (schnum[i] > 0) and (schnum[i] <> schnum[i-1])then
     numofsch := numofsch + 1;
   end;
 end;

procedure peoplechecking;
 var modelans,studentAns:string;
     n, i :integer;
 begin
  for n := 1 to numofstud do
   begin
    for i := 1 to numofquestions do
     begin
      modelans := copy(ans,i,1);
```

```
     studentans := copy(studans[n],i,1);
     if modelans = studentans then
      studcount[n] := studcount[n] + 1;
    end;
  end;
 end;


procedure questionchecking;
 var modelans,studentans:string[1];
    p,q,I :integer;
 begin
  for q := 1 to 30 do
   begin
    for p := 1 to 20 do
     begin
        studentans := copy(studans[q],p,1);
        modelans := copy(ans,p,1);
        if (modelans = studentans) then
         quecount[p] := quecount[p] + 1;
     end;
   end;
  for i := 1 to 20 do
   begin
    quepercent[i] := (quecount[i] / 30) * 100;
   end;
 end;


procedure peoplepercent;
 var i :integer;
 begin
  for i := 1 to 30 do
   peopercent[i] := (studcount[i]/numofquestions)*100;
 end;

 procedure school_average;
 var i,n :integer;
 begin
  for i := 1 to numofsch do
   for n := 1 to max do
    if schnum[n]=i then
     begin
      schmark[i]:=schmark[i]+studcount[n];
      studsch[i]:=studsch[i]+1;
     end;
  for i := 1 to numofsch do
   average[i]:=schmark[i]/studsch[i];
 end;

procedure peoplesorting;
```

```
 var biggermark : integer;
    biggername : string;
    i, n :integer;
 begin
  for i := 1 to numofstud do
   begin
    sortstud[i] := studcount[i];
    sortname[i] := name[i];
   end;
  for n := 1 to numofstud-1 do
   for i := 1 to numofstud-n do
    begin
     if sortstud[i] < sortstud[i+1] then
      begin
       biggermark := sortstud[i];
       sortstud[i] := sortstud[i+1];
       sortstud[i+1] := biggermark;
       biggername := sortname[i];
       sortname[i] := sortname[i+1];
       sortname[i+1] := biggername
     end;
    end;
 end;

procedure schoolsorting;
 var n,i : integer;
    smallername : string;
    smallermark :real;
 begin
  for i := 1 to numofsch do
   begin
    averagesort[i] := average[i];
    schnamesort[i] := schname[i];
   end;
  for n := 1 to numofsch-1 do
   for i := 1 to numofsch-n do
    begin
     if averagesort[i] < averagesort[i+1] then
      begin
       smallermark := averagesort[i];
       averagesort[i] := averagesort[i+1];
       averagesort[i+1] := smallermark;
       smallername := schnamesort[i];
       schnamesort[i] := schnamesort[i+1];
       schnamesort[i+1] := smallername
      end;
    end;
 end;

 procedure savereport;
 var reportfilename : string;
```

```
    i,m : integer;
    outfile : text;
 begin
  writeln;
  writeln('Enter the file name of the Report file : ');
  writeln('(Please enter the file name within four words and with
a .txt at the ending)');
  writeln('(e.g. abcd.txt)');
  readln(reportfilename);
  assign(outfile,reportfilename);
  rewrite(outfile);
  writeln(outfile);
  writeln(outfile,'               <<Report of Math
Competition>>');

writeln(outfile,'**************************************
**********************************');
  writeln(outfile,'1.The number of participants and schools');

writeln(outfile,'----------------------------------------
-----------------------------------');
  writeln(outfile,'  The number of participanting schools are
',numofsch);
  writeln(outfile,'  The number of participants are
',numofstud);

writeln(outfile,'======================================
====================================');
  writeln(outfile,'2.The results for the participants');

writeln(outfile,'----------------------------------------
-----------------------------------');
  writeln(outfile,'Name':15,'Correct
answers':25,'Percentage':15);
  for i := 1 to numofstud do

writeln(outfile,name[i]:18,studcount[i]:15,peopercent[i]:20
:1,'%');

writeln(outfile,'======================================
====================================');
  writeln(outfile,' 3.The results for the participating
schools');

writeln(outfile,'----------------------------------------
-----------------------------------');
  writeln(outfile,'School name':15,'Number of
student':25,'Average':15);
  for i := 1 to numofsch do

writeln(outfile,schname[i]:15,studsch[i]:15,average[i]:25:1
```

```
);

writeln(outfile,'==========================================
====================================');
  writeln(outfile,' 4.The percentage of the right answers for
each questions');

writeln(outfile,'------------------------------------------
-----------------------------------');
  writeln(outfile,'Questions':10, 'Answer
correct':20,'Percentage':20);
  for i := 1 to 20 do
     writeln(outfile,i:5,quecount[i]:20,quepercent[i]:20:1);

writeln(outfile,'==========================================
====================================');
  writeln(outfile,' 5.The graph of the results for each
questions');

writeln(outfile,'------------------------------------------
-----------------------------------');
  writeln(outfile,'           The results for each questions
');

writeln(outfile,'------------------------------------------
---------------');
  writeln(outfile,' Y');
  for i := 1 to numofquestions do
     begin
        write(outfile,i:2,'|');
        for m := 1 to quecount[i] do
          write(outfile,'* ');
         writeln(outfile,'|',quecount[i]);
     end;

writeln(outfile,'------------------------------------------
------------------- X');
  writeln(outfile,'Y-axis = questions number');
  writeln(outfile,'X-axis = number of students answer
correctly');

writeln(outfile,'==========================================
====================================');
  writeln(outfile,' 6.The graph of the results for each
participants');

writeln(outfile,'------------------------------------------
-----------------------------------');
  writeln(outfile,'         The results for each
participants');
```

```
writeln(outfile,'----------------------------------------
---------------');
  writeln(outfile,' Y');
  for i := 1 to numofstud do
     begin
      write(outfile,i:2,'|');
      for m := 1 to studcount[i] do
       write(outfile,'* ');
      writeln(outfile,'|',studcount[i]);
     end;

writeln(outfile,'----------------------------------------
------------------- X');
  writeln(outfile,'X-axis = Student number');
  writeln(outfile,'Y-axis = Number of questions answer
correctly');

writeln(outfile,'========================================
====================================');
  writeln(outfile,' 7.The rank for the participants');

writeln(outfile,'----------------------------------------
-----------------------------------');
  writeln(outfile,'Rank':10,'Name':20,'Mark':20);

writeln(outfile,'----------------------------------------
--------------');
  for i := 1 to numofstud do
     writeln(outfile,i:10,sortname[i]:20,sortstud[i]:20);

writeln(outfile,'========================================
====================================');
  writeln(outfile,' 8.The rank for the participanting
schools');

writeln(outfile,'----------------------------------------
-----------------------------------');
  writeln(outfile,'Rank':10,'School name':20,'Average':20);

writeln('----------------------------------------------
------');
  for i := 1 to numofsch do

writeln(outfile,i:5,schnamesort[i]:20,averagesort[i]:20:1);

writeln(outfile,'========================================
====================================');
  writeln(outfile,' 9.The winners results');

writeln(outfile,'========================================
==============');
```

```
  writeln(outfile,'The individual awards:');

writeln(outfile,'----------------------------------------
--------------');
  writeln(outfile,'The champion is ',sortname[1],' and he gets
',sortstud[1],' marks');
  if sortstud[1] = sortstud[2] then
   begin
    writeln(outfile,'The other champion is ',sortname[2],' and
he gets ',sortstud[2],' marks');
    if sortstud[2] = sortstud[3] then
     writeln(outfile,'And ',sortname[3],' is also the champion
and he gets ',sortstud[3],' marks')
    else
     writeln(outfile,'And the second runner up is
',sortname[3],' and he gets ',sortstud[3])
   end
  else
   begin
    writeln(outfile,'The fisrt runner up is ',sortname[2],' and
he gets ',sortstud[2]);
    if sortstud[2] = sortstud[3] then
     writeln(outfile,'And the other thirst runner is
',sortname[3],' and he gets',sortstud[3])
    else
     writeln(outfile,'And the second runner up is
',sortname[3],' and he gets ',sortstud[3])
   end;

writeln(outfile,'========================================
==============');
  writeln(outfile,'The school awards') ;
  writeln(outfile,'The champion school is ',schnamesort[1],'
and he gets ',averagesort[1]:5:1,' marks');
  if averagesort[1] = averagesort[2] then
   begin
    writeln(outfile,'The other champion school is
',schnamesort[2],' and he gets ',averagesort[2],' marks');
    if averagesort[2] = averagesort[3] then
    writeln(outfile,'And ',schnamesort[3],' is also the
champion school and he gets ',averagesort[3],' marks')
    else
     writeln(outfile,'And the second runner up school is
',schnamesort[3],' and he gets ',averagesort[3])
   end
  else
   begin
    writeln(outfile,'The fisrt runner up school is
',schnamesort[2],' and he gets ',averagesort[2]);
    if averagesort[2] = averagesort[3] then
     writeln(outfile,'And the other thirst runner school is
```

```
',schnamesort[3],' and he gets',averagesort[3])
   else
    writeln(outfile,'And the second runner up school is
',schnamesort[3],' and he gets ',averagesort[3])
  end;

writeln(outfile,'****************************************
****************************************');
 close(outfile);
 writeln;
 writeln('File save!');
 writeln('Thank you for using this program');
 writeln('Please press <enter> to leave');
 readln;
end;

procedure mainbody;
 var p,q,e,f,i :integer;
 begin
  textcolor(11);
  writeln('      x      x   xxxxxxx   xx    x   x     x
');
  delay(25);
  writeln('      xx    xx   x        x x   x   x     x
');
  delay(25);
  writeln('      x x   x x   xxxxxx    x  x  x   x     x
');
  delay(25);
  writeln('      x  x x  x   x        x   x x   x     x
');
  delay(25);
  writeln('      x   x   x   xxxxxxx   x    xx    xxxxxx
-menu-   ');
  delay(25);

writeln('==============================================
======');
  delay(25);
  textcolor(9);
  writeln('Basic information:');
  delay(25);
  textcolor(13);
  writeln(' 1.The number of participants and schools');
  delay(25);
  writeln(' 2.The results for the perticipants');
  delay(25);
  writeln(' 3.The results for the participanting schools');
  delay(25);
  writeln(' 4.The percentage of the right answers for each
questions');
```

```
  delay(25);
  writeln(' 5.The graph of the results for each questions');
  delay(25);
  writeln(' 6.The graph of the results for each participants');
  delay(25);
  writeln('-----------------------------------');
  delay(25);
  textcolor(9);
  writeln('The results for the competition:');
  delay(25);
  textcolor(13);
  writeln(' 7.The rank for the participants');
  delay(25);
  writeln(' 8.The rank for the participating schools');
  delay(25);
  writeln(' 9.The winners results');
  delay(25);
  writeln('-----------------------------------');
  delay(25);
  textcolor(9);
  writeln('Checking information:');
  delay(25);
  textcolor(13);
  writeln(' 10.Find  percentage for one participants');
  delay(25);
  writeln(' 11.Find  average for one school');
  delay(25);

writeln('-------------------------------------------------
---------');
  delay(25);
  textcolor(7);
  write('Please enter a number<1-11> or any other number to quit :
');
  delay(25);
  readln(k);
  clrscr;
  writeln;

  if k = 1 then
   begin
    writeln;

writeln('=================================================
======');
    delay(25);
    writeln(' The number of participating schools are
',numofsch);
    delay(25);
    writeln(' The number of participants are ',numofstud);
    delay(25);
```

```
writeln('================================================
======');
    delay(25);
    writeln;
    delay(25);
    writeln('Please press <Enter> for back to menu');
    readln;
    clrscr;
    mainbody;
   end;



  if k = 2 then
   begin

writeln('================================================
======');
    delay(25);
    writeln('Name':15,'Correct answers':25,'Percentage':15);
    delay(25);

writeln('----------------------------------------------
------');
    for q := 1 to 30 do
     begin

writeln(name[q]:18,studcount[q]:15,peopercent[q]:20:1,'%');
      delay(25);
     end;
    delay(25);

writeln('================================================
======');
    writeln;
    writeln('Please press <Enter> for back to menu');
    readln;
    clrscr;
    mainbody;
   end;

  if k = 3 then
   begin

writeln('================================================
======');
    delay(25);
    writeln('School name':15,'Number of
student':25,'Average':15);
    delay(25);
```

```
writeln('-------------------------------------------------
------');
    for i := 1 to numofsch do
      begin
       writeln(schname[i]:15,studsch[i]:15,average[i]:25:1);
       delay(10);
      end;
    delay(25);

writeln('=================================================
======');
    delay(25);
    writeln;
    delay(25);
    writeln('Please press <Enter> for back to menu');
    readln;
    clrscr;
    mainbody;
   end;

  if k = 4 then
   begin
    writeln('Questions':10, 'Answer
correct':20,'Percentage':20);
    delay(25);

writeln('-------------------------------------------------
------');
    for i := 1 to 20 do
     begin
      delay(10);
      writeln(i:5,quecount[i]:20,quepercent[i]:20:1);
     end;

writeln('=================================================
======');
    delay(25);
    writeln;
    delay(25);
    writeln('Please press <Enter> for back to menu');
    readln;
    clrscr;
    mainbody
   end;

  if k = 5 then
   begin
    writeln('           The results for each questions
');
    delay(25);
```

```
writeln('-------------------------------------------------
-------');
    delay(25);
    writeln(' Y');
    for i := 1 to numofquestions do
     begin
      delay(25);
      write(i:2,'|');
      for f := 1 to quecount[i] do
       write('* ');
       writeln('|',quecount[i]);
     end;

writeln('-------------------------------------------------
----------- X');
    writeln('Y-axis = questions number');
    writeln('X-axis = number of students answer correctly');

writeln('=================================================
======');
    writeln;
    writeln('Please press <Enter> for back to menu');
    readln;
    clrscr;
    mainbody
   end;

  if k = 6 then
   begin
    writeln('         The results for each participants');

writeln('-------------------------------------------------
-------');
    writeln(' Y');
    for i := 1 to numofstud do
     begin
      delay(25);
      write(i:2,'|');
      for f := 1 to studcount[i] do
       write('* ');
       writeln('|',studcount[i]);
     end;

writeln('-------------------------------------------------
----------- X');
    writeln('X-axis = Student number');
    writeln('Y-axis = Number of questions answer correctly');

writeln('=================================================
======');
```

```
    writeln;
    writeln('Please press <Enter> for back to menu');
    readln;
    clrscr;
    mainbody
   end;

  if k = 7 then
   begin

writeln('==================================================
======');
    writeln('Rank':10,'Name':20,'Mark':20);

writeln('--------------------------------------------------
------');
    for p := 1 to numofstud do
     begin
      writeln(p:10,sortname[p]:20,sortstud[p]:20);
      delay(10);
     end;

writeln('==================================================
======');
    writeln;
    writeln('Please press <Enter> for back to menu');
    readln;
    clrscr;
    mainbody
   end;

  if k = 8 then
   begin

writeln('==================================================
======');
    writeln('Rank':5,'School name':25,'Average':22);

writeln('--------------------------------------------------
------');
    for p := 1 to numofsch do
     begin
      writeln(p:5,schnamesort[p]:25,averagesort[p]:20:1);
      delay(10);
     end;

writeln('==================================================
======');
    writeln;
    writeln('Please press <Enter> for back to menu');
    readln;
```

```
    clrscr;
    mainbody
   end;

  if k = 9 then
   begin

writeln('==========================================================
======');
    writeln('The individual awards:');

writeln('----------------------------------------------------
------');
    writeln('The champion is ',sortname[1],' and he gets
',sortstud[1],' marks');
    if sortstud[1] = sortstud[2] then
     begin
      writeln('The other champion is ',sortname[2],' and he gets
',sortstud[2],' marks');
     if sortstud[2] = sortstud[3] then
      writeln('And ',sortname[3],' is also the champion and he
gets ',sortstud[3],' marks')
     else
      writeln('And the second runner up is ',sortname[3],' and
he gets ',sortstud[3])
     end
    else
     begin
      writeln('The fisrt runner up is ',sortname[2],' and he gets
',sortstud[2]);
     if sortstud[2] = sortstud[3] then
      writeln('And the other thirst runner is ',sortname[3],'
and he gets',sortstud[3])
     else
      writeln('And the second runner up is ',sortname[3],' and
he gets ',sortstud[3])
     end;

writeln('==========================================================
======');
    writeln('The school awards') ;

writeln('----------------------------------------------------
------');
    writeln('The champion school is ',schnamesort[1],' and he
gets ',averagesort[1]:3:1,' marks');
    if averagesort[1] = averagesort[2] then
     begin
      writeln('The other champion school is ',schnamesort[2],'
and he gets ',averagesort[2]:3:1,' marks');
     if averagesort[2] = averagesort[3] then
```

```
      writeln('And ',schnamesort[3],' is also the champion
school and he gets ',averagesort[3]:3:1,' marks')
     else
      writeln('And the second runner up school is
',schnamesort[3],' and he gets ',averagesort[3]:3:1)
    end
   else
    begin
     writeln('The fisrt runner up school is ',schnamesort[2],'
and he gets ',averagesort[2]:3:1);
     if averagesort[2] = averagesort[3] then
      writeln('And the other thirst runner school is
',schnamesort[3],' and he gets',averagesort[3]:3:1)
     else
      writeln('And the second runner up school is
',schnamesort[3],' and he gets ',averagesort[3]:3:1)
    end;

writeln('==============================================
======');
   writeln;
   writeln('Please press <Enter> for back to menu');
   readln;
   clrscr;
   mainbody
   end;

  if k = 10 then
   begin

writeln('==============================================
======');
   write('Please enter a student number(01-30): ');
   readln(enternum);
    while (enternum<1) or (enternum >30) do
     begin;
      write(' Wrong number!Please enter again: ');
      readln(enternum);
     end;
   writeln;
   writeln(' The marks ',name[enternum] ,'get is
',studcount[enternum]);
   writeln(' And his correct answers percentage is
',(studcount[enternum]/numofquestions)*100:0:1, '%');
   repeat
    begin
     writeln;
     write('Please enter a student name(01-30) OR (99) for back
to menu;');
     readln(enterNUM);
     if enternum = 99 then
```

```
      begin
       clrscr;

writeln('==================================================
======');
       mainbody;
      end;
      while (enternum<1) or (enternum >30) do
       begin;
        write('Wrong number!Please enter again: ');
        readln(enternum);
        if enternum = 99 then
         begin
          clrscr;

writeln('==================================================
======');
          mainbody;
         end;
       end;
      writeln;
      writeln(' The marks ',name[enternum] ,'get is
',studcount[enternum]);
      writeln(' And his correct answers percentage is  ',
(studcount[enternum]/numofquestions)*100:0:1,'%');
     end;
    until
     enternum = 999;
    end;

  if k = 11 then
   begin

writeln('==================================================
======');
    write('Please enter a school number(01-10): ');
    readln(enternum);
    while (enternum<1) or (enternum >10) do
     begin;
     write(' Wrong number!Please enter again: ');
     readln(enternum);
    end;
    writeln;
    writeln(' There is/are ',studsch[enternum], ' participant(s)
in ',schname[enternum]);
    writeln(' The marks ',schname[enternum] ,' get is
',schmark[enternum]);
    writeln(' And her average mark is  ',
average[enternum]:0:1);
    repeat
     begin
```

```
     writeln;
     write('Please enter a school number(01-10) OR (99) for back
to menu: ');
     readln(enterNUM);
     if enternum = 99 then
      begin
       clrscr;

writeln('==========================================================
======');
       mainbody;
      end;
     while (enternum<1) or (enternum >10) do
      begin;
       write('Wrong number!Please enter again: ');
       readln(enternum);
       if enternum = 99 then
        begin
         clrscr;

writeln('==========================================================
======');
         mainbody;
        end;
      end;
     writeln;
     writeln(' There is/are ',studsch[enternum], '
participant(s) in ',schname[enternum]);
     writeln(' The marks ',schname[enternum] ,' get is
',schmark[enternum]);
     writeln(' And her average mark is  ',
average[enternum]:0:1);
    end;
   until
    enternum = 999;
   end;

  if (k > 11) or (k < 1) then
  savereport;

end;

begin
 inputfile;
 participant;
 peoplechecking;
 questionchecking;
 peoplepercent;
 school_average;
 peoplesorting;
 schoolsorting;
```

```
  writeln;
  delay(50);
  textcolor(14);
  writeln('
------------------------------------------------------------
-------------');
  delay(50);
  textcolor(12);
  writeln('  |
|');
  delay(50);
  textcolor(14);
  writeln('  |                 X      X              XXXXX
|');
  delay(50);
  textcolor(12);
  writeln('  |                 XX    XX             X
|');
  delay(50);
  textcolor(14);
  writeln('  |                 X X   X X            X
|');
  delay(50);
  textcolor(12);
  writeln('  |                 X  X X  X            X
|');
  delay(50);
  textcolor(14);
  writeln('  |                 X   X   X            X
|');
  delay(50);
  textcolor(12);
  writeln('  |                 X      X              XXXXX
|');
  delay(50);
  textcolor(14);
  writeln('  |
|');
  delay(50);
  textcolor(12);
  writeln('  |     XXXXX  X    X  XXXXXX   XXXXX  X   X   XXXXX
X   X   XXXXX    |');
  delay(50);
  textcolor(14);
  writeln('  |    X      X    X  X      X      X  X     X
XX   X  X        |');
  textcolor(12);
  delay(50);
  writeln('  |    X      XXXXXX  XXXXXX  X      XXX      X
X X  X  X        |');
```

```
  textcolor(14);
  delay(50);
  textcolor(12);
  writeln('  |    X      X    X  X      X      X  X     X    X
X X  X  XXXX    |');
  delay(50);
  textcolor(14);
  writeln('  |    X      X    X  X      X      X   X    X    X
XX  X    X    |');
  delay(50);
  textcolor(12);
  writeln('  |     XXXXX  X    X  XXXXXX   XXXXX  X    X  XXXXX
X    X   XXXXX    |');
  delay(50);
  textcolor(14);
  writeln('  |
|');
  delay(50);
  textcolor(12);
  writeln('  |
|');
  delay(50);
  textcolor(14);
  writeln('
-------------------------------------------------------------
-------------');
  delay(50);
  writeln;
  delay(50);
  textcolor(12);
  writeln('   Welcome to use this program !');
  delay(50);
  textcolor(14);
  writeln('   Please press <Enter> to continue');
  delay(50);
  write('   ');
  readln;
  clrscr;
  mainbody;
  writeln;
 end.
```

Appendix 2 working schedules

Date	Event
3rd January	Start planning for the project
4th January	Start planning for the procedure for the project
10th January to 14th January	Doing report charter one (text only)
	Doing report charter two (text only)
15th January To 20th January	Typing the procedure into the Pascal
20th January to 22nd January	Checking and Debug until the program can run (Keeping debug if discover)
25th January to 3rd February	Doing report charter three (text only) Doing report charter four (text only)
4th February to 10th February	Doing report charter five (text only) Doing report charter six (text only)
11th February to 13th February	Adding photo into the report