[ASM] Students Performance Management System

Date : 10/06/2013 - 15/06/2013

Tutor : Fang Yu

Language : Assembly

1. INTRO

This program requires to make comprehensive use of what we have learned in the assembly course. It should implement basic instructional management functions for the managers.The user interface should be simple and clear. There are 10 students, each of whom has 3 subjects. The program allows managers to operate students' data including insertion, modification, deleting, ranking and query.

2. DESIGN THOUGHT

a) If there is no data file in the specified path, the program will create a default data file in the directory. Otherwise, data will be read from the file.

b) Students' data are saved as Characters. Therefore, one could operate the data by modifying the file.

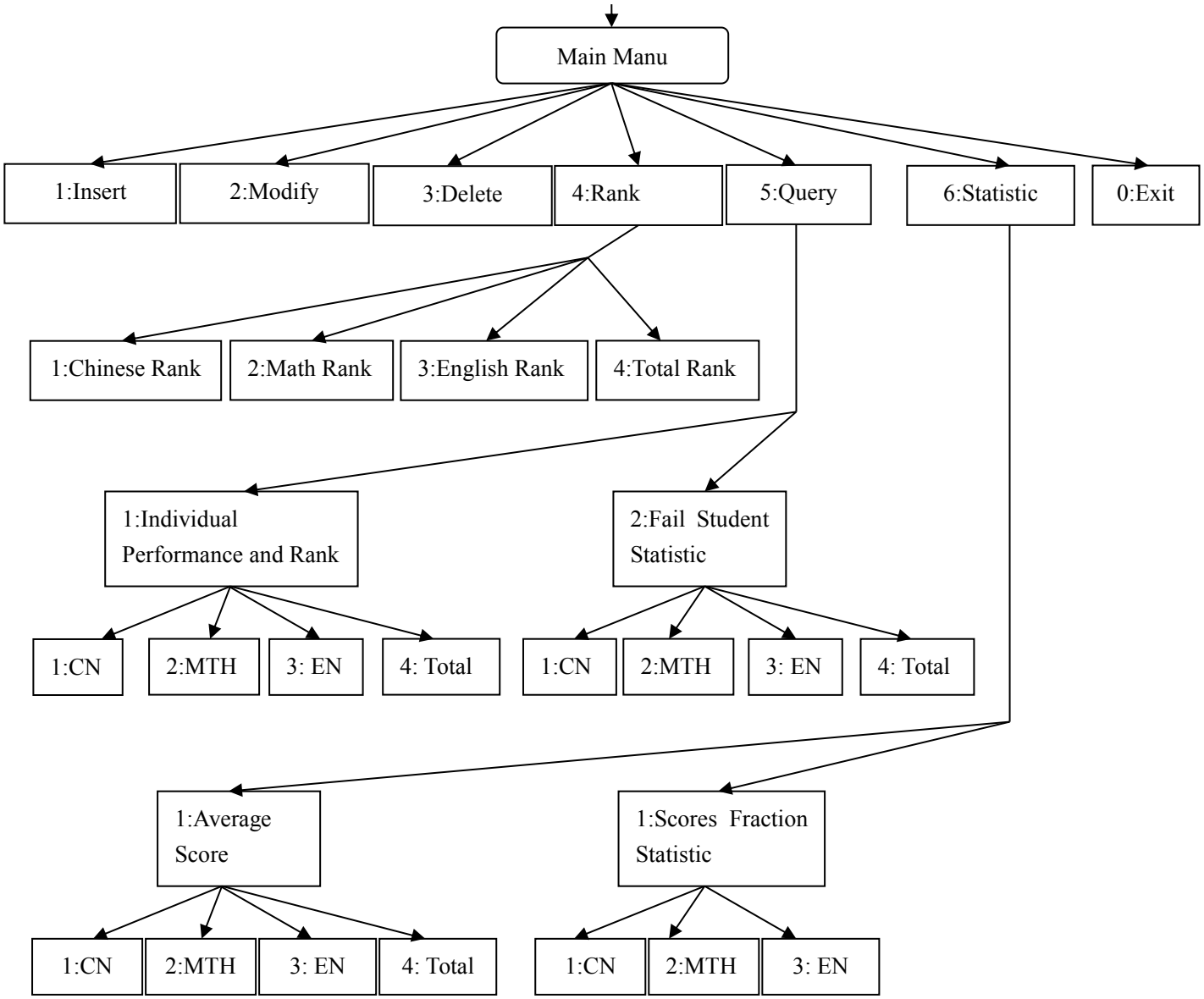
c) After inputting data from the file as Character, the program will transform them into Binary, which is more convenient for

calculating.

d) The program starts calculating data (rank, total score, average score, etc) when it begins to run, then save them into a temporary student data structure. Displays results on the screen. Data will be recalculated every time after an operation finished before returning to the main menu.

e) Only when the user select option ‘0:exit’ in the main menu will the program outputs updated data into the data file.

3. Structure



4. Results

When the program is run for the first time, an error will occur, caused by date file missing.

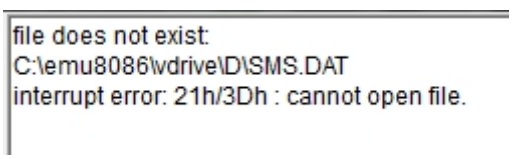


FIGURE 1

Pressed continue. The program will create a data file named

‘SMS.DAT’ in the path ‘local emu8086 director/vdrive/D’.

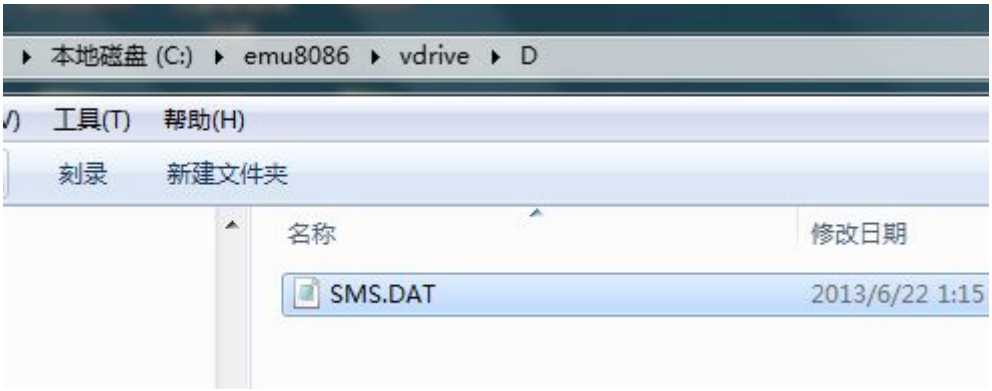


FIGURE 2

Inputs data from the file and initializes them.

FIGURE 3

After inputting and calculating data, main menu is displayed.

ID - Student ID, CN - Chinese Score, CR - Chinese Rank, MTH - Math Score, MR - Math Rank, EN - English Score, ER - English Rank, TTL - Total Score, TR - Total Rank, AVR - Average Score

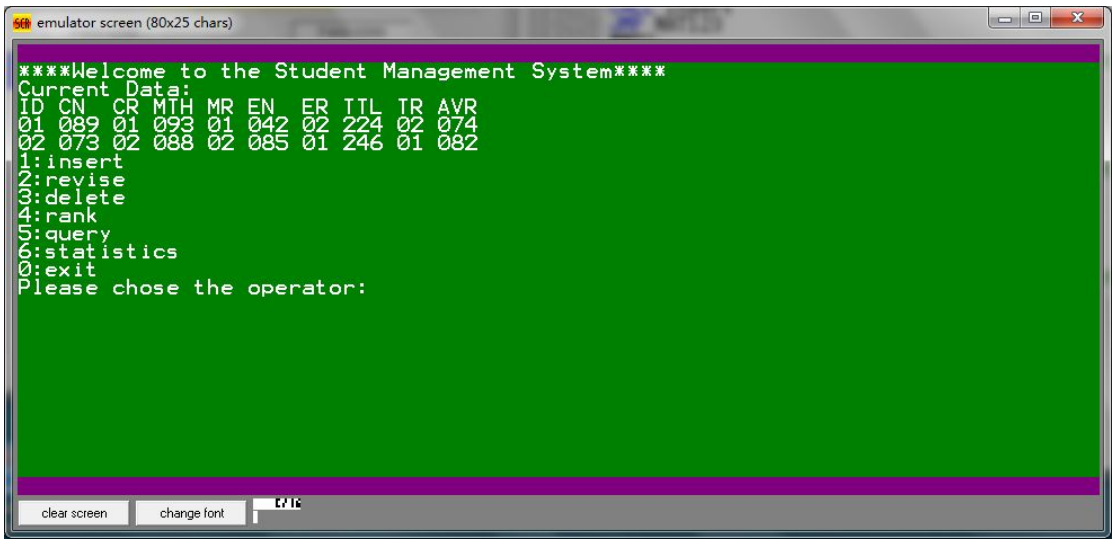


FIGURE 4

Selects ‘1:insert’ to insert a student’s data.

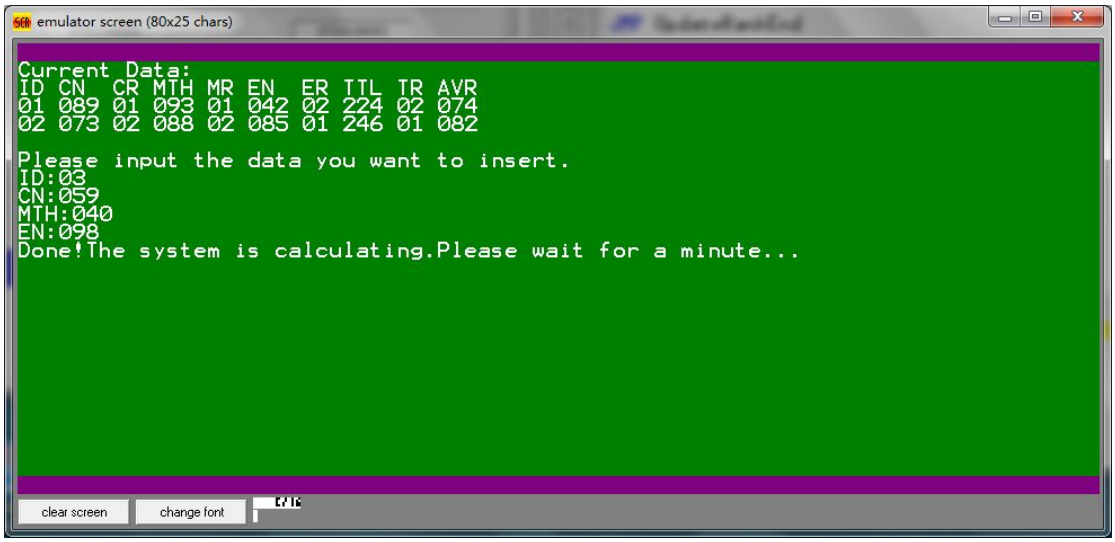


FIGURE 5

Insert successfully.

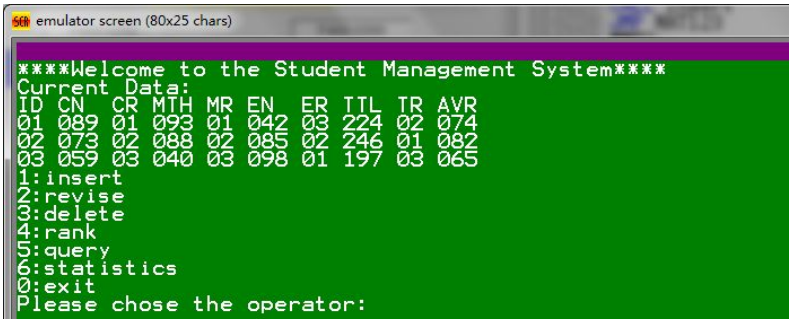


FIGURE 6

Selects ‘2:revise’ to modify a student’s data. If an ID number that does not exist in the system was inputted, it would cause an

error message.

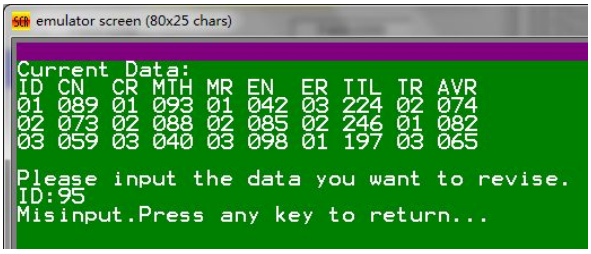


FIGURE 7

Select ‘2:revise’ to modify a student’s data.

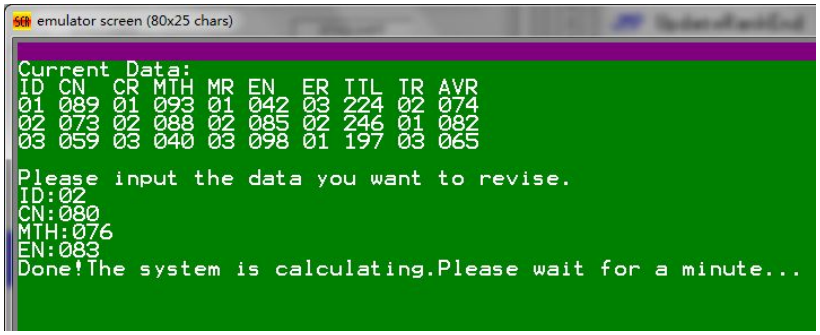


FIGURE 8

Modifies successfully.

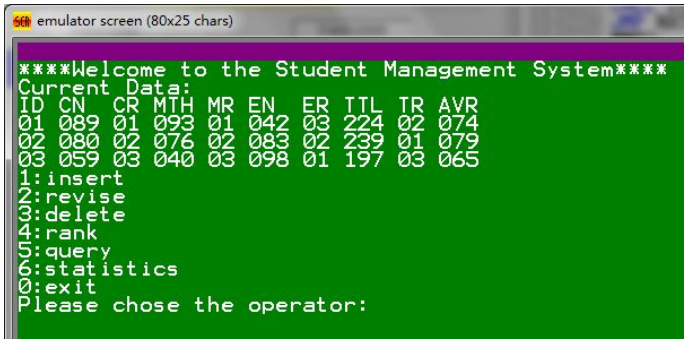


FIGURE 9

Select ‘3:delete’ to delete a student’s data.

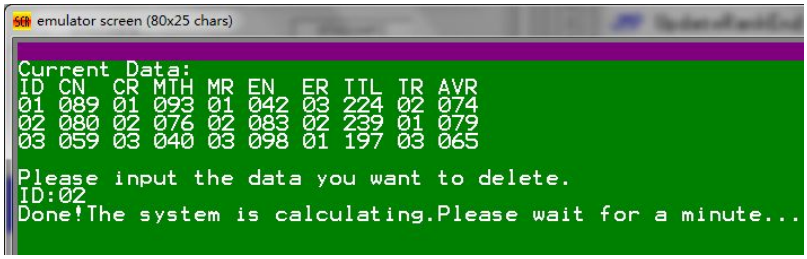


FIGURE 10

Deletes successfully.

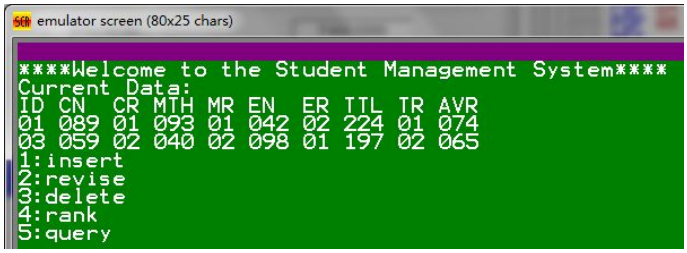
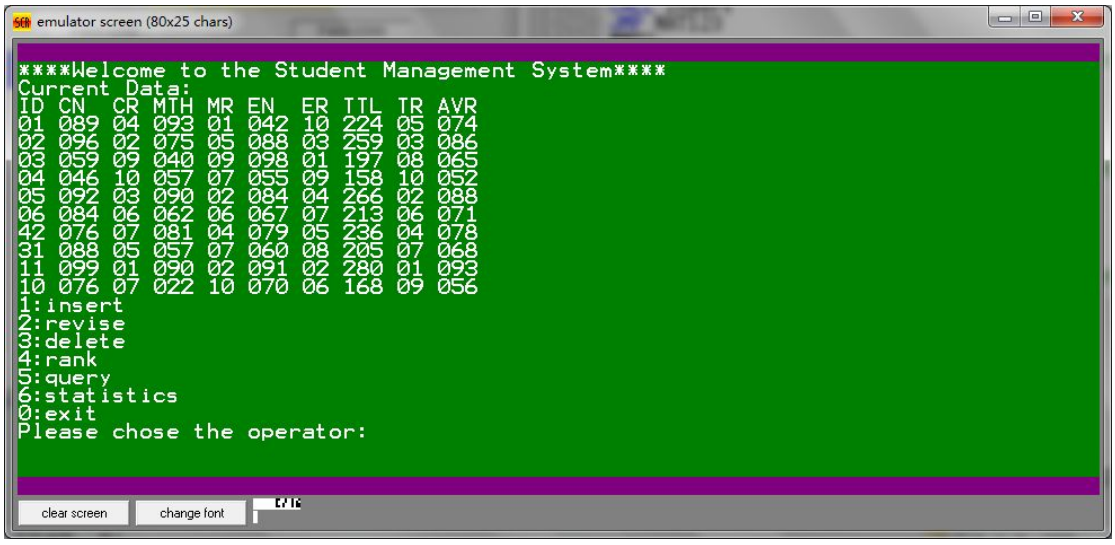


FIGURE 11

Inserts students' data until there are 10 students.



Select '4:rank' to rank students (Chinese, Math, English and total score).

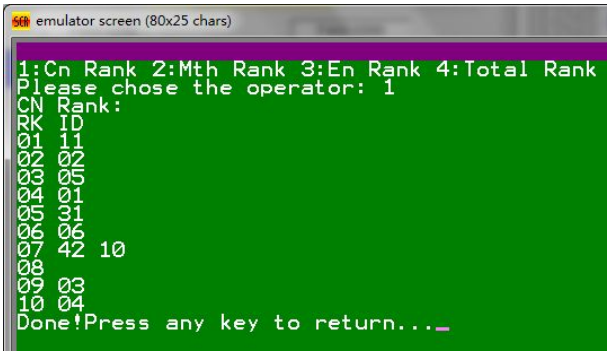


FIGURE 12

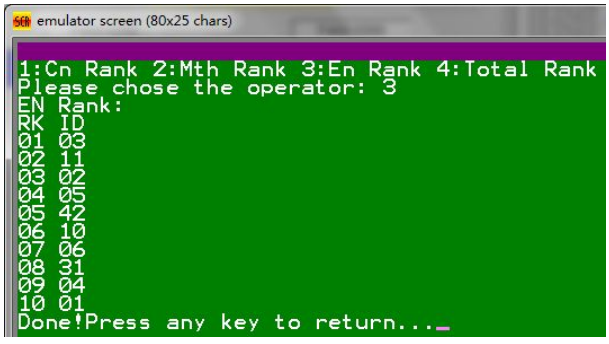


FIGURE 13

Select ‘5:query; to query data.

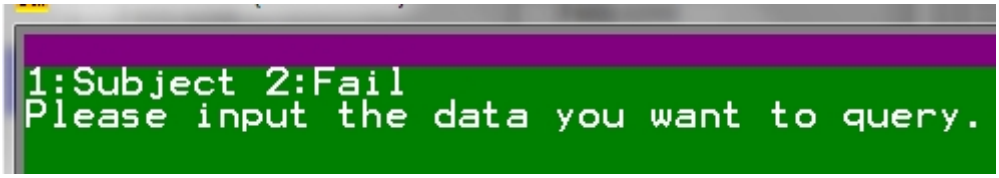


FIGURE 14

Select ‘1:Subject’ to query a subject score and rank or total score and rank of a student.

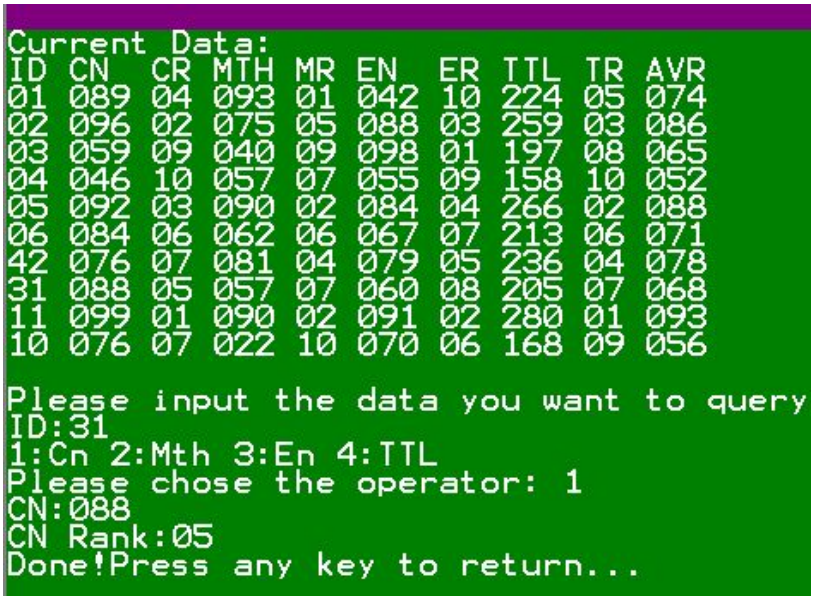


FIGURE 15

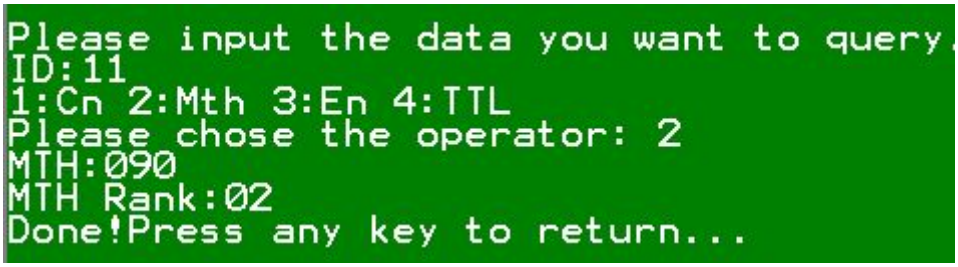


FIGURE 16

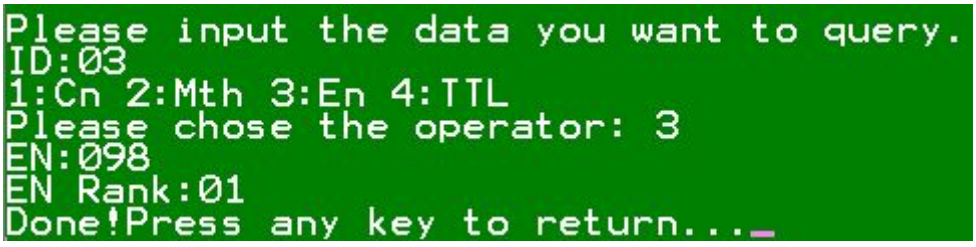


FIGURE 17

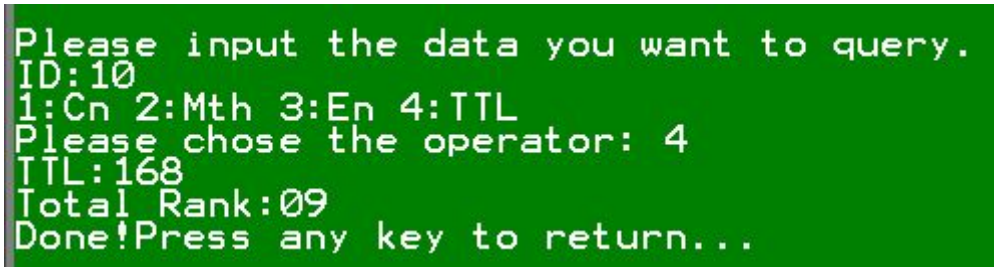


FIGURE 18

Select ‘2:Fail’ to query fail list of a subject.

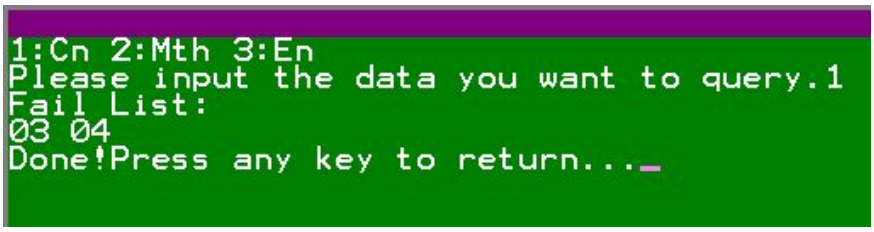


FIGURE 19

FIGURE 20

FIGURE 21

Select ‘6:statistics’.

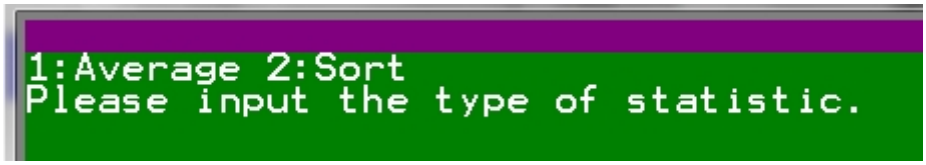


FIGURE 22

Select ‘1:Average’ to query average scores of subjects and total scores.

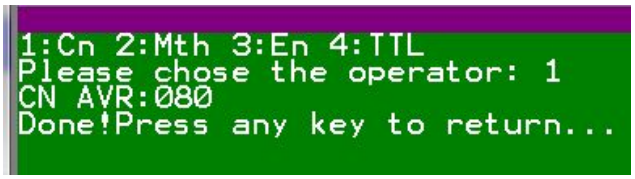


FIGURE 23

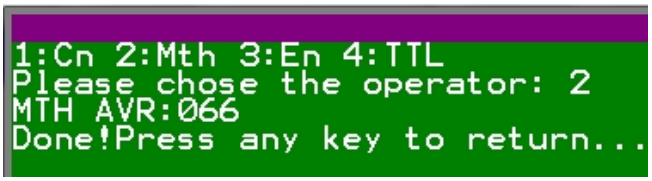


FIGURE 24

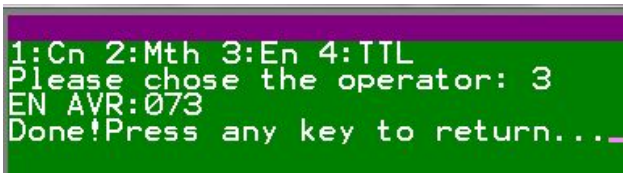


FIGURE 25

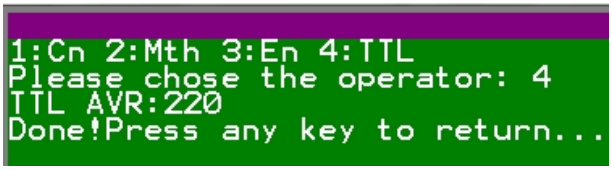


FIGURE 26

Select ‘2:Sort’ to query the number of students of every scores’ fraction.

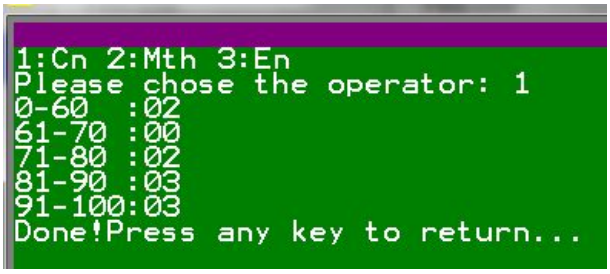


FIGURE 27

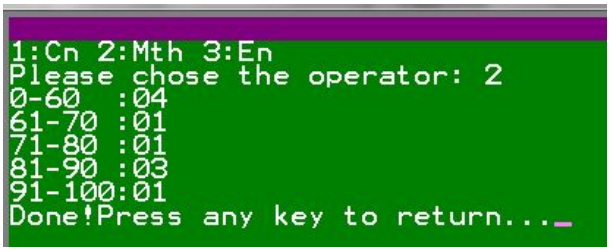


FIGURE 28

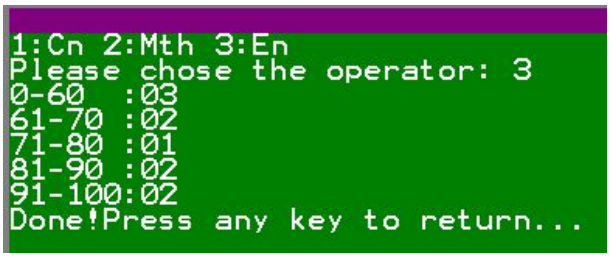


FIGURE 29

Select ‘0:exit’ to update data and exit.

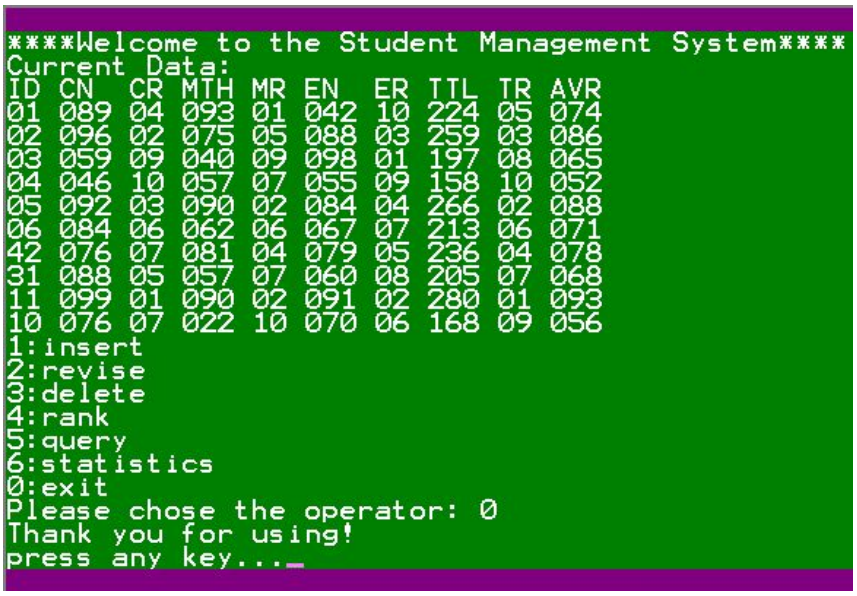


FIGURE 30

Updates file.

FIGURE 31