Course: CISC 105: General Computer Science, sections
Semester: Fall 2006
Professor: Chandra Kambhamettu
Due: Tuesday 10/31/2006, by 1.00PM.

Programming Assignment 2 - Student Score Statistics

1 Academic Dishonesty

Please be sure to re-read the syllabus on Academic Dishonesty. For all programming assignments, students must work INDEPENDENTLY.

2 Grading

80% of the grade depends on your program's correctness and 20% of the grade depends on your program's readability. Your program style should conform with the programs shown in the textbook. Use meaningful variable names and provide indentation to improve readability of your program.

3 Objectives

- Write an algorithm showing the list of steps to take for solving the given problem. In this process, you should identify the functions that are to be used in the program.
- Develop a C program based on the designed algorithm.
- Correct the program for any syntax errors.
- Test the program for any logic errors.

4 Assignment

4.1 Overview

You will be writing a program that reads a table of data from a file that user is prompted for, where each row corresponds to a student's score in a given class. Similar to the present class, the student has scores for i) 13 Labs, ii) 3 Programs, iii) 2 Mid-terms, and iv) 1 Final. The input will first have the student-lastname, the student-status, labs (13), the programs (3), the mid-term scores, and the final score. The student-status shows year of the student: 1 = freshman, 2 = Sophomore, 3 = Junior, 4 = Senior. There are total 80 students in the class, so there are 80 rows and 20 columns in the table.

You need to compute various student statistics, including i) student average of labs, ii) student average of programs, iii) weighted average of student's total score (out of 100), and iv) student's grade in the class. The weights to be used will be put up on the class web page. Menu-interface should be provided so that one can select a particular student-lastname, and then ask for any of the above four choices. Student's grade should be calculated using the table, to be posted in the class homepage.

You also need, `statistics` which would let you calculate additional statistics. This additional statistics will produce the following information for students belonging to all years, freshman year, sophomore year, junior year and senior year (categorized individually): i) high, low and mean of the weighted-total-score, ii) high, low and mean of lab-average, and iii) high, low and mean of program-average, iv) histogram of labs, programs, exams.

4.2 Program Design

The program design should make extensive use of functions, arrays and pointers. Here is an outline of what `main()` should do.

1. Invoke a function `show_instructions` which displays a welcome message and general instructions to the user.
2. Prompt the user for the input: a, b, c, d, s, q (a: student lab average, b: student program average, c: weighted total score, d: student grade, s: statistics, q: quit). Invoke the functions appropriately.
3. Use 1D/2D arrays in order to store the entire data.
4. The input will be in a file, containing the table of scores of 80 students; sample file will be provided.
5. DONOT use `return` statements to return values from your functions. Use pointers to output values from functions.

4.3 Sample Run

```
---------------------------------------------------------------------------
Welcome to the Student Score Statistics!! This program allows you to find
selected statistics of a student's score.
---------------------------------------------------------------------------

Type any of the following alphabet now.

a for student-lab-average
b for student-prog-average
c for weighted-total-score
d for overall-student-grade
s for statistics
q for quit

Type-in now: a

Input Student-name: XYZ

The student's (name : XYZ) lab average is, 86.3.
```

```
a for student-lab-average
b for student-prog-average
c for weighted-total-score
d for overall-student-grade
s for statistics
q for quit

Type-in now: d

Input Student-name: XYZ

The student's (name : XYZ) overall grade is, B+.

a for student-lab-average
b for student-prog-average
c for weighted-total-score
d for overall-student-grade
s for statistics
q for quit

Type-in now: q

           Thank you for using SSS (Student Score Statistics).
```

5 What to Hand In

Run your program for each menu operation and make a script file for grading. The data can be obtained from `student.data` file, provided to you. For those doing extra credit, run the program on each extra option created for the additional statistics.

6 Extra Credit (10 points)

Print out the list of sorted student scores obtained in any of the following chosen categories: i) lab-average, ii) program-average, iii) mid-term, iv) final and v) weighted-total-score. User should be prompted for choosing a category.