Assignment #5

Introduction to C Programming – COP 3223

Objectives

1. To learn how to use arrays to store and retrieve data to help solving problems.
2. To review reading information from a file.

Introduction: Exercise UCF

UCF is committed to student health and already has a great Recreation and Wellness center on campus for students to use. To encourage students to find out more about themselves and their fitness, they are implementing some new initiatives.

Exercise UCF is one of their new programs to educate students about weight, calories, and Body Mass Index. UCF would like to the help of the Computer Science division to create some of these programs.

Problem: The Best of the Best (best.c)

The Exercise UCF program has been a hit so far. But after weeks of monitoring their workouts and calorie intake, some participants are starting to get bored. To combat this, Exercise UCF has proposed a competition. The student with the best workout records for the final four weeks of the program will receive a trophy and an awesome prize!

Prizes will also be awarded for students who participated the longest and showed the most improvement. However, Exercise UCF wants you to focus on the program that will determine the best of the best for the final four weeks.

The field has already been narrowed to the top ten participants. These top ten will record and submit their workout information for both their Weight Workouts and Cardio Workouts for the final four weeks. This workout information is preprocessed to contain the best data from each participant. That is, each participant will be submitting, for each workout, the best time to run a mile for a cardio workout and the highest number of counting reps for a weight workout. Your program should store this information in a set of arrays in order to determine the following:

For each participant (they should be called Participant 1 – Participant 10 to preserve secrecy until the winner is announced) you should output details about their workouts. For their Cardio Workouts, you should print their best time for running a mile, their worst time, and their average time. For their Weight Workouts, you should print their most sets, least sets, and average sets per workout.

Finally, you should announce the winner of the competition. This will be the participant that has both the lowest average time for running a mile and the highest average number of sets per workout. You are guaranteed that only one participant will fit these criteria.

Input Specification

1. The input will be in a file called input.txt

Input File Format

The input file will contain 10 sets of 2 lines each (a total of 20 lines).

The first line of a set will contain 12 decimal values, representing the best times for running a mile from each of three workouts for four weeks.

The second line a set will contain 12 integer values, representing the best number of sets from each of three workouts for four weeks.

Output Specification

Output to the screen.

For each participant, you should print information about both types of workouts. For their Cardio Workout: print their best time for running a mile, their worst time, and their average time. For their Weight Workouts: print their most sets, least sets, and average sets per workout.

Then print the winner of the competition.

This information should be printed according to the specification below.

Participant #1
Cardio Workouts:
    Best Time:
    Worst Time:
    Average Time:
Weight Workouts:
    Most Sets:
    Least Sets:
    Average Sets:

Participant #2
...

The winner is Participant #X! Congratulations!

Output Sample

Below is a sample output of running the program. **Note that this sample is NOT a comprehensive test.** You should test your program with different data than is shown here based on the specifications given above. In the sample run below, for clarity and ease of reading, the user input is given in *italics* while the program output is in **bold**. (Note: When you actually run your program no bold or italics should appear at all. These are simply used in this description for clarity's sake.)

Sample Input File (input.txt)

```
9   10  11  9.5  10.2  9.8  10  8.7  9.3   9.1  9   8.9
4   5   5   4    6     5    4   3    4     5    5   4
8   11  10  8.5  10.7  9    11  8.5  9.4   9    10  9.9
5   5   5   5    5     5    5   5    5     5    5   5
9   9   9   9    9     9    9   9    9     9    9   9
4   5   5   4    6     5    4   3    4     5    5   4
8   9   10  8.5  9.2   8.8  9   7.7  8.3   8.1  8   7.9
7   4   6   6    6     5    6   6    6     6    5   6
10  11  12  10   11.2  9    11  9.7  9     9.1  10  9.9
6   5   6   6    6     5    6   6    6     6    5   6
10  9   11.2  9.5  10.8  9   10.3  8   9.3   9.1  9   8.9
5   4   5     4    6     5   4     3   4     5    5   4
9   10  10.4  8.5  10.2  9.2  11.1 8   9.4   9    10  9.9
6   4   5     5    5     5    5    5   5     5    5   5
10  8   9.6   9    9.6   9.7  9    9   9.7   8    10  9
5   4   5     4    6     5    4    3   4     5    5   4
9   8   10.8  8.5  9.5   8    9.5  7   8.3   8.1  8.8 9
5   4   5     4    6     5    4    3   4     5    5   4
11  10  11    10   11.3  9.4  11   9   9.5   8    10  7
4   5   5   4    6     5    4   3    4     5    5   4
```

Sample Output

Participant #1
Cardio Workouts:
- **Best Time: 8.70**
- **Worst Time: 11.00**
- **Average Time: 9.54**

Weight Workouts:
- **Most Sets: 6**
- **Least Sets: 3**
- **Average Sets: 4.50**

Participant #2
Cardio Workouts:
- **Best Time: 8.00**
- **Worst Time: 11.00**
- **Average Time: 9.58**

Weight Workouts:
- **Most Sets: 5**
- **Least Sets: 5**
- **Average Sets: 5.00**

Participant #3
Cardio Workouts:
- **Best Time: 9.00**
- **Worst Time: 9.00**
- **Average Time: 9.00**

Weight Workouts:
- **Most Sets: 6**
- **Least Sets: 3**
- **Average Sets: 4.50**

Participant #4
Cardio Workouts:
- **Best Time: 7.70**
- **Worst Time: 10.00**
- **Average Time: 8.54**

Weight Workouts:
- **Most Sets: 7**
- **Least Sets: 4**
- **Average Sets: 5.75**

Participant #5
Cardio Workouts:
- **Best Time: 9.00**
- **Worst Time: 12.00**
- **Average Time: 10.16**

Weight Workouts:
- **Most Sets: 6**
- **Least Sets: 5**
- **Average Sets: 5.75**

Participant #6
Cardio Workouts:
- **Best Time: 8.00**
- **Worst Time: 11.20**
- **Average Time: 9.51**

Weight Workouts:
- **Most Sets: 6**
- **Least Sets: 3**
- **Average Sets: 4.50**

Participant #7
Cardio Workouts:
- **Best Time: 8.00**
- **Worst Time: 11.10**
- **Average Time: 9.56**

Weight Workouts:
- **Most Sets: 6**
- **Least Sets: 4**
- **Average Sets: 5.00**

Participant #8
Cardio Workouts:
- **Best Time: 8.00**

Worst Time: 10.00
Average Time: 9.22

Weight Workouts:
Most Sets: 6
Least Sets: 3
Average Sets: 4.50

Participant #9
Cardio Workouts:
Best Time: 7.00
Worst Time: 10.80
Average Time: 8.71

Weight Workouts:
Most Sets: 6
Least Sets: 3
Average Sets: 4.50

Participant #10
Cardio Workouts:
Best Time: 7.00
Worst Time: 11.30
Average Time: 9.77

Weight Workouts:
Most Sets: 6
Least Sets: 3
Average Sets: 4.50

The winner is Participant #4! Congratulations!

Deliverables

One source file: *best.c* for your solution to the given problem submitted over WebCourses.

Restrictions

Although you may use other compilers, your program must compile and run using Code::Blocks. Your program should include a header comment with the following information: your name, course number, section number, assignment title, and date. Also, make sure you include comments throughout your code describing the major steps in solving the problem.

Grading Details

Your programs will be graded upon the following criteria:

1) Your correctness

2) Your programming style and use of white space. Even if you have a plan and your program works perfectly, if your programming style is poor or your use of white space is poor, you could get 10% or 15% deducted from your grade.

3) Compatibility – You must submit C source files that can be compiled and executed in a standard C Development Environment. If your program does not compile, you will get a sizable deduction from your grade.