Homework 8: Class Grades Simulator++

Assigned: Monday 22 April
Due Date: Monday 29 April

Objectives

More practice writing a program that uses a `switch` statement, the `+=` assignment operator, and a `for` loop.

Assignment

This assignment is a continuation of Classwork 8. (NOTE: You must still submit Classwork 8 and Homework 8 separately.) For this homework assignment, embellish your program from Classwork 8 with the following features:

1. Do a second randomization inside each case statement to determine an actual numeric value for the grade, and for any non-F grade, determine whether the grade is a +, regular, or -. This will require a nested if-else if-else block. For example, 90-93 is A-, 94-97 is A, and 98-100 is A+.
2. After the `for` loop terminates, print out the *highest* and *lowest* grades generated by the program.
3. After the `for` loop terminates, print out the *average* grade generated by the program. Use a comment when you're doing this calculation to explain why you can calculate the highest and lowest grades inside of the loop, but the average grade is best computed outside of it.

Example Compilation and Execution

```
[arsenaul@linux1 hw8]$ gcc -Wall grades2.c
[arsenaul@linux1 hw8]$ ./a.out
Out of 36 students, here is the class grade breakdown:
A+: 4
A: 0
A-: 0
B+: 3
B: 3
B-: 3
C+: 3
C: 1
C-: 3
D+: 3
D: 3
D-: 2
F: 9
Highest grade: 100
Lowest grade: 2
Average grade: 66.53
```

```
[arsenaul@linux1 hw8]$
```

Notes

- Make a *copy* of your program from Classwork 8.

  ```
  [arsenaul@linux1 ~]$ cp ~/cs104/cw8/grades.c ~/cs104/hw8/grades2.c
  ```

- Think about how many additional `num` variables you need to create to account for all the +s and -s.
- Think about how to calculate the sum of the numbers generated by the program. Use a `float` variable so you get a floating-point value when you compute the average.
- Since the numbers generated by the program must be positive, you can initialize the maximum value to 0. Explain why this works in the variable declaration comment.
- Make sure that you indent each block of code so that you don't get lost and your code compiles.
- Make sure you review the notes for Classwork 8, as they still apply here as well.
- Use the following commands to generate the random numerical grades:
  - If A: `grade = (random() % 11) + 90;`
  - If B: `grade = (random() % 10) + 80;`
  - If C: `grade = (random() %10) + 70;`
  - If D: `grade = (random() % 10) + 60;`
  - If F: `grade = random() % 60;`

Extra Credit

Try to do the following embellishments (notice they are the same as Classwork 8):

Option 1: After the printout of results, add an if-else block that tells the user if First-Year Intervention (FYI) notifications need to be sent, and if so, how many. To determine if any need to be sent, count how many Ds and Fs there are.

Option 2: Ask the user for how many grades to simulate. Warn the user if they enter something less than 0. If the user enters 0, print the following to the screen: "Sorry to see you don't want to use my simulator" with a newline at the end for proper display on screen.

Option 3: Change the provided for loop to a while loop.

Grading Rubric

- Typescript: 15 points
- Header comments: 2 points

- Body comments: 3 points
- Compiles: 40 points
- Gets Max: 15 points
- Gets Min: 15 points
- Gets Avg: 10 points
- EC1: +5 points
- EC2: +5 points
- EC3: +5 points

What to Submit

Use the `script` command to record yourself compiling and running your program 3 times. (Do not record yourself editing your program!) Exit from script. Submit your program and the typescript file.

```
[arsenaul@linux1 hw8]$ submit cmsc104_arsenaul hw8 grades2.c typescript
```

Verify Submission

If you *think* you submitted the assignment, but the `submitls` command doesn't show you your file names, then the files were **not** submitted and no grade will be given.

```
[arsenaul@linux1 hw8]$ submitls cmsc104_arsenaul hw8
```

Last modified: 08 February 2023