CSCI 1310 Introduction to Programming [Fall 2015]
Instructor: Boese
Project #2

Design your own C++ project!

In this project, you will be designing your own assignment! You can work with data files from your area of study, or something you want to do like keep track of your music, or design a game.

The only requirement for the assignment is that you include the following programming features in a Python program:

At least:

- Driver class (that makes use of your 2+ classes)
- 2 classes (set up correctly with header files for the class declarations)
- 2 instance variables in each class
- 2 functions in each class (functions that you write)
- 1 loop (either while or for or do…while)
- 1 if/else if/else blocks (can be if/else, if/else if, if/else if/else, …)
- 2 numerical variables
- 2 strings

In addition to those features, your assignment also needs to include **one** of the following:

- File I/O and whatever data processing is needed (this is both file input and file output)
- Making use of a 3rd party library (not the libraries standard to C++)
- Create graphics with C++

What kind of programs can you write? Here are some example ideas:

- Dungeon Game modified: Modify the Dungeon Game assignment to include classes and file I/O. *You must re-work the map, cannot have the same rooms and paths as the homework.*

How to tackle this project

Clearly, this project is a bit bigger than the weekly assignments we have done so far. However, you have the skill set to complete it. Here are my recommendations for how to be successful in this project.

- Come up with a plan
    - Spend some time thinking about your project before you begin coding. Imagine the interaction between the computer and user, and articulate that either on paper, to a friend, or to one of the TAs, LAs, or the Instructor. Explaining your idea to someone else will help you understand it better.

    - Once you have a good idea of the game functionality, translate that functionality into concepts. Generate an outline or a flowchart of those concepts.
    - Translate the concepts into coding plans. For example, when you explain the game, if you come to a point where you say something such as, "then the user has to make a decision", you know that is a place for an if statement.
- Write code
    - Start small and implement a basic set of features first. Get each piece working before moving on to the next piece. For example, if your game has a while loop that checks for user input, get that working first before moving on and writing the code to handle the input.
- Test, test, test to make sure your code is working
- Be awesome. You just wrote your first computer game, enough said.

What to submit

- For this project, you will submit:
- Your code. Include all .cpp and .h (or .hpp) files, zip them together into one file.
- A 1-2 page description of what your game or program does and how to run it
- Screenshots showing the game working
- You will also need to sign up for an interview grading session.

Incentives

A project just wouldn't be a project without a contest. To incentivize we will be having a contest for who submits the best project.

FAQs:

1. Do I have to design my own project, or can I work with a friend to design a project.
    a. You can work with a friend and pair program. Be sure to both sign up for separate individual grading meetings. You will need to understand all the code submitted.
2. How will projects be graded?
    a. All projects will be interview graded to give you the opportunity to brag about your work
3. How will projects be judged in the contest?
    a. The judging criteria will be code correctness and style, creativity of idea, and clarity of the written explanation.
4. Who will be judging the contest?
    a. The TAs will nominate projects, and then the TAs and I will discuss all of the nominated projects and select the winners.
5. What do I get if I win?
    a. 2 points added to your final grade. For example, if I calculate your grade and it is an 89, and then I see that you won the programming contest, then your final grade will be a 91.
6. What if I don't want to design a game; is there another project I can do instead?
    a. Yes. See some example default projects you can do.

7. If I do the absolute minimum, can I still get a good grade?
    a. Yes. Any project that fulfills the requirements listed above, works, and is explained well, will receive an A.
8. What percentage of my course grade is this project?
    a. 5%