

CSCI 1300 Introduction to Programming
Instructor: Hoenigman
Project 1
Due: Wednesday, October 21 by 8am.

Design your own game

In this project, you will be designing your own interactive computer game. Yep, designing your own game. The idea for the game and the code to implement your idea will come from you. The only requirement for the assignment is that your game include the following programming features:

At least:

- 4 functions
- 2 loops (either while or for, or one of each)
- 4 if/elif/else blocks
- 4 numerical variables
- 4 strings

In addition to those features, the game also needs to include one of the following:

- File I/O
- Sound or graphics (we'll discuss more in class)
- An API call using either the Requests module or a third-party Python interface, such as Geopy

What is a game?

The definition of computer game for this assignment is anything that interacts with the user and continues going until the user says Quit. Clearly, this is a more relaxed definition than might generally be used. Your game can be command-line based, such as the "I'm thinking of a number between 1 and 100" game that we did in lecture. Or, something more sophisticated that includes sound, graphics, and external sources of data.

Game subject areas

The subject area of your game is also your choice. If you want to write an educational game that quizzes your knowledge of chemistry, sounds great. A choose your own adventure game? This is also a fabulous idea.

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 skills necessary 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, CAs, or

the Instructors. 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.
- A 1-2 page description of what your game does and how to play it.
- Screenshots showing the game working.
- Zip all files together and submit them to moodle. Name the zip file <Lastname>_Project1.zip.

Incentives

A game project just wouldn't be a game project without a game contest. To incentivize against going out to the Internet and downloading a pre-built Sudoku game, we will be having a contest for who submits the best game. (We'll also be interview grading everyone, so you better be able to explain that Sudoku game.) In the interest of fairness, there will be different categories for submitting your game, since some of you came into the class with a little bit of programming experience and most of you did not. In your interview grading session for the project, we will ask you about your prior programming experience. No sandbagging please.

FAQs:

1. Do I have to design my own game, or can I work with a friend to design a game.
 - a. You can work with a friend to come up with a game concept and you can help each other work through coding issues, but each of you needs to submit individual work.
2. How will projects be graded?
 - a. All projects will be interview graded to give you the opportunity to brag about your work and also discuss parts of your project that might not be working quite the way you want them to.

3. How will games 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. 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.
7. What percentage of my course grade is this project?
 - a. 5%