Project 1 Reflection
Jacob Willard

Overview

For my project, I created a text-based puzzle adventure game called Anything But The Machine. The adventure itself is sort of similar to Alice in Wonderland in that you go from surrealist setting to surrealist setting all the while thinking that there might be some metaphor you are missing, when in reality it is entirely possible that there just isn't. The text-based mechanics allow you to pick up and drop objects, use objects in the game, talk to non-playable characters, and also obtain detailed or not so detailed information about your environment. The game has 3 chapters, each with at least one puzzle that you have to solve in order to progress.

Reflection

I learned a huge amount about programming while working on this project. The process of designing the game mechanics were a great way to get used to the Object Oriented style of programming. Thinking of objects in python as real objects in a virtual world is a very intuitive way of getting warmed up to that way of thinking.

It happened that the most instructive period of time working on this project was the last few days before it was due. Two days before the due date I was nearly finished with what I thought was a pretty successful build, but then I decided to test it outside of Jupyter notebook for the first time. I learned the hard way that Jupyter apparently protects the user from certain mistakes in programming, and all of a sudden there was a whole list of problems I had to deal with and the due date was right around the corner. This was when I learned a really important lesson: just because it runs doesn't mean it's good enough, and if you can make it more simple and intuitive you absolutely should. All the time I spent working on the project I had gotten used to sticking with my first approach for no reason other than that it worked well enough. Rather than go back and see if there is a better or less risky way of doing it, I just moved on to the next problem without considering the kind of issues that might create later on. By the time I had nearly finished scripting all of the levels, I suddenly started seeing problem after problem show up. Most of these issues had to do with abusing global variables and class attributes. Trying to debug my code felt like trying to cut off all of Hydra's heads. Thankfully, I was able to overhaul my core mechanics and reconstruct the levels in time. In the process, I learned more about programming than I did during the entire rest of the time working on the project.

Testing

To test out my game, I created a .bat file called "play game" that automatically runs that game in the command line. However, I recognize that the syntax for running python from the command line is a little different on windows than it is on other systems, so if the .bat file doesn't work, just run "game_file.py" from terminal. Be sure to select "Info" from the game's menu before starting a new game. Also, since I know nobody wants to spend a ton of time figuring out the puzzles in order to grade my project, and since it is my first time designing a game so the puzzles might not be balanced very well, I included a walkthrough of the game in my repository.