

Dylan Fosse-Previs

Programing

18/10/23

### Programing midterm

For this year's Programming midterm project we had to create a hangman program. The program needed to pick a random word and then turn that word into a hangman game. We earned extra points if we drew a picture on screen of the dude.

For starters the first and easiest thing I did was write the framework for the program. The dude was quite easy, for his code I wrote a function to take his health out of 6 total points and display a figure representing that health point. Next I wrote code to pick a random word out of a list luckily I had some help I used the url to lib api and the random api to take a random word from mit.edu's list of 1000 words after a sanitized them of course for words shorter than 3 letters because then its no fun i also had to clean them of extra characters likely added because of the api. Once that was done I was ready to start the actual project.

To start the actual code a wrote an intro that let the GM decide whether to use their own word or our randomly provided word then I wiped the screen so the player could not see the word. For the actual game I needed 4 basic values, the answer, the guess, the life, and the used letters. Once I defined them I created a script in order to take the guessed letters, check if they were in the final word, add them if they were and subtract from the player's life if not. I did this all in a while loop so whenever the player's life dropped below 0 the loop will close and the

game will determine whether you won or lost and print the appropriate reaction from the computer.

Despite not needing to and likely making the code much harder, I enjoyed accessing the internet via python in order to find the random word. Testing it was very hard, especially the process of cleaning the word for use in the game. I received a lot of errors and had to use a lot of debug print statements in order to figure out where or how the program got stuck. During testing I found that my code was actually taking some time to load but I can't tell if this is an accomplishment or a bug I need to fix. It was likely the result of python looking through so many words on the internet so I don't think I need to or even can fix it with my limited knowledge.

Converting strings to list and list to string was something I found myself doing a surprising amount. Not only did I need to do so in an attempt to clean my random word generator but I also did so whenever I needed to check if a letter was in the final word. Python never does make it easy but from my time testing I found many Angles to go off of whenever I needed to.

This project will never be my best work, I don't think anything will. But, I do think this project was some of the most fun I have had coding a project. It was just the right amount of me thinking to myself how easy it was going to be and then realizing how much work I was going to have to do that I know all programmers feel at some point. I am curious what this class has in store for me next but I do know that if I approach it with the same approach I did this time I will likely have a good time.

