

Homework 1 Descriptions

1. I first print out all of the rules of the game and then store a variable which indicates whether there has been a winner, or the previous game ended in a tie. After this, I decided to make a loop until the winner variable is not True which means this loop will keep running while the games end in a tie. Inside of my loop, I have the logic of picking a choice at random for the computer and then storing the choice of the user. I also defined different cases which will display the correct and possibly end the loop.
2. Since there is no repetition in this problem, I decided to use simple IF statements to take care of all conditions. First check if there is a file path given to us in the command line and take care of that and then keep checking the rest of the conditions using IF statements.
3. For this problem, I first take care of the command line arguments and check if at least three files are given to the program. After this, I make the backup directory using the current date and then also create a log file inside that directory. Lastly, I use a FOR loop to go through all of the arguments provided and do necessary checking before making a copy with the new name in the backup directory.
4. This last problem involved a menu running at all times which is perfect for a loop which only ends with a specific command and in our case it would be 'exit'. I first created a while True loop which acts as the menu and keeps looping until the user would say 'exit'. To help the menu, I made 2 functions which handle displaying files and looking into a subdirectory. I keep running the display_file() function until the user says no or there is no more content of the file to be shown and after the loop is over, it would go back to the

original menu and start the process over. The `search_recent_files()` function searches inside a directory for files that have been modified in the last 24 hours and return a list of files back to the menu.