Sania Islam & Ella Reamer

Professor Bankston

CMPS 1100

Due: 27 Sept 2024

Midterm Project Reflection

1) How long (cumulative) have you spent on the code?
    a. I spent probably somewhere between 15-20 hours on this program. I did not use any kind of timer extension to track the time spent on every part, but I will be sure to do so in the future.
2) What was the most time-consuming part?
    a. The most time-consuming part was likely just debugging the program because I was completely unfamiliar with the process and kept getting lost in the code.
3) In retrospect, how could you have worked for efficiently?
    a. In retrospect, I would have relied on ChatGPT and the online community more. Before this class, I have almost exclusively taken liberal arts courses in which the use of AI is generally really frowned upon. I was hesitant to rely on those sources in fear that I wouldn't understand how to code without those resources if need be. However, over the course of this project, I have grown to understand that tools like ChatGPT and online repositories are integral to learn a subject that is constantly evolving. I like that coding is so communal and how we are constantly building upon each other's projects, rather than reinventing the wheel. I have learned to use ChatGPT to explain aspects of code I am unfamiliar with and propose fixes that I wouldn't have thought of immediately rather than replacing my code with AI written ones. I think these tools can help me be more efficient and more creative.
4) What libraries/starter code were most useful? To what extent did you need to modify them?
    a. I mostly used the Geeks for Geeks backtracking algorithm and changed the dimensions for the rows, columns, and subgrids. Using this code for a 16x16 grid, however, greatly increased the difficulty and thereby time spent both to input and solve these puzzles. I added a function to check the grid for misplaced numbers before the program attempted to solve it to avoid it getting stuck, and also created a simpler method for users to input the puzzle, row-by-row with a mechanism to correct the previous line, because I noticed how often I would make mistakes in inputting a 16x16 grid. I also added strings of dashes and vertical bars between subgrids to make the puzzles easier for the user to read in the terminal.