

Colab 6D

None of our group completed a clean compile of the lab. We felt like we understood what needed to be done, but in the end, had trouble executing the proper code.

The first problem was an out of bounds issue with the 2-D array. The change was a tweak in an if statement (rows-1) and (columns-1) to keep from being OOBs. The original recursive base case did not work. The original was just a test to see if the location was a 0 on the last row.

The change was made to directly compare the current location to the desired location stated earlier in the program. Also, there was a situation in which the code provided a path to complete the maze down, but not up. To fix it, two more cases were added to satisfy all the different possible angles besides downwards. Next, we had an issue where the implementation of the 2-D array was not correct, due to the lack of a real initialization for the array. The solution was found to be using a pointer to a pointer such as `int** maze` to declare the array.

The algorithm itself was found to be an issue for this lab. Instead of trying to be too meticulous, with too many small functions, try and keep it simple, instead of being too in depth. More on the array, it needed to be defined in multiple loops, with the first two defining each of the dimensions alone, and then finally by using them both together in the loop with `i` and `j`. In regard to `mazePoint`, I wrongfully used it, whereas Harrison used it correctly as a saving mechanisms for points to be correctly identified.

In the end, we felt like problems should be solved with a simpler mindset before any in depth work is done. As well as that, we felt that we could've started our work earlier since there was an extended time period, so that more collaboration and troubleshooting could've been executed pre-PartD. Breaking the lab up into parts to do different days (time permitting) gives more of a chance to properly learn and conquer the current situation as well as to correctly execute the code.