Lab Project 1 Mastermind

Introduction

My Mastermind game has all the qualities that the game is expected to have, you can change the size of the code and get responses for both red and white pins. Now, the biggest thing lacking is the input of numbers instead of colors, which in terms of playing the game changes very little but the feel of the game is slightly different. Hopefully, the game is up to your specifications and everything is included that you wanted. I feel like this was the smoothest project I have had to make for you thus far, it was a decent amount of fun to make and I tried my best to integrate the ideas we’ve learned in class while making it. The program is about 200 lines long, has 14 variables, 2 pointers, 9 functions, and includes the iostream, iomanip, ctime libraries.

With a decent amount of planning beforehand, this game turned out to be not as big a task I thought it was going to be. The more I do this process the easier it becomes so I guess in that way I appreciate the work you make us do throughout this whole endeavor. One thing that I really improved on this time was segmenting the code into small chunks. I feel each function that I made does one specific thing and it does it well.

Pseudocode

//this version of mastermind is focused on actually playing the game, before was just figuring out how to generate the code sequence

//create a dynamically allocated array

//input size of array from codeLength()

//input size of array from codeLength()

//this is to run either the dup or noDup code generator

//displays code if not guessed correctly

//make sure to delete dynamically allocated memory

//best practice, return pointer to nullptr

//switch case to determine length of array

//as this function says, it is just a boolean that returns true for if the player wants duplicates and false if not

/\*

\* this section should be to take care of the generation of the code, take in the pointer, bool and size of array and then output an array with the code sequence

\* from my understanding allowing duplicates just means the array is randomly filled with numbers 0-7

\* not allowing duplicates means the array from 0-7 is shuffled and then put into the code array

\*/

//pretty simple, just randomize digits from 0-7

//this shuffle function is from my blackjack game from last semester

//input validation

//this just loops through the array and catches any mismatches

//this checks for if the guesses are in the correct spot

//this checks if the guess has any that are correct but not in the right spot