

Project 1

Mastermind

**CSC-5 48309
Kelly Williams
10/23/2022**

Introduction

Title: Lab Project 1: Mastermind

This is the Mastermind puzzle game. The computer generated a sequence of colors of a given length and specified allowance of duplicate colors.

The objective of the game is to guess the exact sequence of colors within ten guesses.

After each guess, you will receive feedback on how many correct colors are in the right place, and how many correct colors are in the wrong place.

You win the game when you guess the exact sequence of colors. Otherwise, if you run out of guesses without guessing the correct sequence, you lose the game.

Summary

Project size: 108 lines

Number of variables: about 10

I tried to create separate functions for creating the computer's sequence, getting the user's guess, and testing each guess, but I ran into too many errors and figured I could just write it all into main without it being too long.

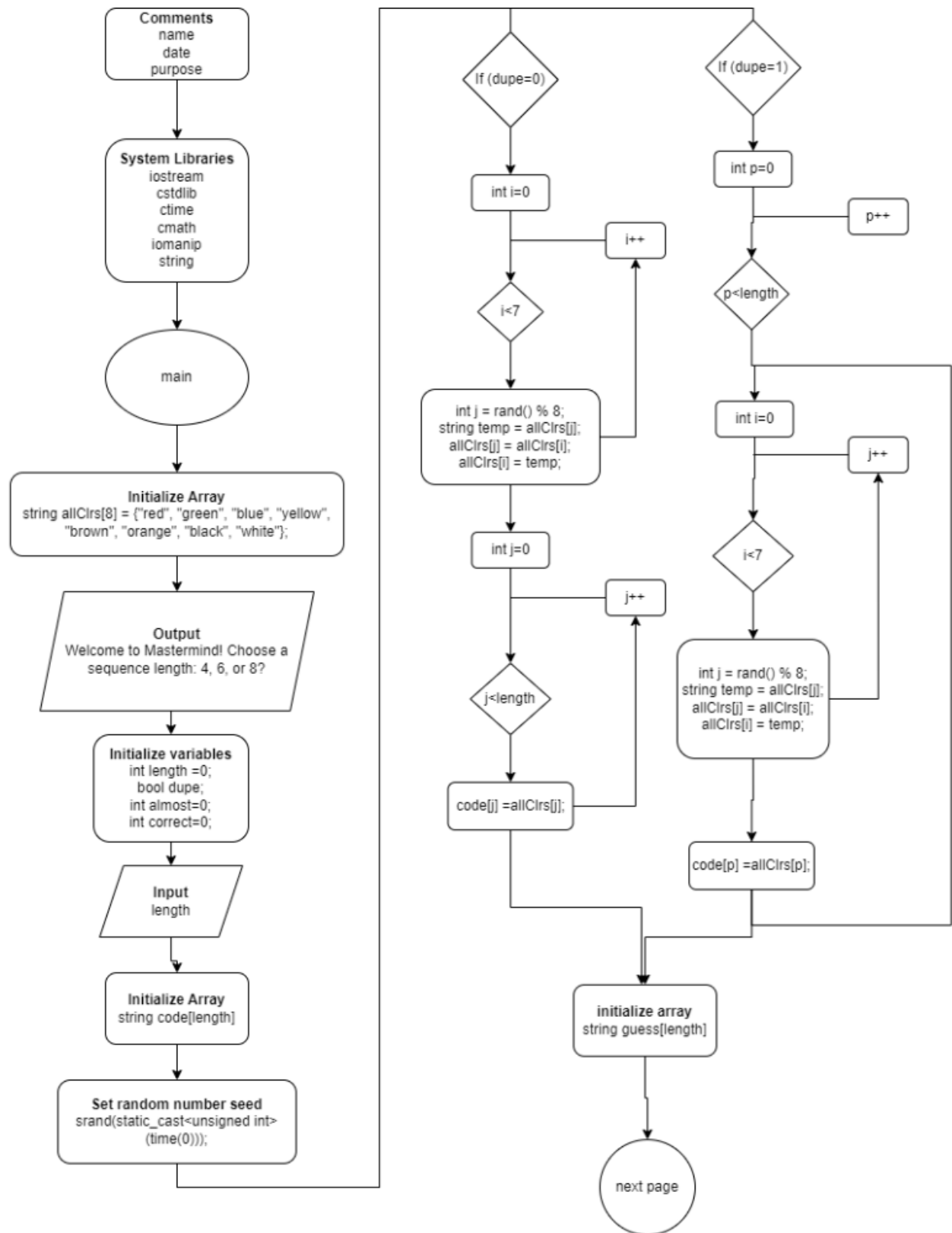
I used arrays to hold the values of each sequence, and a random number generator to scramble the array of all the colors which was then used to fill the array for the computer's sequence.

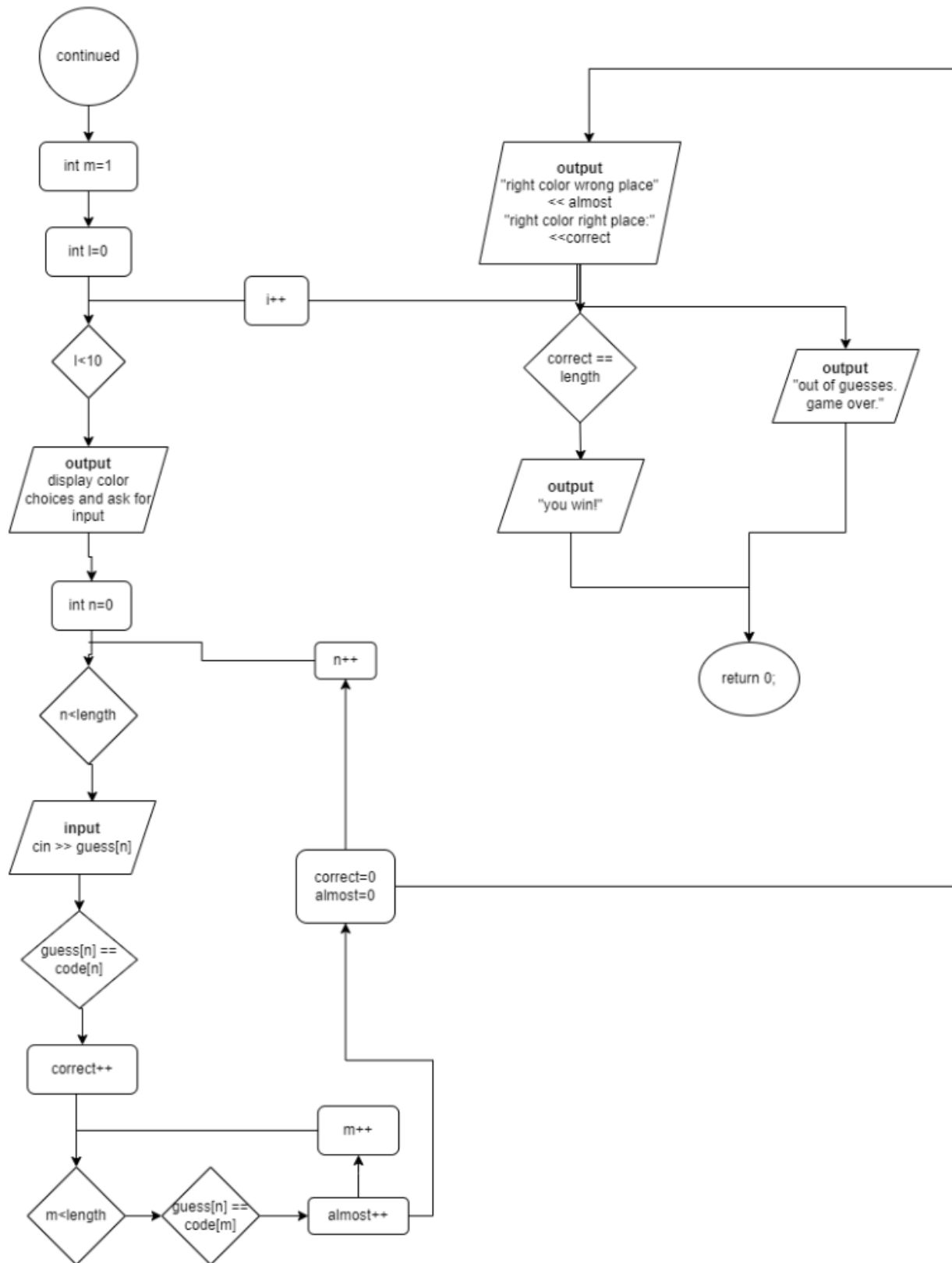
It took about seven hours total.

Description

The main point of this program is to create and execute a text-based version of the game Mastermind.

Flow Chart





Pseudo Code

Create master string of all the colors.

Output game intro.

Initialize variables for length, duplicates, almost, and correct.

Output to ask if duplicates are allowed.

Input duplicate variable value.

Create string array for the computer code with the desired length.

Set random number seed.

If duplicates are allowed:

- Loop the length
 - Loop one less than the amount of all colors
 - Set int j = random number 1-8
 - Swap array values to shuffle
 - Assign computer code array from shuffled all colors array.

If duplicates are not allowed:

- Loop one less than the amount of colors
 - Swap values
- Loop the length
 - Assign computer code array from the shuffled all colors array

Create an array to store each guess.

Set int m=1.

Loop for 10 tries

- Output the color choices and ask for input.
- Loop the length of the array.
 - Store each input as a new element in the guess array
 - If current guess is equal to the computer code in the same spot
 - Correct count
 - While m<length
 - If the current guess is anywhere in the computer code
 - Almost count
 - Increase count for almost
 - Reset correct and almost for next guess.
- Output right color wrong place and right color right place
- If all colors are correct
 - You win! Exit game

Output that the game is over and end the game.

Code

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * File:   main.cpp
 * Author: Kelly Williams
 * Purpose: Mastermind Game
 * Created on October 23, 2022, 7:44 PM
 */

#include <iostream> //Input/Output Library
#include <cstdlib> //Random function location
#include <ctime> //Time Library
#include <cmath> //Math Library
#include <iomanip> //Format Library
#include <string>

using namespace std;

//function prototypes

//void getGuess(int);
//void testGuess(int);

int main(int argc, char** argv) {

    //string array for all colors
    string allClrs[8] = {"red", "green", "blue", "yellow", "brown", "orange", "black", "white"};
    cout << "Welcome to Mastermind!\n" << "Choose a sequence length: 4, 6, or 8?\n";
    int length = 0;
    cin >> length;
    cout << "Allow duplicates? Y=1 or N=0\n";
    bool dupe;
    cin >> dupe;
    string code[length];
    int almost = 0;
    int correct = 0;
    //call createCode
    srand(static_cast<unsigned int>(time(0)));
```

```

if (dupe=1)
{
    for (int p=0; p<length; p++)
    {
        for (int i=0; i<7;i++)
        {
            int j = rand() % 8;
            string temp = allClrs[j];
            allClrs[j] = allClrs[i];
            allClrs[i] = temp;
        }
        code[p] = allClrs[p];
    }
}

```

```

if (dupe=0)
{
    for (int i=0; i<7;i++)
    {
        int j = rand() % 8;
        string temp = allClrs[j];
        allClrs[j] = allClrs[i];
        allClrs[i] = temp;
    }
    for (int j=0; j<length; j++)
    {
        code[j] =allClrs[j];
    }
}

```

```

string guess[length];
int m=1;
for (int l=0; l<10; l++)
{
    cout << "Color choices: red, green, blue, yellow, brown, orange, black, white\n"
        << "Enter one color at a time, lowercase:\n";
    for (int n=0; n<length; n++)
    {
        cin >> guess[n];
        if (guess[n] == code[n])
            correct++;
        while (m<length)
        {

```

```
        if (guess[n] == code[m])
            almost++;
        m++;
    }
    //reset almost and correct counters for next guess
    correct = 0;
    almost = 0;
}
cout << "\nRight color, right place: " << correct << endl;
cout << "Right color, wrong place: " << almost << endl;
}

if (correct == length)
{
    cout << "\nYou win!" << endl;
    return 0;
}
cout << "\nOut of guesses. Better luck next time!" << endl;

return 0;
}
```