Mastermind

By: Jennifer Felton

CSC-5 45277

Summer 2016

Project 1

**Intro:**

Mastermind is a game for two players. One is the Codemaker, and the other is a Codebreaker. The game is played on a board with 12 rows, with 6 different color pegs. The Codemaker picks 4 colors and hides them from the Codebreaker. The Codebreaker has 12 chances to guess the right 4 pegs in the right order. As the Codebreaker guesses, the Codemaker places a white or black peg to let the Codebreaker know which pegs are the right color in the right place, or the right color in the wrong place. The Codemaker wins when the code breaker does not guess the combination. Another way to play is to give points for a black/white peg and points for each wrong guess. Whoever has the most points wins. There are 1296 different patterns that can be made.

**Modifications:**

For this version I modified the white/black pegs to let the Codebreaker know what peg in what position was correct. I did this at the time because I wasn’t sure how to do the white/black pegs. With further thought, I think a For Loop could be used to count the white and black pegs for each guess. I am going to try to make that change work in the future, or make a menu to play a hard version or easy version.

**Summary:**

Project size: 177 lines of code

Number of comments: 25

Number of blank lines: 31

Number of variables: 15

**Notable Things I learned:**

I learned to be very careful with how many {} brackets and to make sure they close properly. I also learned how to correctly use “break”. I also learned that testing my game on another person is important, because watching them play it pointed out some things I could better clarify in the directions. I also might have better used a switch in the beginning of the program to choose if you want directions or to play, but my switches seem to go on forever. I am planning on trying some different things with this game in the future to improve my coding skills.

|  |  |  |
| --- | --- | --- |
| **Chapter #** | **Concept** | **Code Line** |
| 1 | Flowchart | Pg 5 of Documentation |
| 1 | PsuedoCode | Pg 6 of Documentation |
| 2.2 | Cout | Lines: 35, 36, 37, 41, 45, 47, 49, 56, 59, 61, 62, 134, 135, 139, 141, 143, 145, 147, 149, 151, 153, 155, 158, 163, 164, 165, 166 |
| 2.3 | #include | 9,10,11 |
| 2.4 | Standard C++ | 9,10,11, 23, 176,177 |
| 2.5 | Variables | 25,26,27,28 |
| 2.6 | Identifiers |  |
| 2.7 | Integer Data Types | 25,26 |
| 2.9 | The char Data Type | 27,28 |
| 2.11 | The bool Data Type | 40, 46, 48, 58, 60, 72, 74, 76, 78, 80, 82, 87, 89, 91, 93, 95, 97, 102, 104, 106, 108, 110, 112, 117, 119,  121, 123, 125, 127, 138, 140, 142, 144, 146, 148, 150, 152, 154, |
| 2.16 | Comments | 2, 3, 4, 5, 9, 10, 11, 12, 14, 16, 18, 20, 24, 25, 26, 27, 28, 30, 33, 46, 51, 54, 64, 66, 67, 72, 73, 75, 77, 78, 79, 81, 84, 87, 102, 131, 169, 170, 171, 172, 175 |
| 2.17 | Programing Style | See entire program |
| 2.18 | Putting it all together | See entire program- it works! |
| 3.1 | The cin object | 38, 57, 136, |
| 3.5 | Named constant | 66,67,68,69, 73, 75, 77, 79, 81, 83, 88, 90, 92, 94, 96, 98, 103, 105, 107, 109, 111, 113, 118, 120, 122, |
| 3.8 | Working with Characters | 27, 28, 73, 75, 77, 79, 81, 83, 88, 90, 92, 94, 96, 98, 103, 105, 107, 109, 111, 113, 118, 120, 122, |
| 4.1 | Relational Operator | 40, 46, 48, 51, 58, 60, 64, 72, 74, 76, 78, 80, 82, 87, 89, 91, 93, 95, 97, 102, 104, 106, 108, 110, 112, 117, 119, 121, 123, 125, 127, 138, 140, 142, 144, 146, 148, 150, 152, 154, 161 |
| 4.2 | If | 40, 58, 60, |
| 4.3 | If/else statement | 40-50, 72-84, 87-99, 102-114, 117-129 |
| 4.5 | Menu Driven Programs | 34-51 |
| 4.6 | Nested If statements | 138-156 |
| 4.7 | Logical Operators | 46, 58, 60, 64,138, |
| 4.8 | Validating User Input | 46, 58, |
| 4.14 | Validating Output Results | 169, 170, 171 |
| 5.2 | While loop for input validation | 34-51 |
| 5.3 | Increment and Decrement Operators | 159 |
| 5.15 | The while loop | 34-51, 55-64, 133-161 |
| 5.7 | Keeping a running total | 159, 164 |
| 5.11 | Breaking out of a loop | 139 |

**Flowchart:**



**Psuedocode:**

Start

#include ctime

#include random number generator

Welcome to MasterMind!

To learn how to play, Press 1;

To play, Press 2;

For 1: Welcome to Mastermind! The objective of the game is to guess the 4 colors, in the correct order, that the Codemaker has picked. The Codemaker will let you know if you are right or wrong. Good Luck!

Hello, Code Breaker.

How many tries do you need? Choose a number between 8 and 12.

Player enters a number not between 8 and 12.

Return back to Hello, Code Breaker. How many tries do you need?

Player enters a number between 8 and 12.

Set rndm1=1-6

Set rndm2=1-6

Set rndm3=1-6

Set rndm4=1-6

IF 1 is R, 2 is B, 3 is G, 4 is Y, 5 is O and 6 is P.

You have nTries to use. Make a guess of 4 colors.

The colors are ‘R’, ‘B’, ‘G’, ‘Y’, ‘O’, ‘P’.

Guess=c1,c2,c3,c4

IF c1==rndm1 output W, else output X

If c2==rndm2 output W, else output X

`If c3==rndm3 output W, else output X

If c4==rndm4 output W, else output X

If none are correct, output XXXX

If one is correct, output WXXX

If two are correct, output WWXX

If three are correct, output WWWX

If all 4 are correct, output “You Win!”

Add ++ to nGuess and check it against the number nTries;

If guesses are not all correct, repeat until nGuess == nTries

Or player guesses correctly.

If player guesses all 4 correctly, output “You Win”

If incorrect until nGuess==nTries,

Output, the correct answer was () () () ().

Do you want to play again? Y, N.

Y returns player to beginning.

No stops the program

End

**Code:**

|  |
| --- |
|  |
| /\* |
|  | \* File: main.cpp |
|  | \* Author: Jennifer Felton |
|  | \* \* Created on July 13, 2016 |
|  | \* Purpose: Mastermind Game |
|  | \*/ |
|  |  |
|  | //system Libraries |
|  | #include <iostream> //input/output Library |
|  | #include <ctime>//time for random number seed |
|  | #include <cstdlib> //random number seed |
|  | using namespace std; //Namespace of the System Libraries |
|  |  |
|  | //User Libraries |
|  |  |
|  | //Global Constants |
|  |  |
|  | //Function Prototypes |
|  |  |
|  | //Execution Begins Here! |
|  |  |
|  |  |
|  | int main(int argc, char\*\* argv) { |
|  | //Declare Variables |
|  | int rndm1,rndm2,rndm3,rndm4;//random numbers to turn to colors |
|  | int nGuess=0, nTries, play;//guesses made, tries needed, play y/n |
|  | char g1, g2, g3, g4; //user choices of colors |
|  | char m1,m2,m3,m4;//turn numbers into colors master colors |
|  |  |
|  | //Set random number seed |
|  | srand(static\_cast<unsigned int>(time(0))); |
|  |  |
|  | //Input Data |
|  | do { |
|  | cout<<"Welcome to Mastermind!"<<endl; |
|  | cout<<"To learn how to play, press 1."<<endl; |
|  | cout<<"To play, press 2"<<endl; |
|  | cin>>play; |
|  |  |
|  | if (play==1) { |
|  | cout<<"The objective of the game is to guess the 4 colors,"<<endl<< |
|  | "in the correct order, that the Codemaker has picked. The Codemaker"<<endl<< |
|  | "will let you know if you are right with a 'O' or wrong with a 'X'."<<endl<< |
|  | "Good Luck! and Have fun!"<<endl; |
|  | cout<<endl; |
|  | } else if (play!=1&&play!=2) { //if its not 1 or not 2 |
|  | cout<<"Please pick again"<<endl; |
|  | } else if (play==2) { |
|  | cout<<"Let's play!"<<endl; |
|  | } |
|  | }while (play!=2); //sends back to start with do while loop |
|  |  |
|  |  |
|  | //Process the Data |
|  | do { |
|  | cout<<"How many tries do you need? Enter a number between 8 and 12"<<endl; |
|  | cin>>nTries; |
|  | if (nTries<8||nTries>12) { |
|  | cout<<"Try again with a number between 8 and 12"<<endl; |
|  | } if (nTries>=8&&nTries<=12) { |
|  | cout<<"You have chosen "<<nTries<<endl; |
|  | cout<<"Good Luck!"<<endl; |
|  | } |
|  | } while (nTries<=7&&nTries<=12); //returns back to number of tries |
|  |  |
|  | rndm1=rand()%6+1; //randomly generate numbers |
|  | rndm2=rand()%6+1; //then equal each number to a letter for a color |
|  | rndm3=rand()%6+1; |
|  | rndm4=rand()%6+1; |
|  |  |
|  |  |
|  | if (rndm1==1){ //turns 1st number to color |
|  | m1='R'; //red |
|  | } else if (rndm1==2){ |
|  | m1='B'; //blue |
|  | } else if (rndm1==3){ |
|  | m1='G'; //green |
|  | } else if (rndm1==4){ |
|  | m1='Y'; //yellow |
|  | } else if (rndm1==5){ |
|  | m1='O'; //orange |
|  | }else if (rndm1==6){ |
|  | m1='P'; //purple |
|  | } |
|  |  |
|  |  |
|  | if (rndm2==1){ //turns 2nd number to color |
|  | m2='R'; |
|  | } else if (rndm2==2){ |
|  | m2='B'; |
|  | } else if (rndm2==3){ |
|  | m2='G'; |
|  | } else if (rndm2==4){ |
|  | m2='Y'; |
|  | } else if (rndm2==5){ |
|  | m2='O'; |
|  | }else if (rndm2==6){ |
|  | m2='P'; |
|  | } |
|  |  |
|  |  |
|  | if (rndm3==1){ //turns 3rd number to color |
|  | m3='R'; |
|  | } else if (rndm3==2){ |
|  | m3='B'; |
|  | } else if (rndm3==3){ |
|  | m3='G'; |
|  | } else if (rndm3==4){ |
|  | m3='Y'; |
|  | } else if (rndm3==5){ |
|  | m3='O'; |
|  | }else if (rndm3==6){ |
|  | m3='P'; |
|  | } |
|  |  |
|  |  |
|  | if (rndm4==1){ //turns 4th number to color |
|  | m4='R'; |
|  | } else if (rndm4==2){ |
|  | m4='B'; |
|  | } else if (rndm4==3){ |
|  | m4='G'; |
|  | } else if (rndm4==4){ |
|  | m4='Y'; |
|  | } else if (rndm4==5){ |
|  | m4='O'; |
|  | }else if (rndm4==6){ |
|  | m4='P'; |
|  | } |
|  |  |
|  | cout<<endl; //some space |
|  |  |
|  | do{ |
|  | cout<<"Pick 4 colors"<<endl; |
|  | cout<<"R,B,G,Y,O, or P"<<endl; |
|  | cin>>g1>>g2>>g3>>g4; |
|  |  |
|  | if (g1==m1&&g2==m2&&g3==m3&&g4==m4){ |
|  | cout<<"You won!"<<endl;break; |
|  | } if (g1==m1){ |
|  | cout<<"O"; |
|  | } else if (g1!=m1){ |
|  | cout<<"X"; |
|  | } if (g2==m2){ |
|  | cout<<"O"; |
|  | } else if (g2!=m2){ |
|  | cout<<"X"; |
|  | }if (g3==m3){ |
|  | cout<<"O"; |
|  | } else if (g3!=m3){ |
|  | cout<<"X"; |
|  | }if (g4==m4){ |
|  | cout<<"O"; |
|  | } else if (g4!=m4){ |
|  | cout<<"X"; |
|  | } |
|  |  |
|  | cout<<endl; |
|  | nGuess++; |
|  |  |
|  | }while (nGuess<nTries); |
|  |  |
|  | cout<<"Game Over"<<endl; |
|  | cout<<"You tried "<<nGuess<<" times"<<endl; |
|  | cout<<"The code was "<<m1<<"-"<<m2<<"-"<<m3<<"-"<<m4<<endl; |
|  | cout<<"Thanks for Playing Mastermind!"<<endl; |
|  |  |
|  |  |
|  | //I left these here to show how I tested as I went along |
|  | //cout<<g1<<"-"<<g2<<"-"<<g3<<"-"<<g4<<endl; //this is for testing delete when done |
|  | //cout<<m1<<"--"<<m2<<"--"<<m3<<"--"<<m4<<endl;//this is for testing delete when done |
|  | //cout<<nGuess<<"guesses should go up"<<endl; |
|  |  |
|  |  |
|  | //Exit Stage Right |
|  | return 0; |
|  | } |
|  |  |