Mastermind (Board Game)

Minkyu Ray Park

CIS 17A

Summer 2019

46097

**About the Mastermind Board Game:**

Mastermind is a logic utilizing coding game that challenges user to solve the random codes.

Total of 6 colors are available for level 1 and 2!

For difficulty level 1 and 2, 4 random colors will be chosen.

Total of 8 colors are available for level 3 and 4!

For difficulty level 3 and 4, 5 random colors will be chosen.

User must guess all the right colors as well as getting the colors in the right place to win the game.

The colors may overlap!!

There will be only 10 turns for the user to experiment different color codes. User must guess within 10 tries.

In the difficulty level 1, the Program will prompt the user if the color is right and the place is right INDIVIDUALLY.

In the difficulty level 2, the Program will prompt the user if the color is right and the place is right BROADLY.

In the difficulty level 3, the Program will prompt the user if the color is right and the place is right with X and O. 5th answer will come out as HIDDEN as the user is supposed to guess the right color for the 5th without hints.

In the difficulty level 4, the Program will prompt the user if the color is right and the place is right with X and O. 4th and 5th answers will be HIDDEN as the user is supposed to guess the right colors for the 4th and 5th without hints.

Difficulty Level 1 and 2 will have 4 answers and 10 tries total.

Difficulty Level 3 and 4 will have 5 answers and 10 tries total.

The user must think it through to solve the code.

**How I started coding:**

Utilized random number generator.

Begin creating structures and EMUMs.

Created Classes and header file.

Declared all the variables needed and created two different levels.

The class variables were declared separately for level 3 and 4.

Level 1 will show the user if each of the guesses are in the right color or right place specifically.

Level 2 will show the user plain numbers to make hint broader for them to figure it out by using more logics.

Users are supposed to input first Alphabet of colors for level 1 and 2 (R B R G).

Level 3 and 4 will show the user X or O. But 4th and 5th colors may be HIDDEN with no hints.

Users are supposed to input the whole colors for level 3 and 4 in CAPS (RED BLUE ORANGE PURPLE WHITE).

Opened file to stream all the outputs into.

Converted random numbers (1-6) into colors.

Used structure to convert random numbers (1-8) into colors.

Created while loop for play again function.

Created Try, Grab, Throw for User choice of difficulty level.

Used if statement to help the user pick which level of difficulty they would like to play in and to check if User guess is correct.

To convert numbers into characters and to set the random answers, I utilized Ternary Op and one-dimensional array using structure function.

To convert numbers into string and to set the random answers, within the class, utilized Inheritance and if-statements.

Mainly used Array to organize the numbers.

Created multiple classes and linked them in a way to organize the number of answers and guesses.

For Loop was used to make turn counter.

Template was created to make turn counter for classes (10).

Used Structures, if-statements, ternary Op, boolean statements, arrays, classes, enums, constructors, header file h, inheritance, templates and exceptions to run the game.

When all Boolean codes matched, returned true, displayed Win else it displayed Lose.

When all the classes.guesses and classes.answers matched, displayed Win else it displayed Lose.

**Header Code (Number.h):**

/\*

\* 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: Number.h

\* Author: pmk0625

\*

\* Created on July 26, 2019, 5:37 PM

\*/

#ifndef NUMBER\_H

#define NUMBER\_H

class Number{

protected:

int number1, number2, number3, number4, number5;

public:

Number(){

number1=0;

number2=0;

number3=0;

number4=0;

number5=0;

}

void setNumber1(int);

void setNumber2(int);

void setNumber3(int);

void setNumber4(int);

void setNumber5(int);

int getNumber1()const;

int getNumber2()const;

int getNumber3()const;

int getNumber4()const;

int getNumber5()const;

};

void Number::setNumber1(int x1){

number1 = x1;

}

void Number::setNumber2(int x2){

number2 = x2;

}

void Number::setNumber3(int x3){

number3 = x3;

}

void Number::setNumber4(int x4){

number4 = x4;

}

void Number::setNumber5(int x5){

number5 = x5;

}

int Number::getNumber1()const{

return number1;

}

int Number::getNumber2()const{

return number2;

}

int Number::getNumber3()const{

return number3;

}

int Number::getNumber4()const{

return number4;

}

int Number::getNumber5()const{

return number5;

}

#endif /\* NUMBER\_H \*/

**Code (Main.cpp):**

/\*

\* File: main.cpp

\* Author: Minkyu R. Park

\* Created on July 17, 2019, 08:30 PM

\* Purpose: Project 2: Mastermind V8

\*/

//System Libraries

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <cstring>

#include <fstream>

using namespace std;

//User Libraries

#include "Number.h"

enum colors {

RED, BLUE, GREEN, YELLOW, ORANGE, //ENUM COLORS

PURPLE

};

//Structures

struct guesses{

char guess1, guess2, guess3, guess4; //STRUCTURE GUESSES

};

//Template

template<class T>

T timer(T t){

int time;

time=t+1;

return time;

}

//Classes

class Letters:public Number{

private:

string color1, color2, color3, color4, color5;

public:

Letters(){

color1 = "";

color2 = "";

color3 = "";

color4 = "";

color5 = "";

}

void convert(){

int n1 = getNumber1();

if (n1 == 0){

color1 = "RED";

}else if (n1 == 1){

color1 = "BLUE";

}else if (n1 == 2){

color1 = "GREEN";

}else if (n1 == 3){

color1 = "YELLOW";

}else if (n1 == 4){

color1 = "ORANGE";

}else if (n1 == 5){

color1 = "PURPLE";

}else if (n1 == 6){

color1 = "BLACK";

}else{

color1 = "WHITE";

}

int n2 = getNumber2();

if (n2 == 0){

color2 = "RED";

}else if (n2 == 1){

color2 = "BLUE";

}else if (n2 == 2){

color2 = "GREEN";

}else if (n2 == 3){

color2 = "YELLOW";

}else if (n2 == 4){

color2 = "ORANGE";

}else if (n2 == 5){

color2 = "PURPLE";

}else if (n2 == 6){

color2 = "BLACK";

}else{

color2 = "WHITE";

}

int n3 = getNumber3();

if (n3 == 0){

color3 = "RED";

}else if (n3 == 1){

color3 = "BLUE";

}else if (n3 == 2){

color3 = "GREEN";

}else if (n3 == 3){

color3 = "YELLOW";

}else if (n3 == 4){

color3 = "ORANGE";

}else if (n3 == 5){

color3 = "PURPLE";

}else if (n3 == 6){

color3 = "BLACK";

}else{

color3 = "WHITE";

}

int n4 = getNumber4();

if (n4 == 0){

color4 = "RED";

}else if (n4 == 1){

color4 = "BLUE";

}else if (n4 == 2){

color4 = "GREEN";

}else if (n4 == 3){

color4 = "YELLOW";

}else if (n4 == 4){

color4 = "ORANGE";

}else if (n4 == 5){

color4 = "PURPLE";

}else if (n4 == 6){

color4 = "BLACK";

}else{

color4 = "WHITE";

}

int n5 = getNumber5();

if (n5 == 0){

color5 = "RED";

}else if (n5 == 1){

color5 = "BLUE";

}else if (n5 == 2){

color5 = "GREEN";

}else if (n5 == 3){

color5 = "YELLOW";

}else if (n5 == 4){

color5 = "ORANGE";

}else if (n5 == 5){

color5 = "PURPLE";

}else if (n5 == 6){

color5 = "BLACK";

}else{

color5 = "WHITE";

}

}

string print1(){

return color1;

}

string print2(){

return color2;

}

string print3(){

return color3;

}

string print4(){

return color4;

}

string print5(){

return color5;

}

};

class Level3:public Letters{

private:

string guess1, guess2, guess3, guess4, guess5;

public:

Level3(){

guess1 = "";

guess2 = "";

guess3 = "";

guess4 = "";

guess5 = "";

}

void setGuess1(string g1){

this->guess1 = g1;

}

void setGuess2(string g2){

this->guess2 = g2;

}

void setGuess3(string g3){

this->guess3 = g3;

}

void setGuess4(string g4){

this->guess4 = g4;

}

void setGuess5(string g5){

this->guess5 = g5;

}

string getGuess1(){

return guess1;

}

string getGuess2(){

return guess2;

}

string getGuess3(){

return guess3;

}

string getGuess4(){

return guess4;

}

string getGuess5(){

return guess5;

}

};

//Global Constants - Math/Physics Constants, Conversions,

// 2-D Array Dimensions

const int SIZE=4;

const int NUM\_GUESS=10;

//Function Prototypes

void displayIntro();

void answer(colors , char \*);

bool check1(guesses \*, char []);

bool check2(char \*, char []);

char LEVEL(char);

//Execution Begins Here

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

//Set the random number seed

srand(static\_cast<unsigned int>(time(0)));

//Your Code Goes here

guesses guess[NUM\_GUESS];

char ans[SIZE],test[SIZE]; //Number of Answers Array

bool playAgain = true;

char symbol;

char choice;

string level; //String level comment

level = "\*\*\*DIFFICULTY LEVEL\*\*\* ";

//Reading an Entire Binary File

streampos memory;

char \*memBlock;

//Opening File

ifstream file ("stats.txt", ios::in | ios::binary | ios::ate);

if (file.is\_open()){

memory = file.tellg();

memBlock = new char [memory];

file.seekg(0, ios::beg);

file.read(memBlock, memory);

file.close();

//Play Again Loop

while (playAgain){

choice = LEVEL(choice);

//Difficulty level SET

if (choice >= '1' && choice <= '4'){

switch (choice){

case '1':

{

cout<<level<<" 1 "<<endl;

//Randomize the Answer

colors set = static\_cast<colors>(rand()%PURPLE);

//Set the Answer

answer(set, ans);

//Output data

displayIntro(); //Display instruction

check1(guess, ans);//Check if User gets the correct answer

}

case '2':

{

cout<<level<<" 2 "<<endl;

//Randomize the Answer

colors set = static\_cast<colors>(rand()%PURPLE);

//Set the Answer

answer(set, ans);

//Output data

displayIntro(); //Display instruction

check2(test, ans);

}

case '3':

{

cout<<level<<" 3 "<<endl;

int t=0;

int n1, n2, n3, n4, n5;

string g1, g2, g3, g4, g5;

//Calling Classes

Letters colors;

Level3 user;

//Set random answers individually

n1=rand()%8;

n2=rand()%8;

n3=rand()%8;

n4=rand()%8;

n5=rand()%8;

//Set the numbers in variables

colors.setNumber1(n1);

colors.setNumber2(n2);

colors.setNumber3(n3);

colors.setNumber4(n4);

colors.setNumber5(n5);

colors.convert();

//Introduction

cout<<"This is difficulty level 3! Please input the color as WHOLE WORDS, "

<<"Pick 5 colors! and ALL CAPS"<<endl;

cout<<"In the format: For example, RED BLUE BLUE BLACK PURPLE"<<endl;

cout<<"To adjust the difficulty level, Answer 5 will be HIDDEN at all times"<<endl;

cout<<"Good Luck!!"<<endl;

//User guess

for (int i=0; i<NUM\_GUESS; i++){

t++;

cout<<"Please input your guesses: "<<endl;

cout<<"Again, the colors are RED, BLUE, GREEN, YELLOW,"

<<"ORANGE, PURPLE, BLACK, WHITE"<<endl;

cin>>g1>>g2>>g3>>g4>>g5;

//Setting guesses in variables

user.setGuess1(g1);

user.setGuess2(g2);

user.setGuess3(g3);

user.setGuess4(g4);

user.setGuess5(g5);

//Checking if the user guess is correct or incorrect

if (user.getGuess1() == colors.print1()){

cout<<"O"<<endl;

}else if (user.getGuess1() == colors.print2() || user.getGuess1() == colors.print3() ||

user.getGuess1() == colors.print4() || user.getGuess1() == colors.print5()){

cout<<"X"<<endl;

}else{

cout<<" "<<endl;

}

if (user.getGuess2() == colors.print2()){

cout<<"O"<<endl;

}else if (user.getGuess2() == colors.print1() || user.getGuess2() == colors.print3() ||

user.getGuess2() == colors.print4() || user.getGuess2() == colors.print5()){

cout<<"X"<<endl;

}else{

cout<<" "<<endl;

}

if (user.getGuess3() == colors.print3()){

cout<<"O"<<endl;

}else if (user.getGuess3() == colors.print1() || user.getGuess3() == colors.print2() ||

user.getGuess3() == colors.print4() || user.getGuess3() == colors.print5()){

cout<<"X"<<endl;

}else{

cout<<" "<<endl;

}

if (user.getGuess4() == colors.print4()){

cout<<"O"<<endl;

}else if (user.getGuess4() == colors.print1() || user.getGuess4() == colors.print2() ||

user.getGuess4() == colors.print3() || user.getGuess4() == colors.print5()){

cout<<"X"<<endl;

}else{

cout<<" "<<endl;

}

if (user.getGuess5() == colors.print5()){

cout<<"HIDDEN"<<endl;

}else if (user.getGuess5() == colors.print1() || user.getGuess5() == colors.print2() ||

user.getGuess5() == colors.print3() || user.getGuess5() == colors.print4()){

cout<<"HIDDEN"<<endl;

}else{

cout<<"HIDDEN"<<endl;

}

//User guess matches random answers -> WIN

if (user.getGuess1() == colors.print1() && user.getGuess2() == colors.print2() &&

user.getGuess3() == colors.print3() && user.getGuess4() == colors.print4() &&

user.getGuess5() == colors.print5()){

cout<<"You have beaten the game! Congrats! You win!"<<endl;

exit(0);

}

timer(t);

cout<<"Number of tries: "<<t<<endl;

//Turn ends -> LOSE

if (i == NUM\_GUESS-1){

cout<<"Sorry you lost! Try Again!"<<endl;

cout<<"The Answer was: "<<endl;

cout<<colors.print1()<<" "<<colors.print2()<<" "<<colors.print3()<<" "

<<colors.print4()<<" "<<colors.print5()<<endl;

exit(0);

}

}

}

case '4':

{

cout<<level<<" 4 "<<endl;

int t=0;

int n1, n2, n3, n4, n5;

string g1, g2, g3, g4, g5;

//Calling Classes

Letters colors;

Level3 user;

//Set random answers individually

n1=rand()%8;

n2=rand()%8;

n3=rand()%8;

n4=rand()%8;

n5=rand()%8;

//Set answers in variables

colors.setNumber1(n1);

colors.setNumber2(n2);

colors.setNumber3(n3);

colors.setNumber4(n4);

colors.setNumber5(n5);

colors.convert();

//Introduction

cout<<"This is difficulty level 4! Please input the color as WHOLE WORDS, "

<<"Pick 5 colors! and ALL CAPS"<<endl;

cout<<"In the format: For example, RED BLUE BLUE BLACK PURPLE"<<endl;

cout<<"To Adjust the difficulty level, Last 2 Answers will be HIDDEN at all times"<<endl;

cout<<"Good Luck!!"<<endl;

//User guess

for (int i=0; i<NUM\_GUESS; i++){

t++;

cout<<"Please input your guesses: "<<endl;

cout<<"Again, the colors are RED, BLUE, GREEN, YELLOW,"

<<"ORANGE, PURPLE, BLACK, WHITE"<<endl;

cin>>g1>>g2>>g3>>g4>>g5;

//Setting guesses in variables

user.setGuess1(g1);

user.setGuess2(g2);

user.setGuess3(g3);

user.setGuess4(g4);

user.setGuess5(g5);

//Checking if the user guess is correct or incorrect

if (user.getGuess1() == colors.print1()){

cout<<"O"<<endl;

}else if (user.getGuess1() == colors.print2() || user.getGuess1() == colors.print3() ||

user.getGuess1() == colors.print4() || user.getGuess1() == colors.print5()){

cout<<"X"<<endl;

}else{

cout<<" "<<endl;

}

if (user.getGuess2() == colors.print2()){

cout<<"O"<<endl;

}else if (user.getGuess2() == colors.print1() || user.getGuess2() == colors.print3() ||

user.getGuess2() == colors.print4() || user.getGuess2() == colors.print5()){

cout<<"X"<<endl;

}else{

cout<<" "<<endl;

}

if (user.getGuess3() == colors.print3()){

cout<<"O"<<endl;

}else if (user.getGuess3() == colors.print1() || user.getGuess3() == colors.print2() ||

user.getGuess3() == colors.print4() || user.getGuess3() == colors.print5()){

cout<<"X"<<endl;

}else{

cout<<" "<<endl;

}

if (user.getGuess4() == colors.print4()){

cout<<"HIDDEN"<<endl;

}else if (user.getGuess4() == colors.print1() || user.getGuess4() == colors.print2() ||

user.getGuess4() == colors.print3() || user.getGuess4() == colors.print5()){

cout<<"HIDDEN"<<endl;

}else{

cout<<"HIDDEN"<<endl;

}

if (user.getGuess5() == colors.print5()){

cout<<"HIDDEN"<<endl;

}else if (user.getGuess5() == colors.print1() || user.getGuess5() == colors.print2() ||

user.getGuess5() == colors.print3() || user.getGuess5() == colors.print4()){

cout<<"HIDDEN"<<endl;

}else{

cout<<"HIDDEN"<<endl;

}

//User guess matches random answers -> WIN

if (user.getGuess1() == colors.print1() && user.getGuess2() == colors.print2() &&

user.getGuess3() == colors.print3() && user.getGuess4() == colors.print4() &&

user.getGuess5() == colors.print5()){

cout<<"You have beaten the game! Congrats! You win!"<<endl;

exit(0);

}

timer(t);

cout<<"Number of tries: "<<t<<endl;

//Turn ends -> LOSE

if (i == NUM\_GUESS-1){

cout<<"Sorry you lost! Try Again!"<<endl;

cout<<"The Answer was: "<<endl;

cout<<colors.print1()<<" "<<colors.print2()<<" "<<colors.print3()<<" "

<<colors.print4()<<" "<<colors.print5()<<endl;

exit(0);

}

}

}

}

}

cout<<"Would you like to play again? (Y or N)"<<endl;

cin>>symbol;

playAgain = (symbol == 'Y' || symbol == 'y') ? true : false;

}

delete[] memBlock;

}

else cout<<"Unable to open file";

//Exit stage right!

return 0;

}

//Display intro and instructions on how to play

void displayIntro()

{

cout << "======================= Welcome to Mastermind =======================" << endl;

cout << "Mastermind is a game of logic." << endl;

cout << "The goal is to guess the correct four color combination" << endl;

cout << "by using your previous guesses as clues." << endl;

cout << "Find specific colors and the orders of the colors"

<<" that matches exactly with the answer" << endl;

cout << "To guess, enter the first letter of each color." << endl;

cout << "The colors consist of : R=RED, B=BLUE, G=GREEN, Y=YELLOW,"

<< " O=ORANGE,P=PURPLE" << endl;

cout << "An sample guess would be look like this: R R G B" << endl;

cout << "=====================================================================" << endl;

cout << endl;

}

//Set answers for Difficulty Level 1

void answer(colors set, char \*ans) {

\*(ans + 0) = ((set + 0) == 0 ? 'R' : ((set + 0) == 1 ? 'B' :

((set + 0) == 2 ? 'G' : ((set + 0) == 3 ? 'Y' :

((set + 0) == 4 ? 'O' : 'P')))));

\*(ans + 1) = ((set + 1) == 0 ? 'R' : ((set + 1) == 1 ? 'B' :

((set + 1) == 2 ? 'G' : ((set + 1) == 3 ? 'Y' :

((set + 1) == 4 ? 'O' : 'P')))));

\*(ans + 2) = ((set + 2) == 0 ? 'R' : ((set + 2) == 1 ? 'B' :

((set + 2) == 2 ? 'G' : ((set + 2) == 3 ? 'Y' :

((set + 2) == 4 ? 'O' : 'P')))));

\*(ans + 3) = ((set + 3) == 0 ? 'R' : ((set + 3) == 1 ? 'B' :

((set + 3) == 2 ? 'G' : ((set + 3) == 3 ? 'Y' :

((set + 3) == 4 ? 'O' : 'P')))));

}

//Checking if the User gets the answer right for Difficulty Level 1

bool check1(guesses \*guess, char ans[]){

bool colorR[4];

bool placeR[4];

for (int i=0; i<4; i++){

colorR[i]=false;

placeR[i]=false;

}

for (int i=0; i<NUM\_GUESS; i++){

cout<<"Again, the colors consist of : R=RED, B=BLUE, G=GREEN, Y=YELLOW,"

<< " O=ORANGE,P=PURPLE" <<endl;

cout <<"Please input your guesses! "<< endl;

cin>>guess[0].guess1>>guess[1].guess2>>guess[2].guess3>>guess[3].guess4;

if (guess[0].guess1 == ans[0]){

placeR[0] = true;

cout<<"You guessed Answer 1 Right!"<<endl;

}else if (guess[0].guess1 == ans[1] || guess[0].guess1 == ans[2] || guess[0].guess1 == ans[3]){

colorR[0] = true;

cout<<"Right color but Wrong place"<<endl;

}else{

placeR[0] == false;

colorR[0] == false;

cout<<"Try again"<<endl;

}

if (guess[1].guess2 == ans[1]){

placeR[1] = true;

cout<<"You guessed Answer 2 Right!"<<endl;

}else if (guess[1].guess2 == ans[0] || guess[1].guess2 == ans[2] || guess[1].guess2 == ans[3]){

colorR[1] = true;

cout<<"Right color but Wrong place"<<endl;

}else{

placeR[1] == false;

colorR[1] == false;

cout<<"Try again"<<endl;

}

if (guess[2].guess3 == ans[2]){

placeR[2] = true;

cout<<"You guessed Answer 3 Right!"<<endl;

}else if (guess[2].guess3 == ans[0] || guess[2].guess3 == ans[1] || guess[2].guess3 == ans[3]){

colorR[2] = true;

cout<<"Right color but Wrong place"<<endl;

}else{

placeR[2] == false;

colorR[2] == false;

cout<<"Try again"<<endl;

}

if (guess[3].guess4 == ans[3]){

placeR[3] = true;

cout<<"You guessed Answer 4 Right!"<<endl;

}else if (guess[3].guess4 == ans[0] || guess[3].guess4 == ans[1] || guess[3].guess4 == ans[2]){

colorR[3] = true;

cout<<"Right color but Wrong place"<<endl;

}else{

placeR[3] == false;

colorR[3] == false;

cout<<"Try again"<<endl;

}

//End game

if (placeR[0] && placeR[1] && placeR[2] && placeR[3]){

cout<<"You won! Congrats!"<<endl;

return true;

}else if (i == NUM\_GUESS-1){

cout<<"You lost! Sorry, try again..."<<endl;

cout<<"The answer was..."<<endl;

cout<<ans[0]<<" "<<ans[1]<<" "<<ans[2]<<" "<<ans[3]<<endl;

return false;

}

cout<<endl;

}

cout<<endl;

}

//Checking if the User gets the answer right for Difficulty Level 2

bool check2(char \*test, char \*ans){

int allRight=0;

int justColor=0;

int temp;

bool colorR[4];

bool placeR[4];

bool allAnswers;

bool inList = false;

for (int i = 0; i < 4; i++){

colorR[i] = false;

placeR[i] = false;

}

cout<<"This is more difficult version!"<<endl;

cout<<"The more tries accumulated, hint becomes more confusing because it will add up"<<endl;

cout<<endl;

cout<<"GOOD LUCK!!!"<<endl;

cout<<endl;

for (int i=0; i<NUM\_GUESS; i++){

cout<<"Again, the colors consist of : R=RED, B=BLUE, G=GREEN, Y=YELLOW,"

<<" O=ORANGE,P=PURPLE"<<endl;

cout<<"Please input your guesses! "<< endl;

cin>>test[0]>>test[1]>>test[3]>>test[4];

allAnswers = false;

for (int i=0; i<4; i++){

if (test[i]==ans[i]){

allRight++;

placeR[i] = true;

}

}

for (int i=0; i<4; i++){

if (!placeR[i]){

for (int j=0; j<4; j++){

if (!placeR[j] && i != j){

if (test[i] == ans[j] && !colorR[j]){

inList = true;

temp = j;

}

}

}

if (inList){

justColor++;

inList = false;

colorR[temp] = true;

}

}

}

//End game

if (allRight == 4){

cout<<"You won! Congrats!"<<endl;

allAnswers = true;

return true;

}else{

cout<<"Just Color Correct-> "<<justColor<<" "<<

"Color and Place Correct -> "<<allRight<<endl;

allAnswers = false;

}

if (i == NUM\_GUESS-1){

cout<<"You lost! Sorry, try again..."<<endl;

cout<<"The answer was..."<<endl;

cout<<ans[0]<<" "<<ans[1]<<" "<<ans[2]<<" "<<ans[3]<<endl;

return false;

}

cout<<endl;

}

cout<<endl;

}

//Declaring Level using try, catch.

//Player will enter 1 through 4 to choose the difficulty level.

char LEVEL(char choice){

try{

cout << "Welcome to the Logic based game called Mastermind!"<<endl;

cout << "This version of Mastermind is more than just code solving game!"<<endl;

cout << "Think of it as code and riddle solving game"<<endl;

cout << "To continue, please choose from the following Menu" << endl;

cout << "Type 1 for MasterMind Difficulty Level 1" << endl;

cout << "Type 2 for MasterMind Difficulty Level 2" << endl;

cout << "Type 3 for MasterMind Difficulty Level 3" << endl;

cout << "Type 4 for MasterMind Difficulty Level 4" << endl;

cin>>choice;

if (choice < 1 || choice > 4){

throw choice;

}

}

catch (char x){

cout<<"Choice number invalid! "<<endl;

}

if (choice == 1 || choice == 2 || choice == 3 || choice == 4){

return choice;

}

}

**END OF CODE.**

**Thank You for Reviewing.**

**Reference:**

“Starting Out with C++: From Control Structures through Objects” Gaddis, Tony. 8th Edition. (Textbook)