Final Project

<MasterMind>

CIS 5
Name: Cassandra Leon
Date: June 3, 2017

Introduction

Title: Mastermind

      Mastermind is a game where a randomly generated code is produced and the player has the guess that code. The code can be any combination of a series of three numbers. The player puts in their guess and the program will output where the number is correct and whether it is in the right place. The place has a limited number of guesses and wins when they correctly put in the code. They lose if they exceed the maximum number of guesses. I also included a version of the game where the program will tell the player if the number is too high or too low.

Summary

Project size: 290 lines
The number of variables: 19

      I implemented the concepts we learned in class and tried to do so where there were as few lines as possible. Compared to my previous project, I am much happier with this one. It took me a about a week to program it and I spent an hour or two each day working on it. I faced some difficulties that were solved by referencing the textbook and past homework and lab assignments but one challenge with my code is still unresolved despite countless hours trying to solve it.

Description

This program is made up of several functions and depends on if statements and loops to test each guess the player inputs.

Pseudo Code

declare void menu function
Set the random number seed
Declare Variables
initialize variables
Begin time calculation
      display menu and players chooses game

Check if player wishes to leave program

switch case to play each game

Initialize function for game 2

Separate integer into digits

If statement to check if guess is valid

If statement to check is guess is correct

If statements to check correct position and number

Output correct numbers and position

Output score

Open the file

Read the File

Close the file

Open the file

Write to the File

Close the file

Initialize function for game 2

If statement to check if guess is valid

If statement to check is guess is correct

Separate guess into digits

If Statements to check if guess is too high or too low

Output score

Open the file

Read the File

Close the file

Open the file

Write to the File

Close the file

Player chooses to play again or not

End time of Game play,output time

Display High score

Fill Function with high score values

Sort Highscores

Map inputs to outputs or process the data

Output the transformed data

Exit Program

Major Variables

Type	Name	Description	Location
char	outName	Name of file to output scores to	Line 143
integer	guess	The users guess at the code	Line 33
	tries	How many tries the user takes	Line 34
	dright dmidd dleft	The code separated into digits, the user guess	Line 35
	aright amidd aleft	The code separated into digits, the answer	Line 36
	cNum	The number of correct digits in the code	Line 37
	cPos	The number of correct digits in the correct position in the code	38
	choice	Which game the player wants to play	39
	choice2	Whether the player would like to play again	39

	score	Array holding the score the player got	40
const	SIZE	Size of the array holding the highscores	41
	array	Array holding the highscores	42

Reference

1. Textbook
2. Previous problems

Program

```
/*
 * File:   main.cpp
 * Author: Cassandra Leon
 * Purpose: This is the Game of Mastermind.
 * Created on 05/24/17
 */

#include <iostream>
#include <cstdlib> // srand and rand function
#include <ctime> // time to set the seed
#include <cmath> // math function
#include <iomanip> //setw function
#include <fstream>  //File I/O
#include <string>
using namespace std;

//declare void menu function
void menu(void);
int game1(int,int,int,int,int,int,int,int,int,int,int,int []);
int game2(int,int,int,int,int,int,int,int,int,int,int,int []);
void win();
void rdFile();
void wrtFile(int [],int);
void scrSort(int [],int);
```

```cpp
void prntSrt(int [], int, int);
void filSort(int [],int);

int main(int argc, char** argv) {
 //Set the random number seed
    srand(static_cast<unsigned int>(time(0)));

//Declare Variables
  int guess;              //User's guess
  int tries;              //number of tries
  int dright, dmidd, dleft; //digits in guess
  int aright, amidd, aleft; //digits in answer
  int cNum;               //correct number
  int cPos;               //correct position
  int choice,choice2;
  int score[]={};
  const int SIZE=10;
  int array[SIZE]={};

  //initialize counter
  tries=1;

  //Begin time calculation
  int beg=time(0);

  //do while loop to continue game if player wishes
  do{

 //display menu and players chooses game
 menu();
 cin>>choice;
 // Check if player wishes to leave program
 if((!(choice==2))&&(!(choice==1))){
    cout<<"Goodbye!"<<endl;
    exit(choice);
 }
 //switch case to play each game
 switch(choice){
    case 1: {
```

```cpp
        int randint = 100 + rand()%899;
        int answer=randint;     //Code to Break
        //Separate integer into digits
        aright = answer % 10;
        aleft = answer / 100;
        amidd = (answer / 10) % 10;
        game1(tries,guess,answer,dleft,dmidd,dright,
            aleft,amidd,aright,cPos,cNum,score);
        rdFile();
        wrtFile(score,tries);
          break;
            }
      case 2:{
        int randint = 100 + rand()%899;
        int answer=randint;     //Code to Break
        //Separate integer into digits
        aright = answer % 10;
        aleft = answer / 100;
        amidd = (answer / 10) % 10;
        game2(tries,guess,answer,dleft,dmidd,dright,
            aleft,amidd,aright,cPos,cNum,score);
        rdFile();
        wrtFile(score,tries);
          break;
            }
          }
  //player chooses to play again or not
  cout<<"To play again type 3."<<endl;
  cin>>choice2;
  }while(choice2==3);
  //End time of Game play,output time
 int end=time(0);
cout<<"Total time played = "<<end-beg<<" seconds."<<endl;

//High score
cout<<endl;
cout<<"  High Scores!"<<endl;
cout<<"~~~~~~~~~~~~~~~~~~~~"<<endl;
```

```cpp
    filSort(array,SIZE);
    //Map inputs to outputs or process the data
    scrSort(array,SIZE);
    //Output the transformed data
    prntSrt(array,SIZE,1);
    cout<<"Goodbye!"<<endl;

    return 0;
}

void filSort(int a[],int n){
    for(int indx=0;indx<n;indx++){
        a[indx]=3000+rand()%1999;//Fill with 2 digit number
    }
}

void scrSort(int a[],int n){
    for(int pos=0;pos<n-1;pos++){
        for(int indx=pos+1;indx<n;indx++){
            if(a[pos]>a[indx]){
                int temp=a[pos];
                    a[pos]=a[indx];
                    a[indx]=temp;
            }
        }
    }
}

void prntSrt(int a[],int n,int perLine){
    for(int indx=0;indx<n;indx++){
        cout<<setw(9)<<a[indx]<<" ";
        if(indx%perLine==(perLine-1))cout<<endl;
    }
    cout<<endl;
}

void wrtFile(int score[],int tries){
    //Declare variables
    ofstream out;
```

```cpp
    //Open the file
    char outName[]="mastermind.dat"; //Character Array Name
    out.open(outName);            //Open the Output file
    out<<score[tries]<<endl;
    //Close the file
    out.close();
}

void rdFile(){
    //Declare variables
    ifstream in;
    //Open the file
    string inName="mastermind.dat";   //String Name
    in.open(inName.c_str());        //Open the Input file
    //Close the file
    in.close();
}

int game2(int tries,int guess,int answer,int dleft,int dmidd,int dright,int aleft,
        int amidd,int aright,int cPos,int cNum,int score[]){
    while (!(guess == answer)) {
        cout << "Guess #" << tries << ": Enter a number between 100 and 999: ";
        cout<<answer;
        cin >> guess;
         int choice;
            //check guess is valid
        if (guess >= 100 && guess <= 999) {
       //check if answer is right
          if (guess == answer) {
             win();
            cout << "Right!  You took " << tries<< " move";
            if (tries != 1) cout << "s" ;
            cout << "." << endl;
          } else {
       //separate guess into digits
            dright = guess % 10;
            dleft = guess / 100;
            dmidd = (guess / 10) % 10;
            if(dleft>aleft){
```

```cpp
            cout<<"First digit too high."<<endl;
        }
        if(dleft<aleft){
            cout<<"First digit too low."<<endl;
        }
        if(dleft==aleft){
            cout<<"First digit correct!"<<endl;
        }
        if(dmidd>amidd){
            cout<<"Second digit too high."<<endl;
        }
        if(dmidd<amidd){
            cout<<"Second digit too low."<<endl;
        }
        if(dmidd==amidd){
            cout<<"Second digit correct!"<<endl;
        }
        if(dright>aright){
            cout<<"Last digit too high."<<endl;
        }
        if(dright<aright){
            cout<<"Last digit too low."<<endl;
        }
        if(dright==aright){
            cout<<"Third digit correct!"<<endl;
        }
        }
    } else {
      cout << "Between 1 and 999, please."<<endl;
    }
    tries++;
//output score
    if(guess==answer){
    cout<<"Your score is:"<<endl;
    score[tries]=5000*1/tries;
    cout<<score[tries]<<endl;
    }

        }
```

```cpp
    return tries;
}
int game1(int tries,int guess,int answer,int dleft,int dmidd,
     int dright,int aleft,int amidd,int aright,int cPos,int cNum,int score[]){
   while (!(guess == answer)) {
       cout << "Guess #" << (int)tries << ": Enter a number between 100 and 999: ";
       cin >> guess;
      //check guess is valid
       if (guess >= 100 && guess <= 999) {
      //check if answer is right
         if (guess == answer) {
            win();
           cout << "Right!  You took " << tries<< " move";
           if (tries != 1) cout << "s" ;
           cout << "." << endl;
         } else {
      //separate guess into digits
           dright = guess % 10;
           dleft = guess / 100;
           dmidd = (guess / 10) % 10;
      //Position counter, how many in correct position
           cPos = 0;
           if (dright == aright) cPos++;
           if (dleft == aleft) cPos++;
           if (dmidd == amidd) cPos++;
      //Number counter, how many correct numbers
           cNum = 0;
           if (dright == aright || dright == amidd || dright == aleft) cNum++;
           if (dmidd == aright || dmidd == amidd || dmidd == aleft) cNum++;
           if (dleft == aright || dleft == amidd || dleft == aleft) cNum++;
      //Output correct position and correct number
           cout << "Correct position: " << cPos << endl;
           cout << "Correct number:   " << cNum << endl;
         }
       } else {
         cout << "Between 1 and 999, please."<<endl;
       }
       tries++;
    //output score
```

```cpp
        if(guess==answer){
        cout<<"Your score is:"<<endl;
        score[tries]=5000*1/tries;
        cout<<score[tries]<<endl;
        }
         }
    return tries;
}


void win(){
   cout<<" ** **  ****** ** **    **   ** *** **** **  *** "<<endl;
   cout<<" ** ** **  ** ** **    **   ** *** ***** **  ***"<<endl;
   cout<<"  **   ** ** ** **    ** * ** *** ** ** **  ***"<<endl;
   cout<<"  **   ** ** ** **    **** **** *** ** ** **    "<<endl;
   cout<<"  **   ****** *****    **   ** *** ** **** ***"<<endl;
}
void menu(void){
   //Output directions on how to play.
cout<<"        This is the game of Mastermind.           "<<endl;
cout<<"  Your objective is to guess the randomly generated code.  "<<endl;
cout<<"     The regular version will tell you if the numbers     "<<endl;
cout<<"  are in the correct position and are the right number.   "<<endl;
cout<<"The easier version will tell you if each digit is too high "<<endl;
cout<<"                or too low.                 "<<endl;
cout<<"~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"<<endl;
//output menu
cout<<"        Choose a level of difficulty.   "<<endl;
cout<<"          1 for the regular version.      "<<endl;
cout<<"      2 for an easier version of the game   "<<endl;
cout<<"              Any key to exit          "<<endl;

}
```