Game Project 2

<Mastermind Game>

CSC 5- 41202

Heidy Tamayo

Date: 02/11/16

**Introduction**

Title: Mastermind Game

Mastermind is played online where there is a table showing four empty slots allowing the user to enter a round marble color. The colors to choose from are red, blue, black, green, white, orange, brown and yellow. The computer generates its own colors and the user must guess which colors the computer has picked. The user types in the color in order from left to right and is then told if they have made a wrong guess. The is allowed ten turns to win the game but can use more than that if they like. If they have guessed wrong, the user is then asked if they would like a hint. If yes, the program gives a hint indicating if one of the colors is in the correct spot if not it goes on to the next try for the user to guess the colors. The user can keep guessing the colors the computer has generated up to the end of the number of tries the user inputted or if the user guesses all the colors correctly. If the user cannot guess the colors, the message displays they have lost the game. If the user has won, the percent accuracy is generated along with the number of tries it took the user to complete the game. At the end of the game whether win or lose, the computer asks the user if they would like to play again or end the program. Throughout the game there is a chart shown of each try the user has inputted.

**Summary**

Project size: 275 lines

Concepts Utilized:

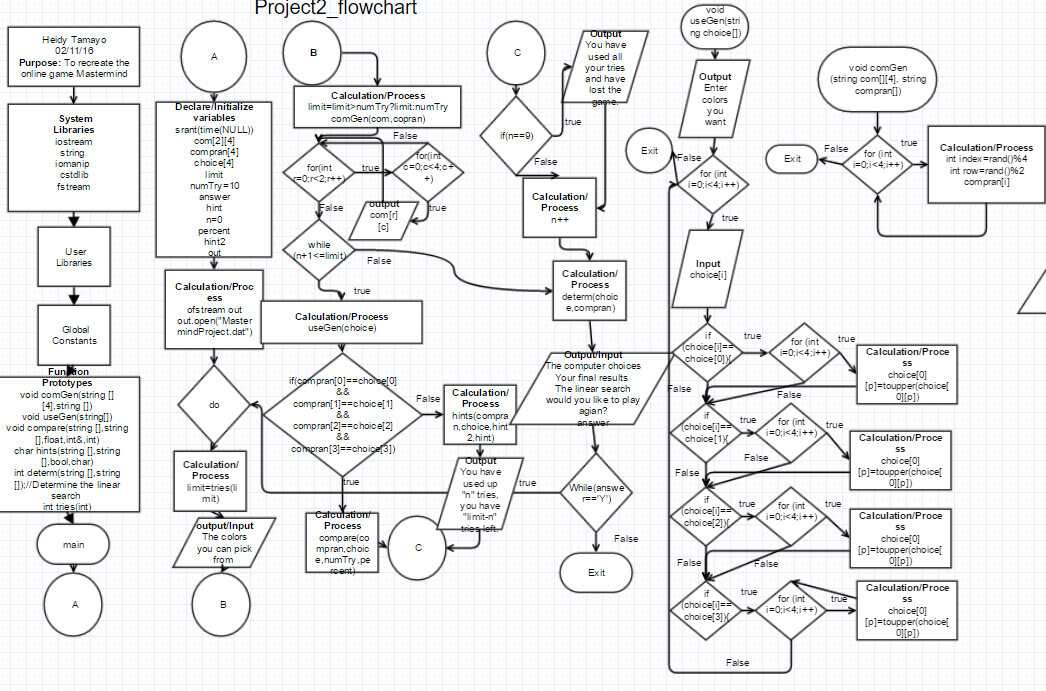
* Character data types
* Integer data types
* Float data types
* Boolean data types
* Ternary operators
* Single if-else statements
* Expanded if-else statements (if, else if, else)
* Formatting (setting the precision of a decimal)
* Single if
* Switch decision
* While loops
* Do-while loops
* For loops
* File (opening a file containing the outputted results)
* Random number seed
* pass by value calling a value to a function and then outputting it
* pass by reference where the & sign comes in the function
* defaulted parameters
* returning primitive data types
* outputted to file
* arrays
* parallel arrays
* searching required
* 2D arrays
* passing 2D arrays into/out of function

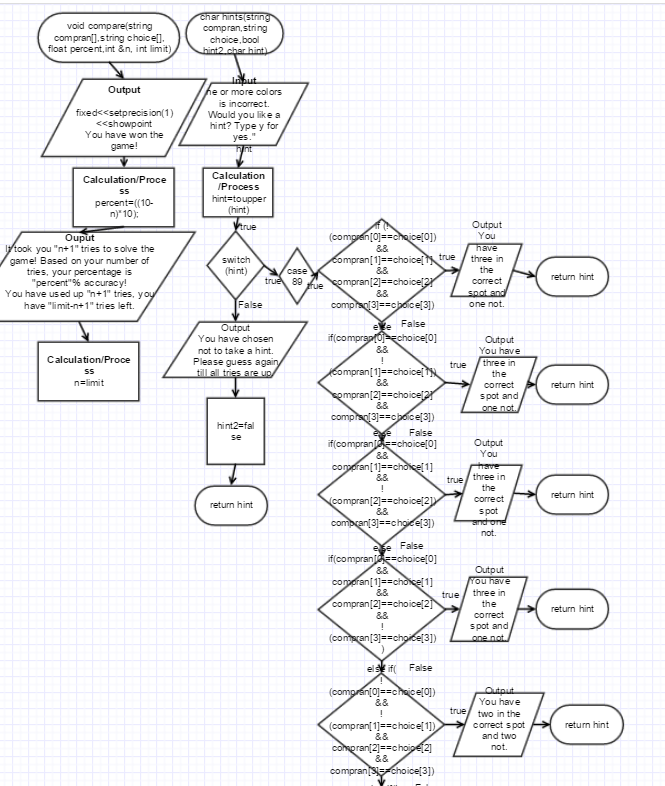
I tried to improve the game by showing the colors have used in the previous turns but had difficulty making a 2D array for the user inputs. I decided to leave it how it is. The hints stayed the same since I felt if I elaborated any more it would make it too easy for the user to find the correct colors. I hope I can improve this game by making the 2D without having errors. The use of functions made the program flow easier and have an understanding what was going in each section.

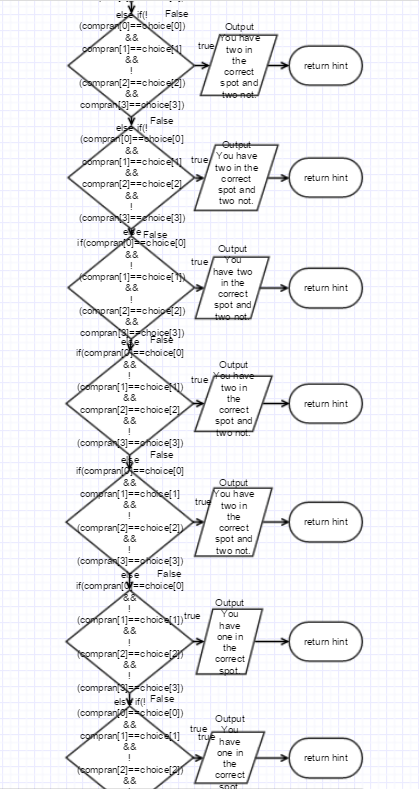
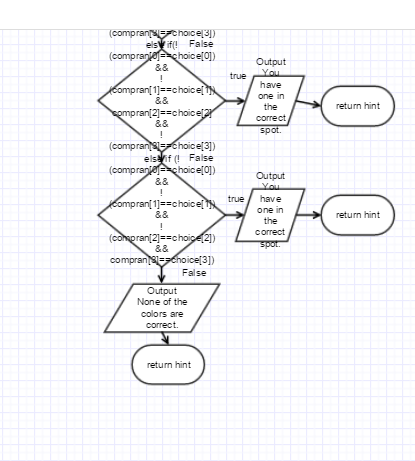
**Description**

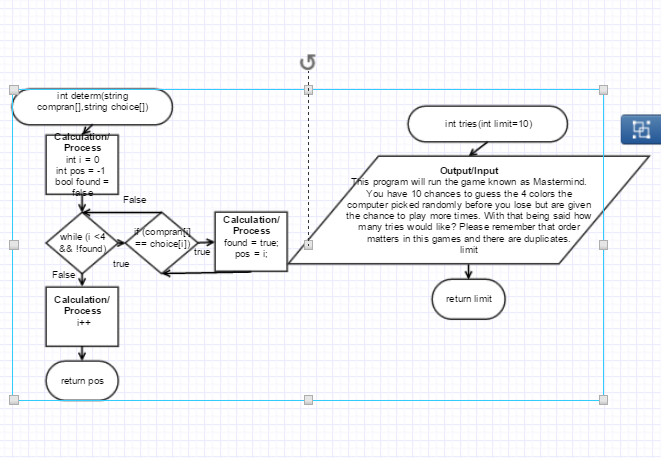
The main point of the program is to have the user think through the game based on the hints the computer generates and allow the user to guess the colors the computer has generated randomly.

**Flowchart**







**Pseudo Code**

Declaring and initialization of variables

Introduction to the game

Ask user for the number of tries

Displays which colors to pick from

Computer generates random colors

Initiate the number of turns the user has inputted

Ask for the user for their color choices

Comparing the users color to the computers

If they are all correct they win

Say their correct percentage and the number of tries they took and have left

If they are not, asks user for hints

Say how many tries they have spent and have left

After ten tries indicates the user has lost

Repeats until the user input of tries end

Determine the linear search of the computer generated and user

Shows the computer and the users results

Asks if the user would like to play again

End

**Major Variables**

|  |  |  |
| --- | --- | --- |
| **Type** | **Variable Name** | **Description** |
| int | limit | The limit the user inputs |
| string | Compran[4] | An array of 4 of the computer pick |
| string | Choice[4] | An array of 4 of the choice of the student |
| string | Com[2][4] | The 8 choice the computer picks from randomly |
| Int | NumTry=10 | The number of tries the user has before they lose the game |
| char | answer | The answer the user inputs |
| Char | hint | Determines if the user will like a hint |
| int | n=0 | Increments the while loop goes through |
| Float | Percent | The percentage the user gets if they win the game |
| Bool | Hint2 | Statement to determine if true and allow the output |
| Ofstream | Out | File |

**Constructs**

|  |  |  |
| --- | --- | --- |
| **Chapter** | **Type** | **Location(line)** |
| 2 | Char | 41  102  104  196  203 |
|  | int | 39  40  42  70  114 |
|  | Float | 43  186 |
|  | Bool | 44  243 |
|  | Ternary operator | 65 |
| 4 | if | 87  147-174 |
|  | If else statement | 75-85 |
|  | If else if | 208-236 |
|  | switch | 203 |
| 5 | while | 70 |
|  | Do-while | 51  104 |
|  | file | 45  48  107 |
|  | random | 32 |
| 6 | Pass by value | 54  94  114  259 |
|  | Pass by reference | 196  259 |
|  | Defaulted parameter | 114 |
|  | Returning primitive data types | 181 |
|  | functions | 21-26  54  65  67  73  82  94 |
| 7 | Parallel Arrays | 37  38  75 |
|  | Arrays | 37  38 |
|  | 2D Arrays | 35 |
| 8 | Searching | 259 |

**Program**

/\*

\* File: main.cpp

\* Author: Heidy Tamayo

\* Created on January 29, 2016, 5:15 PM

\* Purpose:

\*/

//System Libraries

#include <iostream> //I/O

#include <string> //string

#include <iomanip> //Formatting

#include <cstdlib> //srand and rand function

#include <fstream> //File I/O

using namespace std;

//User Libraries

//Global Constants

//Functional Prototypes

void comGen(string [][4],string []);//Function to generate the 4 random colors from the computer

void useGen(string[]);//Function to allow the user to enter their four colors

void compare(string [],string [],float,int&,int);//Comparison of the computer generated and the users

char hints(string [],string [],bool,char);//hints the user may have if they choose to

int determ(string [],string []);//Determine the linear search

int tries(int);//The number of tries the user would want

//Execution Begins Here

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

//Set the random number seed

srand(time(NULL));

//Declare and initialize variable

string com[2][4]={{"RED","BLUE","GREEN","BROWN"},

{"WHITE","BLACK","ORANGE","YELLOW"}};//Array that the computer picks from

string compran[4]; //The 4 colors the computer generates

string choice[4];//The 4 colors the user chooses

int limit;//The limit the user inputs

int numTry=10;//The number of tries the user gets before it is considered they have lost.

char answer, hint;//The response of whether the user would like to play again or take a hint

int n=0;//The increments to indicate the turns allowed before the user loses

float percent;//The intervals of the turns, the percentage of the accuracy

bool hint2=true;//Statement to determine if the statement is true

ofstream out;//File

//Open the file

out.open("MastermindProject.dat");

//Do while loop to see if the user would like to play again

do

{

//Output description

limit=tries(limit);

//Color choices avaliable

cout<<"The colors you can pick from are"<<endl;

for(int r=0;r<2;r++)

{

for(int c=0;c<4;c++)

{

cout<<com[r][c]<<" ";

}

}

//function to determine the limit

limit=limit>numTry?limit:numTry;

//Determining colors by the computer

comGen(com,compran);

//while loop to generate the number tries for the user

while (n+1<=limit)

{

//The color the user wants to pick

useGen(choice);

//if else statement to determine if the user has won or not

if(compran[0]==choice[0]&&compran[1]==choice[1]&&compran[2]==choice[2]&&compran[3]==choice[3]){

{

compare(compran,choice,percent,n,limit);

}

}

else

{

hints(compran,choice,hint2,hint);

cout<<endl<<"You have used up "<<n+1<<" tries, you have "<<10-(n+1)<<" tries left before it is considered you have lost the game."<<endl;

cout<<"You do however have "<<limit-(n+1)<<" tries left."<<endl;

}

//if statement to indicate the user has used up the ten turns they had

if (n==9)

{

//Output of results after all tries have been used up

cout<<"You have used all your tries and have lost the game."<<endl<<endl;

}

n++;

}

determ(choice,compran);

//Output of results

cout<<"The computer choices were "<<compran[0]<<" "<<compran[1]<<" "<<compran[2]<<" "<<compran[3]<<endl;

cout<<"Your final results were "<<choice[0]<<" "<<choice[1]<<" "<<choice[2]<<" "<<choice[3]<<endl;

cout<<"The linear search was "<<determ<<"."<<endl;

cout<<endl<<"Would you like to play again?"<<endl;

cin>>answer;

cout<<endl;

answer=toupper(answer);

n=1;

}while(answer=='Y');

//Exit stage right

out.close();

return 0;

}

//000000011111111112222222222333333333344444444445555555555666666666677777777778

//345678901234567890123456789012345678901234567890123456789012345678901234567890

// Allows the user to enter the number of tries

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int tries(int limit=10)

{

cout<<"This program will run the game known as Mastermind."<<endl;

cout<<"You have 10 chances to guess the 4 colors the computer picked randomly before you lose"<<endl;

cout<<"but are given the chance to play more times. With that being said how many tries would"<<endl;

cout<<"like? Please remember that order matters in this games and there are duplicates."<<endl;

cin>>limit;

return limit;

}

//000000011111111112222222222333333333344444444445555555555666666666677777777778

//345678901234567890123456789012345678901234567890123456789012345678901234567890

// Generates the computer's random values

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void comGen (string com[][4], string compran[])

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

{

int index=rand()%4;

int row=rand()%2;

compran[i]=com[row][index];

}

}

//000000011111111112222222222333333333344444444445555555555666666666677777777778

//345678901234567890123456789012345678901234567890123456789012345678901234567890

// Gets the four choices the user inputs

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void useGen(string choice[])

{

cout<<endl<<"Please pick your colors you would want displayed going from"<<endl;

cout<<"left to right."<<endl;

//Output request the user to enter the 4 colors

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

{

cin>>choice[i];

if (choice[i]==choice[0])

{

for (int p=0;p<choice[0].size();p++)

{

choice[0][p]=toupper(choice[0][p]);

}

}

if (choice[i]==choice[1])

{

for (int p=0;p<choice[1].size();p++)

{

choice[1][p]=toupper(choice[1][p]);

}

}

if (choice[i]==choice[2])

{

for (int p=0;p<choice[2].size();p++)

{

choice[2][p]=toupper(choice[2][p]);

}

}

if (choice[i]==choice[3])

{

for (int p=0;p<choice[3].size();p++)

{

choice[3][p]=toupper(choice[3][p]);

}

}

}

}

//000000001111111112222222222333333333344444444445555555555666666666677777777778

//345678901234567890123456789012345678901234567890123456789012345678901234567890

/\* Results if the user has won the game \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void compare(string compran[],string choice[], float percent,int &n, int limit)

{

//Output of results

cout<<fixed<<setprecision(1)<<showpoint;

cout<<endl<<"You have won the game!"<<endl;

percent=((10-n)\*10);

cout<<"It took you "<<n+1<<" tries to solve the game! Based on your number of tries, your";

cout<<" percentage is "<<percent<<"% accuracy!"<<endl;

cout<<endl<<"You have used up "<<n+1<<" tries, you have "<<limit-n+1<<" tries left."<<endl<<endl;

n=limit;

}

//000000001111111112222222222333333333344444444445555555555666666666677777777778

//345678901234567890123456789012345678901234567890123456789012345678901234567890

/\* Results if the user has not won \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

char hints(string compran[],string choice[],bool hint2,char hint)

{

cout<<"One or more colors is incorrect. Would you like a hint? Type y for yes."<<endl;

cin>>hint;

hint=toupper(hint);

//Switch statement to determine if the user would like a hint

switch (hint)

{

case 89:

{

//else if statements to determine which hint will be outputted

if (!(compran[0]==choice[0])&&compran[1]==choice[1]&&compran[2]==choice[2]&&compran[3]==choice[3]){

cout<<"You have three in the correct spot and one not."<<endl;}

else if(compran[0]==choice[0]&&!(compran[1]==choice[1])&&compran[2]==choice[2]&&compran[3]==choice[3]){

cout<<"You have three in the correct spot and one not."<<endl;}

else if(compran[0]==choice[0]&&compran[1]==choice[1]&&!(compran[2]==choice[2])&&compran[3]==choice[3]){

cout<<"You have three in the correct spot and one not."<<endl;}

else if(compran[0]==choice[0]&&compran[1]==choice[1]&&compran[2]==choice[2]&&!(compran[3]==choice[3])){

cout<<"You have three in the correct spot and one not."<<endl;}

else if(!(compran[0]==choice[0])&&!(compran[1]==choice[1])&&compran[2]==choice[2]&&compran[3]==choice[3]){

cout<<"You have two in the correct spot and two not."<<endl;}

else if(!(compran[0]==choice[0])&&compran[1]==choice[1]&&!(compran[2]==choice[2])&&compran[3]==choice[3]){

cout<<"You have two in the correct spot and two not."<<endl;}

else if(!(compran[0]==choice[0])&&compran[1]==choice[1]&&compran[2]==choice[2]&&!(compran[3]==choice[3])){

cout<<"You have two in the correct spot and two not."<<endl;}

else if(compran[0]==choice[0]&&!(compran[1]==choice[1])&&!(compran[2]==choice[2])&&compran[3]==choice[3]){

cout<<"You have two in the correct spot and two not."<<endl;}

else if(compran[0]==choice[0]&&!(compran[1]==choice[1])&&compran[2]==choice[2]&&!(compran[3]==choice[3])){

cout<<"You have two in the correct spot and two not."<<endl;}

else if(compran[0]==choice[0]&&compran[1]==choice[1]&&!(compran[2]==choice[2])&&!(compran[3]==choice[3])){

cout<<"You have two in the correct spot and two not."<<endl;}

else if(compran[0]==choice[0]&&!(compran[1]==choice[1])&&!(compran[2]==choice[2])&&!(compran[3]==choice[3])){

cout<<"You have one in the correct spot."<<endl;}

else if(!(compran[0]==choice[0])&&compran[1]==choice[1]&&!(compran[2]==choice[2])&&!(compran[3]==choice[3])){

cout<<"You have one in the correct spot."<<endl;}

else if(!(compran[0]==choice[0])&&!(compran[1]==choice[1])&&compran[2]==choice[2]&&!(compran[3]==choice[3])){

cout<<"You have one in the correct spot."<<endl;}

else if (!(compran[0]==choice[0])&&!(compran[1]==choice[1])&&!(compran[2]==choice[2])&&compran[3]==choice[3]){

cout<<"You have one in the correct spot."<<endl;}

else{

cout<<"None of the colors are correct."<<endl;}

break;

}

default:

{

cout<<"You have chosen not to take a hint. Please guess again till your number of tries is up."<<endl;

hint2=false;

}

}

return hint;

}

//000000011111111112222222222333333333344444444445555555555666666666677777777778

//345678901234567890123456789012345678901234567890123456789012345678901234567890

// Linear Search

//Inputs:

// a->Array or List

// n->Size of List

// strt->Starting Position

// val->Value to find

//Outputs:

// pos->Index where value is found

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int determ(string compran[],string choice[])

{

int i = 0;

int pos = -1;

bool found = false;

while (i <4 && !found)

{

if (compran[i] == choice[i])

{

found = true;

pos = i;

}

i++;

}

return pos;

}