Project 1

Mastermind

CSC-42645

Name: Matthew Yang

Date 4/19/17

Introduction

Title: MasterMind

This is the Mastermind game.

For this Mastermind game, the computer will select 4 repeatable numbers 0-9, and you must guess which numbers they are, and what order those numbers are in within 15 turns.

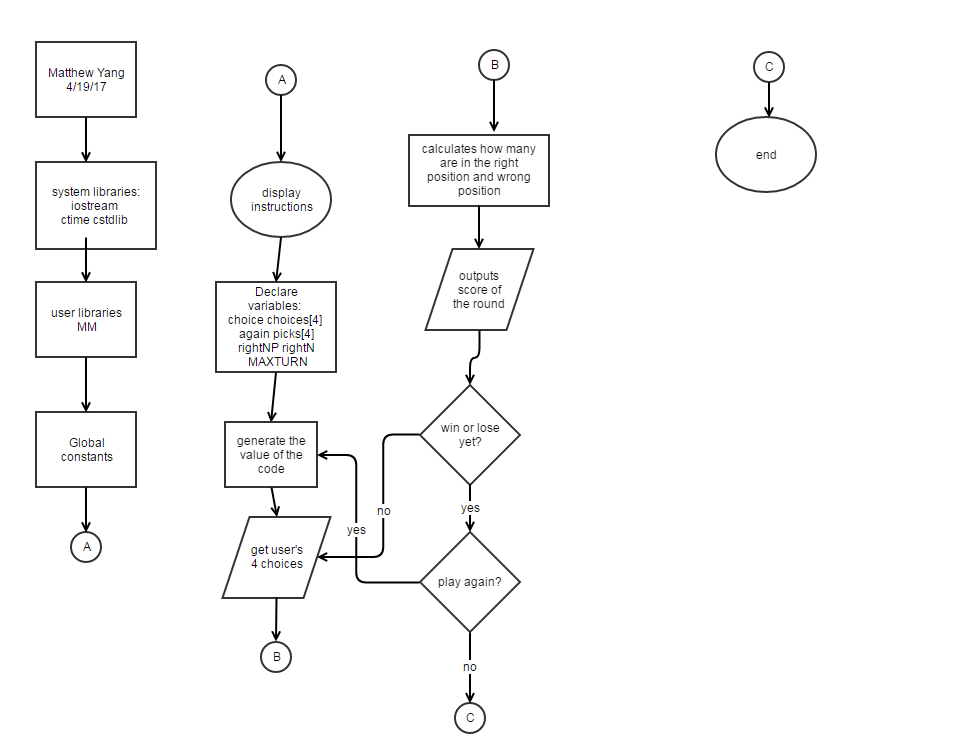
Summary

Project size: 144 lines

Variables: there are eight main variables that are in the declare variables position.

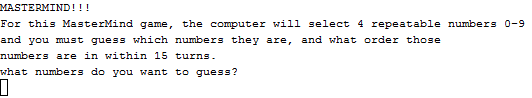
This game was fun for me to make because it was a game that I liked to play. I could have made the code more efficient in many ways, but did not have enough time.

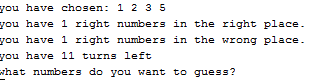
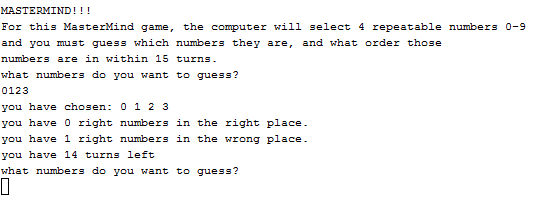
Flowchart



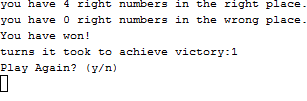
Samples

This is the introduction of my program. You are asked to choose what which 4 numbers you would like to input.





These show you what happens if you get a right number in the right place or wrong place and the turn counter.



This shows you what happens when you win. You are presented with an option to play the game again with a new code to break after being congratulated.

Pseudocode

*makes random code for user to break*

*//receives the user's choice numbers*

*invalidates numbers above 4 digits*

*separates the sequence of numbers that the user inputted*

*calculates the amount of right numbers that are in the right place*

*calculates the amount of right numbers in the wrong place*

*outputs score of the turn*

*asks if you would like to play again if you won*

*asks if you would like to try again if you lost*

Program

int choice=0;

int choices[4];

srand(time(0));

bool again=true; //play again?

MM menu;

int picks[4];

int rightNP=0; //right number and place

int rightN=0; //right number but not place

const int MAXTURN=15; //max turn value

char buffer;

cout<<"MASTERMIND!!!"<<endl;

cout <<"For this MasterMind game, the computer will select 4 repeatable numbers 0-9 "<<endl<<

"and you must guess which numbers they are, and what order those"<<endl<<

"numbers are in within 15 turns."<<endl;

do{

for(int i=0;i<=3;i++){ //makes random code for user to break

picks[i]=menu.Rand();

}

for(int turnC=1;turnC<=MAXTURN;turnC++){

// for(int i=0;i<=3;i++){ //sets code to display the code that the game chose

// cout<<picks[i];

// }

// cout<<endl;

cout << "what numbers do you want to guess?"<<endl; //receives the user's choice numbers

cin>>choice;

while (choice>9999 ){ //invalidates numbers above 4 digits

cout << "please input 4 numbers only"<<endl;

cin>>choice;

}

choices[3]=choice%10; //separates the sequence of numbers that the user inputted

choices[2]=((choice%100-choices[3]))/10;

choices[1]=(choice%1000-(choices[3]+choices[2]))/100;

choices[0]=(choice-(choices[3]+choices[2]+choices[1]))/1000;

cout <<"you have chosen: "<<choices[0]<<" "<<choices[1]<<" "<<choices[2]<<" "<<choices[3]<<endl;

if(choices[0]==picks[0]){rightNP++;} //calculates the amount of right numbers that are in the right place

if(choices[1]==picks[1]){rightNP++;}

if(choices[2]==picks[2]){rightNP++;}

if(choices[3]==picks[3]){rightNP++;}

for(int i=0;i<=3;i++){

for(int j=0;j<4;j++){ //calculates the amount of right numbers in the wrong place

if (choices[i]==picks[j]){rightN++;break;}

}

}

rightN-=rightNP;

cout <<"you have "<<rightNP<<" right numbers in the right place."<<endl; //outputs score of the turn

cout <<"you have "<<rightN<<" right numbers in the wrong place."<<endl;

if (rightNP<4){

cout <<"you have "<<MAXTURN-turnC<<" turns left"<<endl;

}

if (rightNP==4){ //asks if you would like to play again if you won

cout << "You have won!"<<endl<<

"turns it took to achieve victory:"<<turnC<<endl<<

"Play Again? (y/n)"<<endl;

cin >>buffer;

if (buffer=='y'){

turnC=MAXTURN;

}

if (buffer=='n'){

again=false;

turnC=MAXTURN;

}

else{

turnC=MAXTURN;

}

}

if (rightNP!=4 && turnC==MAXTURN){ //asks if you would like to try again if you lost

cout <<"you lose."<<endl<<

"the answer was:"<<picks[0]<<picks[1]<<picks[2]<<picks[3]<<endl<<

"Play Again? (y/n)"<<endl;

cin >>buffer;

if (buffer=='y'){

turnC=MAXTURN;

}

if (buffer=='n'){

again=false;

turnC=MAXTURN;

}

else{

turnC=MAXTURN;

}

}

rightNP=0;

rightN=0;

}

}while (again==true);

return 0;