Project 1

The Card Matching Game

CIS-42829

Name: Matthew Yang

Date 5/4/16

Introduction

Title: Matching Card Game

This is a card matching game.

The cards are set before you, and the user needs to input two cards one at a time in order to reveal what number is underneath those two cards. The goal of the game is to pick two cards that have the same number underneath them.

When all numbers are matched successfully, the game is over.

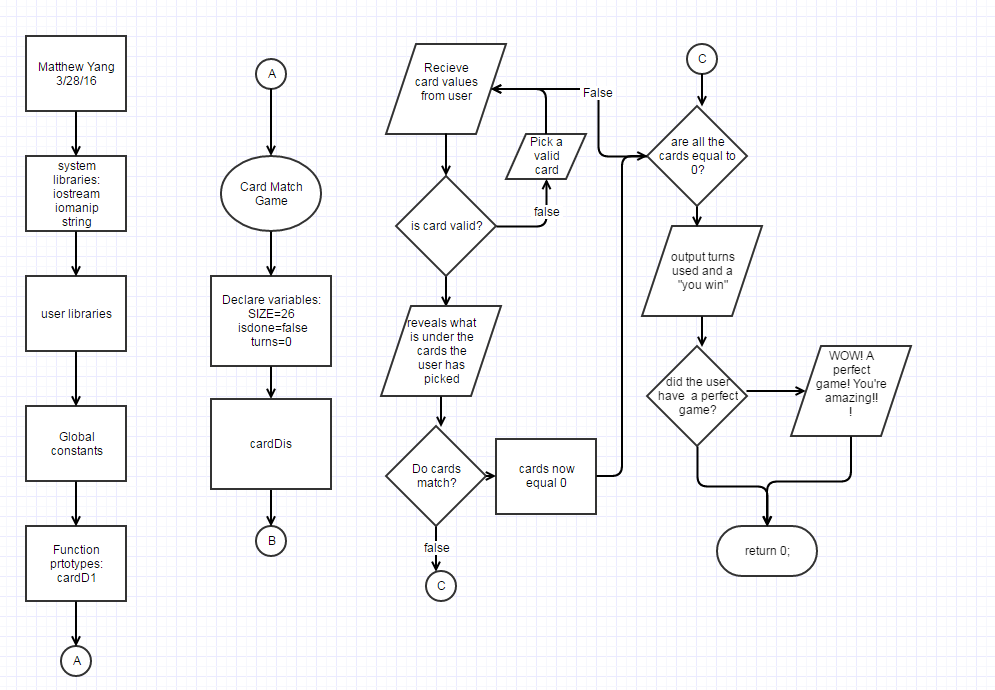
This is a fun family game that many play to see how fast they can finish the game in the least amount of turns.

Summary

Project size: 139 lines

Variables: there are 5 variables that aren’t in an array, and there are 2 arrays with one having 26 variables, and the other containing 13 variables. Therefore, there are 44 variables in all.

This project contains most concepts we have learned in class, I could have displayed the cards each turn with them disappearing if they were eliminated, but couldn’t figure out how to do so, so I gave up trying to do that. And I could have added points if I figured out a clever way to do that as well.

Flowchart

Major variables

Type Variable Name Description Location

|  |  |  |  |
| --- | --- | --- | --- |
| integer | SIZE | Number of cards (26) | In declared variables |
|  | cadpik1 | User input to choose first card to reveal | When program asks the user for the cards to pick (near the beginning) |
|  | cadpik2 | User input to choose second card to reveal | When program asks the user for the cards to pick (near the beginning) |
|  | turns | How many turns has passed | After cards that the user has inputted have been verified |
| Integer arrays | cards[ ] | Value in each card 1-13 | After the declaration of variables, and when the program reads the revealed values back to the user |
|  | Freqofc[ ] | Used to make sure each card is picked only twice | After the declaration of variables |
| bool | isDone | Determines when to finish the game | At the end of main, inside the last for-loop |

Constructs

Chapter Keywords Location

|  |  |  |
| --- | --- | --- |
| Chapter 2 | * Data types: int, bool, string constants | * (in the major variables page) |
| Chapter 3 | * Iomanip: setw( ) * String | * In cardD1( ) function * In declaring variables section |
| Chapter 4 | * If statements * Switch statement | * Many of them everywhere * At the very end of main |
| Chapter 5 | * For loops * Do while loop * Increments | * After the declaring variables * After the first for loop * In every for loop and after verifying user inputs |
| Chapter 6 | * Function | * Before the computer asks for the user to input a card |
| Chapter 7 | * Arrays | * (in the major variables page) |

Program

//declare variables

bool isDone = false; //determines when game is done

const unsigned int SIZE =26; //total #of cards

int cards[SIZE] = {0}; //total card's values

int freqofC[13] = {0}; //frequency of cards

int cadpik1; //card picks 1

int cadpik2; //card picks 1

unsigned int turns=0; //turns used

//creates randomness

unsigned seed = time(0);

srand(seed);

//create values for each card, while making sure each card is repeated only twice

for(int i=0; i<26; ++i) {

int num = rand() % 13 + 1;

while (freqofC[num-1]>=2) { //makes sure that no card has the

num = rand() % 13 + 1; //same value more than twice.

}

++freqofC[num-1];

cards[i] = num;

}

do{

//displays cards

cardD1();

//pick card 1, and makes sure it is a valid number.

cout << "pick one card."<<endl;

cin >> cadpik1;

if (cadpik1<1||cadpik1>26) {cout << "you must pick a number greater than 0,"

"and less than 26"<<endl;

cin >>cadpik1;

}

//pick card 2, and make sure it is a valid number.

cout << "pick Your second Card."<<endl;

cin >> cadpik2;

if (cadpik2<1||cadpik2>26) {cout << "you must pick a number greater than 0,"

"and less than 26"<<endl;

cin >>cadpik2;

}

//makes sure that the cards you inputted are different.

while (cadpik1==cadpik2){

cout << "you must pick a number that is different than the first "<<endl

<<"card you have picked. Pick another card for your second card."

<<endl;

cin >>cadpik2;

}

//outputs the results of the cards

cout << endl << "The first Card you picked has a "<<cards[cadpik1-1] <<

" under it." <<endl;

cout << "The second Card you picked has a "<<cards[cadpik2-1] <<

" under it." <<endl<<endl;

turns++;

cout << "this is turn number "<<turns<< endl;

//if the cards are of the same number, they are eliminated

if (cards[cadpik1-1]==cards[cadpik2-1]){ cout <<cadpik1<< " and "

<<cadpik2 <<" are eliminated."<< endl<<endl;

cards[cadpik1-1]=0;

cards[cadpik2-1]=0;

}

int cnt=0; //counts the number of eliminated cards

for(int i=0; i<26; ++i)

if( cards[i]==0 ) ++cnt;

if(cnt==26) isDone = true;

// # of terminated cards

cout << "You have eliminated " << cnt << " cards out of 26." << endl;

}while(!isDone); //finishes the program

cout << "You have Won! you have completed the game in " <<turns<< "turns.";

switch (turns){

case 13: cout << "WOW! A perfect game! You're amazing!!";

default: cout << "";

}

return 0;

}

//displays the introduction output

void cardD1() {

string cad1P1="\_\_\_", cad1P2= "| 1 |",cad1P3= "|\_\_\_|",cad2P1= "\_\_\_",cad2P2= "| 2 |",cad2P3= "|\_\_\_|",

cad3P1="\_\_\_", cad3P2= "| 3 |",cad3P3= "|\_\_\_|",cad4P1= "\_\_\_",cad4P2= "| 4 |",cad4P3= "|\_\_\_|",

cad5P1="\_\_\_", cad5P2= "| 5 |",cad5P3= "|\_\_\_|",cad6P1= "\_\_\_",cad6P2= "| 6 |",cad6P3= "|\_\_\_|",

cad7P1="\_\_\_", cad7P2= "| 7 |",cad7P3= "|\_\_\_|",cad8P1= "\_\_\_",cad8P2= "| 8 |",cad8P3= "|\_\_\_|",

cad9P1="\_\_\_", cad9P2= "| 9 |",cad9P3= "|\_\_\_|",cad10P1="\_\_\_",cad10P2="|10 |",cad10P3="|\_\_\_|",

cad11P1="\_\_\_", cad11P2="|11 |",cad11P3= "|\_\_\_|",cad12P1="\_\_\_",cad12P2="|12 |",cad12P3="|\_\_\_|",

cad13P1="\_\_\_", cad13P2="|13 |",cad13P3= "|\_\_\_|",cad14P1="\_\_\_",cad14P2="|14 |",cad14P3="|\_\_\_|",

cad15P1="\_\_\_", cad15P2="|15 |",cad15P3= "|\_\_\_|",cad16P1="\_\_\_",cad16P2="|16 |",cad16P3="|\_\_\_|",

cad17P1="\_\_\_", cad17P2="|17 |",cad17P3= "|\_\_\_|",cad18P1="\_\_\_",cad18P2="|18 |",cad18P3="|\_\_\_|",

cad19P1="\_\_\_", cad19P2="|19 |",cad19P3= "|\_\_\_|",cad20P1="\_\_\_",cad20P2="|20 |",cad20P3="|\_\_\_|",

cad21P1="\_\_\_", cad21P2="|21 |",cad21P3= "|\_\_\_|",cad22P1="\_\_\_",cad22P2="|22 |",cad22P3="|\_\_\_|",

cad23P1="\_\_\_", cad23P2="|23 |",cad23P3= "|\_\_\_|",cad24P1="\_\_\_",cad24P2="|24 |",cad24P3="|\_\_\_|",

cad25P1="\_\_\_", cad25P2="|25 |",cad25P3= "|\_\_\_|",cad26P1="\_\_\_",cad26P2="|26 |",cad26P3="|\_\_\_|";

cout << " "<<cad1P1<<setw(6)<<cad2P1<<setw(6)<<cad3P1<<setw(6)<<cad4P1<<setw(6)<<cad5P1<<setw(6)<<cad6P1<<endl;

cout <<cad1P2<<" "<<cad2P2<<" "<<cad3P2<<" "<<cad4P2<<" "<<cad5P2<<" "<<cad6P2<<endl;

cout <<cad1P3<<" "<<cad2P3<<" "<<cad3P3<<" "<<cad4P3<<" "<<cad5P3<<" "<<cad6P3<<endl;

cout << " "<<cad7P1<<setw(6)<<cad8P1<<setw(6)<<cad9P1<<setw(6)<<cad10P1<<setw(6)<<cad11P1<<setw(6)<<cad12P1<<endl;

cout <<cad7P2<<" "<<cad8P2<<" "<<cad9P2<<" "<<cad10P2<<" "<<cad11P2<<" "<<cad12P2<<endl;

cout <<cad7P3<<" "<<cad8P3<<" "<<cad9P3<<" "<<cad10P3<<" "<<cad11P3<<" "<<cad12P3<<endl;

cout << " "<<cad13P1<<setw(6)<<cad14P1<<setw(6)<<cad15P1<<setw(6)<<cad16P1<<setw(6)<<cad17P1<<setw(6)<<cad18P1<<endl;

cout <<cad13P2<<" "<<cad14P2<<" "<<cad15P2<<" "<<cad16P2<<" "<<cad17P2<<" "<<cad18P2<<endl;

cout <<cad13P3<<" "<<cad14P3<<" "<<cad15P3<<" "<<cad16P3<<" "<<cad17P3<<" "<<cad18P3<<endl;

cout << " "<<cad19P1<<setw(6)<<cad20P1<<setw(6)<<cad21P1<<setw(6)<<cad22P1<<setw(6)<<cad23P1<<setw(6)<<cad24P1<<endl;

cout <<cad19P2<<" "<<cad20P2<<" "<<cad21P2<<" "<<cad22P2<<" "<<cad23P2<<" "<<cad24P2<<endl;

cout <<cad19P3<<" "<<cad20P3<<" "<<cad21P3<<" "<<cad22P3<<" "<<cad23P3<<" "<<cad24P3<<endl;

cout << " "<<cad25P1<<setw(6)<<cad26P1<<endl;

cout <<cad25P2<<" "<<cad26P2<<endl;

cout <<cad25P3<<" "<<cad26P3<<endl;

}