Wheel of

Fortune  
Project 2

CSC-5 Intro C++

Sergio R. Montalvan

Summer 2014

Table of Contents

Table of Contents………………………………………………………2

Rules of Game….………………………………………………………3

Reason for Choosing....…………………………………………………3

Issues with Project……………………………………………………3-4

Adjustments……..………………………………………………………4

Pseudo-code…….……………………………………………………..4-7

Code…………………………………………………………………7-15

Rules of the Game

The rules of this game are to spin the wheel to win the money that the wheel spun on. After getting the amount on the wheel added to the players tab they are told to answer the following question, “What word is it?” and they must either buy a type of letter to figure out the word. Vowels cost $300 while a Consonant costs $200 if the player has enough information on the board or feels lucky they can enter the full word and try to win the money immediately without losing money but only a turn. In a sense Wheel of Fortune is just a more complicated, money-driven version of hangman.

Reason for Choosing

The reason I chose this game was due to the time constraint from the first one and it was a work in process. The idea came when we were told to do a game for the first project. I started a little before it was announced but did not finish it due to the little time given to finish the first project. With half of the project already done I started to try to complete it when the first project was supposed to be turned in. With a little help of some classmates and the SI I got the majority of the game completed with a few bugs here and there.

Issues with Project

The issue with this project is that it’s a little screwy. The game sometimes skips the logic or statements put inside it but runs the initial steps and loops to continue the game. It also stores the money so that is how I know that the game keeps looping. The other issue with it is that it is not fully completed since some void functions I used before were not working and gave me some trouble latter on in the project, thus requiring me to take them out or redo them.

Adjustments

At first, the project was going very smoothly but when I tried to a specific void function for the process in which the player was given choices to choose to buy a vowel, consonant…etc… it was backfiring and did not allow me to do so without making the code look very horrendous so I took it out and tried to figure out where to implant it and if they would work without being in the code as a void. So it did but then a new problem arose when the void was implemented inside the code itself instead of being a void function. Thus time ran out and those little problems stayed inside the code but if more time was given it may have been possible to add more players and stabilize the game.

Pseudo-Code

Comment with all Information

//System Level Libraries

//User Libraries

//Global Constants

Structure of Player One

Array Set up for Spins

Array Set up for Foods Chosen

//Function Prototypes

Press Enter for Wheel Spin

Press Enter to continue

Player's choice

//Execution Starts Here...

Setting up Random Wheel

Declaring Variables

Player choices

Money off of Wheel for inside player choices as well

Setting up The Wheel of Fortune for inside of player choices

User's Input

User Entered Full Word

User Input to End Game

User Wins

The amount of tries/guesses

To input a file

If Input is Entered Wrong

correct

similar

//Opening Files

Open Food File

Open Intro File

Opening of the game with Welcome

Enter Player One’s Name

Shows Player One’s Current Balance

Queue’s Enter to Continue

Queue’s Enter to Spin Wheel

Spins Wheel

Add’s Number on Wheel to Player’s Balance

Queue’s Continue for Word

The Loop for Entirety of Game Begins

//The word/phrase generator

Set up Random Generator

Random from 1-30 choices

chooses a word randomly

The temporary word save

The game word

The Guess

Loop for the game word

Game Reset

Wins, Tries are set to 0

Word Similarity is false

Game Loop

{

Correct is True

Queue’s Enter to Get word

Shows word as underscores

Shows word To test if underscores and majority of the code work

Recovery and Display

Switch statement to give player choices

Buy a Vowel

If Player One’s Money is under 0

Player is Bankrupt and ask to end game

Buy a Consonant

If Player One’s Money is under 0

Player is Bankrupt and ask to end game

Choose to Spin Again!

Choose to Answer Full Word

If Answered correctly

Player Wins and asks to continue game

If Answer if incorrect

Player loses a guess but no money penalty

Checks If input is wrong and is only wrong if input is special character or number

Checks if input is right and only if input is a letter

If something more than 1 character or invalid input

And Player loses a guess

End of First Loop

To check if letter inputted matches a piece of the word

Recovery and Display

Continue after all the first initial checks

Switch statement to give player choices

Buy a Vowel

If Player One’s Money is under 0

Player is Bankrupt and ask to end game

Buy a Consonant

If Player One’s Money is under 0

Player is Bankrupt and ask to end game

Choose to Spin Again!

Choose to Answer Full Word

If Answered correctly

Player Wins and asks to continue game

If Answer if incorrect

Player loses a guess but no money penalty

If all Requirements are met Win is true, similarity is true and correct is true

Tell Player they have one and chance to keep playing or keep money

End Game if Entered Yes to End Game

End of Program

Void Function Stating to Press Enter to Spin

Void Function Stating to Press Enter to Continue

Code

/\*

\* File: main.cpp

\* Author: Sergio R Montalvan

\* Created on July 10, 2014, 12:24 PM

\* Project 1 CSC5 46023

\*/

//System Level Libraries

#include <cstdlib>

#include <iostream>

#include <string.h>

#include <iomanip>

#include <ctime>

#include <fstream>

using namespace std;

//User Libraries

//Global Constants

struct player

{

string name;

int money;

}one;

const int spin=24;//Setting up the Array for spin numbers

const int foods=30;//Setting up Array for number of foods

//Function Prototypes

void spinWoF();//Press Enter for Wheel Spin

void ReadyW();//Press Enter to continue

void PlayerC();//Player's choice

//Execution Starts Here...

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

//Setting up Random Wheel

srand(time(0));

unsigned short spins=rand()%24+1;

//Declaring Variables

int pchoice=4;//Player choices

int Wmon;//Money off of Wheel for inside player choices as well

//Setting up The Wheel of Fortune for inside of player choices

int WoF[spin]=

{

800,500,3500,600,300,900,2500,350,425,700,650,

575,325,525,675,850,5000,750,475,625,825,550,450,725

};

char letter;//User's Input

string fword;//User Entered Full Word

string endg="No";

int win=false;//User Wins

int tries=0;//The amount of tries/guesses

ifstream input;//To input a file

int wrong;

bool corr;//correct

bool simi;//similar

//Opening Files

input.open("food.txt");

string food[foods];

for(int i=0;i<30;i++)

input>>food[i];

input.close();

input.open("intro.txt");

string intro;

if (input.is\_open())

{

while ( getline (input,intro) )

{

cout<<intro<<'\n';

}

input.close();

}

//Opening of the game with Welcome

cout<<"Welcome to Wheel of Fortune! What is your name challenger?\n";

cin>>one.name;

cin.ignore();

one.money=0;

cout<<"Your current balance is $"<<one.money<<"."<<endl;

cout<<"Are You Ready to Spin the Wheel?!\n"

<<"(Press Enter to Continue)\n";

cin.ignore();

spinWoF();

cout<<"You Spun a "<<spins<<" with a value of"<<endl;

Wmon=WoF[spins];

cout<<"$"<<Wmon<<endl;

one.money+=Wmon;

cout<<"You have $"<<one.money<<endl;

//The Loop for Entirety of Game

do

{

//The word/phrase generator

srand(time(0));

unsigned short choice=rand()%30+1;

string word =(food[choice]);//chooses a word randomly

char tempw[word.length()];//The temporary word save

char gameword[word.length()];//The game word

char guess[word.length()];

for(int i=0;i<word.length();i++)//Loop for the game word

{

gameword[i]=word[i];

}

//Game Reset

win=false;

tries=0;

simi=false;

do //Game Loop

{

corr=true;

ReadyW();

cout<<"What Word is it?\n";

for(int i=0;i<word.length();i++)//Shows word as underscores

{

guess[i]='\_';

tempw[i]=guess[i];

}

cout<<endl;

// cout<<word<<endl; //to test if underscores and majority of the code work

for(int i=0;i<word.length();i++){

guess[i]=tempw[i];//Recovery and Display

cout<<guess[i]<<" ";

}

cout<<endl;

cout<<"Choose Accordingly!\n\n"//Switch statement to give player choices

<<"1.Buy a Vowel!\n"

<<"2.Buy a Consonant!\n"

<<"3.Spin the Wheel Again!\n"

<<"4.Guess the Word\n";

cin>>pchoice;

cin.ignore();

switch(pchoice)

{

case 1:

{

cout<<"You bought a Vowel!\n";

one.money-=300;

cout<<"You have $"<<one.money<<" left!\n";

cout<<"Please Enter a Vowel!\n";

cin>>letter;

cin.ignore();

tries++;

if(one.money<0)

{

cout<<"You are Bankrupt!\n";

cout<<"End Game? Yes/No?\n";

cin>>endg;

cin.ignore();

}

break;

}

case 2:

{

cout<<"You bought a Consonant!\n";

one.money-=200;

cout<<"You have $"<<one.money<<" left!\n";

cout<<"Please Enter a Consonant!\n";

cin>>letter;

cin.ignore();

tries++;

if(one.money<0)

{

cout<<"You are Bankrupt!\n";

cout<<"End Game? Yes/No?\n";

cin>>endg;

cin.ignore();

}

break;

}

case 3:

{

cout<<"You can spin the wheel again!\n";

srand(time(0));

unsigned short spins=rand()%24+1;

spinWoF();

cout<<"You Spun a "<<spins<<" with a value of"<<endl;

Wmon=WoF[spins];

cout<<"$"<<Wmon<<endl;

one.money+=Wmon;

cout<<"You have $"<<one.money<<endl;

break;

}

case 4:

{

cout<<"What do you believe the word is?\n";

cin>>fword;

cin.ignore();

for(int i=0;i<word.length();i++)

{

word[i]=fword[i];

tries++;

}

if(!strcmp(gameword,tempw))

{

win=true;

simi=true;

cout<<"Congrats that was the right word!\n";

}

if(simi==false)

{

tries++;

cout<<"Your Answer was not correct!\n";

}

break;

}

}

if ((letter>49 && letter<97) || letter>122 || letter<49)//if input is wrong

{

corr=false;

}

if(corr==true){

for(int i=0;i<word.length();i++)

{

if(letter==gameword[i]) //if input is right

{

wrong=1;

guess[i]=word[i];

tries++;

}

}

}

else

{

cout<<"Vowel or Consonant Please!\n";//If something more than 1 character or invalid input

tries++;

corr=false;

}

}while(corr==false);

if(letter!=49)//to check if letter inputted matches a piece of the word

{

for(int i=0;i<word.length();i++)

{

if(letter==gameword[i])

{

word[i]=gameword[i];

tries++;

simi=true;

}

tempw[i]=guess[i];

}

for(int i=0;i<word.length();i++){

guess[i]=tempw[i];//Recovery and Display

cout<<guess[i]<<" ";

}

if(simi==true && win==false)//Continue after all the first initial checks

{

cout<<"Choose Accordingly!\n\n"

<<"1.Buy a Vowel!\n"

<<"2.Buy a Consonant!\n"

<<"3.Spin the Wheel Again!\n"

<<"4.Guess the Word\n";

cin>>pchoice;

cin.ignore();

switch(pchoice)

{

case 1:

{

cout<<"You bought a Vowel!\n";

one.money-=300;

cout<<"You have $"<<one.money<<" left!\n";

cout<<"Please Enter a Vowel!\n";

cin>>letter;

cin.ignore();

tries++;

break;

if(one.money<0)

{

cout<<"You are Bankrupt!\n";

cout<<"End Game? Yes/No?\n";

cin>>endg;

cin.ignore();

}

}

case 2:

{

cout<<"You bought a Consonant!\n";

one.money-=200;

cout<<"You have $"<<one.money<<" left!\n";

cout<<"Please Enter a Consonant!\n";

cin>>letter;

cin.ignore();

tries++;

if(one.money<0)

{

cout<<"You are Bankrupt!\n";

cout<<"End Game? Yes/No?\n";

cin>>endg;

cin.ignore();

}

break;

}

case 3:

{

cout<<"You can spin the wheel again!\n";

srand(time(0));

unsigned short spins=rand()%24+1;

spinWoF();

cout<<"You Spun a "<<spins<<" with a value of"<<endl;

Wmon=WoF[spins];

cout<<"$"<<Wmon<<endl;

one.money+=Wmon;

cout<<"You have $"<<one.money<<endl;

break;

}

case 4:

{

cout<<"What do you believe the word is?\n";

cin>>fword;

cin.ignore();

for(int i=0;i<word.length();i++)

{

word[i]=fword[i];

tries++;

}

if(!strcmp(gameword,tempw))

{

win=true;

simi=true;

corr=true;

cout<<"Congrats that was the right word!\n";

}

if(simi==false)

{

tries++;

cout<<"Your Answer was not correct!\n";

}

break;

}

}

}

}

if(tries<10 && win==true && simi==true){

cout<<"Congrats on Winning!\n"

<<"You won $"<<one.money<<endl;

cout<<"Continue? Yes/No?\n";

cin>>endg;

cin.ignore();

}

} while((endg!="Yes")&&(endg!="yes"));

return 0;

}

void spinWoF()

{

cout<<"Press Enter to Spin the Wheel!\n";

cin.ignore();

}

void ReadyW()

{

cout<<"Are You Ready to Continue? Press Enter for your word?\n";

cin.ignore();

}