

Code Listing

By Adam Chu



#include <windows.h>



#include <iomanip>

using namespace std;



int main() {

Machine Sbeven;

cin.get();

return 0;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Machine.h

#include <string>

#include <vector>



#include "banker.h"



class Machine{

public:



Banker y;



Machine();



void attract\_Mode();



void spin\_Reels();



protected:

std::vector<std::string> reel\_1;

std::vector<std::string> reel\_2;

std::vector<std::string> reel\_3;



int vector\_Correct(int a);



void calcWin(std::string d, std::string e, std::string f);

int reel\_Act1, reel\_Act2, reel\_Act3, top\_Row1, top\_Row2, top\_Row3, top\_Row1\_, top\_Row2\_, top\_Row3\_;

int reel\_Act1\_, reel\_Act2\_, reel\_Act3\_, bot\_Row1, bot\_Row2, bot\_Row3, bot\_Row1\_, bot\_Row2\_, bot\_Row3\_;

std::string a, b, c, d, e, g, f, h, i, j;

};

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Banker.h

#pragma once



class Banker {

public:

Banker();

float curr, jPot;

float input\_Money(float x);

float output\_Money(float x);

int pound, ten\_pence, twenty\_pence, fifty\_pence, x;

};

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Machine.cpp

#pragma once

#include "banker.h"



#include "machine.h"



#include <vector>

#include <string>



#include <cstdlib>



#include <stdlib.h>



#include <ctime>

#include <iostream>

#include <iomanip>

#include <string>

using namespace std;



Machine::Machine() {



srand(time(NULL));



attract\_Mode();



spin\_Reels();



}



void Machine::attract\_Mode() {



y.Banker::jPot = 5.00;



system("cls");



cout << "======================================================================" << endl;

cout << "= =" << endl;



cout << "= JackPot: " << std::fixed << std::setprecision(2) << y.Banker::jPot << " =" << endl;



cout << "= ------------ =" << endl;



cout << "= Money: " << std::fixed << std::setprecision(2) << y.Banker::curr << " =" << endl;

cout << "= ------------ =" << endl;

cout << "= =" << endl;

cout << "= Press Enter To Spin The Reel!!! =" << endl;

cout << "= =" << endl;

cout << "= " << a << " " << b << " " << c << " =" << endl;

cout << "= ---" << d << " " << e << " " << f << "--- =" << endl;

cout << "= " << g << " " << h << " " << i << " =" << endl;

cout << "= =" << endl;

cout << "======================================================================" << endl;

}

void Machine::spin\_Reels() {

string reel\_1[] = { "Orange", "Melon ", " 7 ", "Orange", "Orange", " 7 ", "Melon " };

string reel\_2[] = { " 7 ", "Melon ", " 7 ", "Melon ", "Melon ", "Orange", "Melon " };

string reel\_3[] = { " 7 ", "Orange", "Orange", " 7 ", "Melon ", "Melon ", "Orange" };

reel\_Act1 = 0, reel\_Act2 = 0, reel\_Act3 = 0, top\_Row1 = 0, top\_Row2 = 0, top\_Row3 = 0, top\_Row1\_ = 0, top\_Row2\_ = 0, top\_Row3\_ = 0;

reel\_Act1\_ = 0, reel\_Act2\_ = 0, reel\_Act3\_ = 0, bot\_Row1 = 0, bot\_Row2 = 0, bot\_Row3 = 0, bot\_Row1\_ = 0, bot\_Row2\_ = 0, bot\_Row3\_ = 0;

a, b, c, d, e, g, f, h, i, j;



while (true) {



if (y.Banker::curr >= 0.10) {

y.Banker::curr = y.Banker::curr - 0.10;

reel\_Act1 = (rand() % 7);



reel\_Act2 = (rand() % 7);



reel\_Act3 = (rand() % 7);



cout << endl;

top\_Row1 = reel\_Act1 - 1;



top\_Row2 = reel\_Act2 - 1; a



top\_Row3 = reel\_Act3 - 1;



top\_Row1\_ = vector\_Correct(top\_Row1);



top\_Row2\_ = vector\_Correct(top\_Row2);



top\_Row3\_ = vector\_Correct(top\_Row3);



bot\_Row1 = reel\_Act1 + 1;



bot\_Row2 = reel\_Act2 + 1;



bot\_Row3 = reel\_Act3 + 1;

bot\_Row1\_ = vector\_Correct(bot\_Row1);

bot\_Row2\_ = vector\_Correct(bot\_Row2);

bot\_Row3\_ = vector\_Correct(bot\_Row3);



a = reel\_1[top\_Row1\_];



b = reel\_2[top\_Row2\_];



c = reel\_3[top\_Row3\_];



d = reel\_1[reel\_Act1];



e = reel\_2[reel\_Act2];



f = reel\_3[reel\_Act3];

g = reel\_1[bot\_Row1\_];



h = reel\_2[bot\_Row2\_];



i = reel\_3[bot\_Row3\_];



system("cls");

cout << "======================================================================" << endl;

cout << "= =" << endl;

cout << "= JackPot: " << std::fixed << std::setprecision(2) << y.Banker::jPot << " =" << endl;

cout << "= ------------ =" << endl;

cout << "= Money: " << std::fixed << std::setprecision(2) << y.Banker::curr << " =" << endl;



cout << "= ------------ =" << endl;



cout << "= =" << endl;



cout << "= Press Enter To Spin The Reel!!! =" << endl;

cout << "= =" << endl;



cout << "= " << a << " " << b << " " << c << " =" << endl;

cout << "= ---" << d << " " << e << " " << f << "--- =" << endl;

cout << "= " << g << " " << h << " " << i << " =" << endl;

cout << "= =" << endl;

cout << "======================================================================" << endl;



calcWin(d, e, f);



}



else {

cout << endl;

y.Banker::curr = y.Banker::input\_Money(y.Banker::curr);

}

}



}



void Machine::calcWin(string d, string e, string f) {

if (d == "Orange" && e == "Orange" && f == "Orange") {



y.Banker::curr = y.Banker::curr + 0.50;



y.Banker::jPot = y.Banker::jPot + 0.10;



cout << "You Win!!!!" << endl;



cout << "50p has been added to your Money" << endl;



};



if (d == "Melon " && e == "Melon " && f == "Melon ") {



y.Banker::curr = y.Banker::curr + 1.50;



y.Banker::jPot = y.Banker::jPot + 0.10;

cout << "You Win!!!!" << endl;



cout << "1.50 has been added to your total money" << endl;



}



if (d == " 7 " && e == " 7 " && f == " 7 ") {



y.Banker::curr = y.Banker::curr + y.Banker::jPot;

cout << "JACKPOT!!!!!!!!!!!!!" << endl;



cout << "Your Total Is: " << y.Banker::curr << endl;;



y.Banker::jPot = 0;



}



cout << "Please Input a letter based on what you want to do: " << endl << "1: Input Money " << endl << "2: Output Money" << endl << "3: Continue Playing" << endl; cin >> j;

if (j == "1") {

y.Banker::curr = y.Banker::input\_Money(y.Banker::curr);

cout << endl;



}



else if (j == "2") {



y.Banker::curr = y.Banker::output\_Money(y.Banker::curr);



cout << endl;

}



else {



y.Banker::jPot = y.Banker::jPot + 0.10;



cin.get();



}



}



int Machine::vector\_Correct(int p) {



if (p > 6) {



return 0;



}



else if (p < 0) {



return 6;

}

else if (p >= 0 && p <= 6) {



return p;



}



}



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Banker.cpp



#pragma once;

#include <iostream>

#include <windows.h>

#include <string>

#include "Banker.h"

using namespace std;

Banker::Banker()

{

curr = 0;

jPot = 0;

}

float Banker::input\_Money(float curr) {

ten\_pence = 0, twenty\_pence = 0, fifty\_pence = 0, pound = 0;

cout << "Input coins in this order : 10p's, 20p's, 50p's, Pound Coins: " << endl;

cin >> ten\_pence >> twenty\_pence >> fifty\_pence >> pound;

curr = curr + (ten\_pence \* 0.10) + (twenty\_pence \* 0.20) + (fifty\_pence \* 0.50) + pound;

cout << curr;



return curr;



}



float Banker::output\_Money(float curr) {

cout << "I will now calculate your outputted money: ";

ten\_pence = curr \* 10;

cout << ten\_pence << "\* 10p's Have Been Outputted";

curr = 0;

return curr;

}

