**Leslie’s Awesome Battle Game**

CSC 17-A 42824

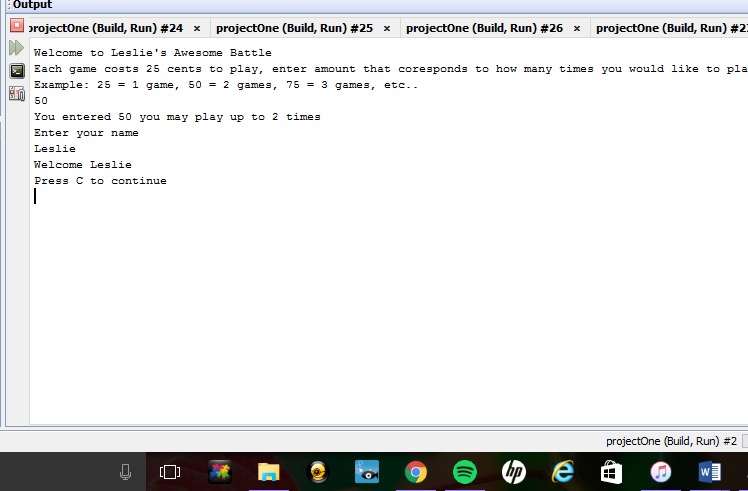
Leslie Rodriguez

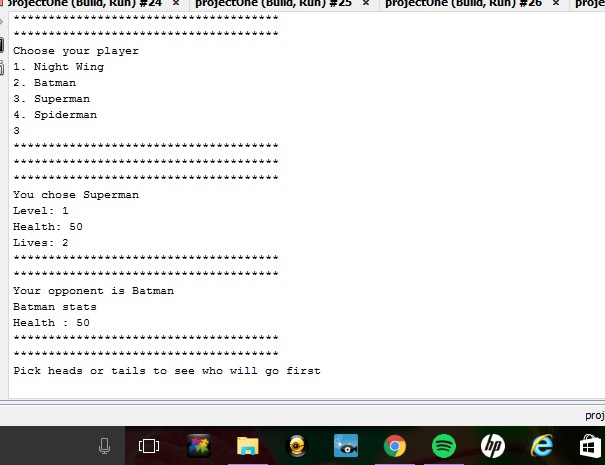
5/4/16

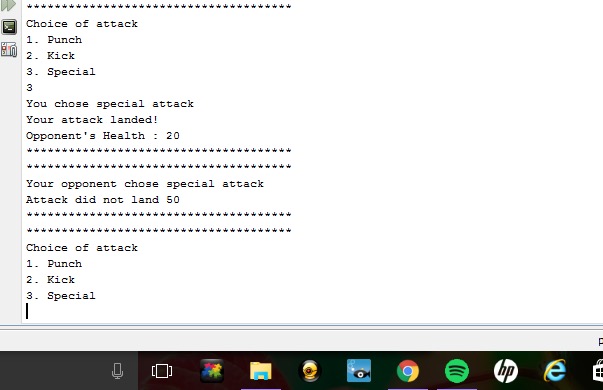
Introduction

My program is a simple combat game. It is made for one player only. That one player battles against the computer and its attacks. The player is given the option of 3 different maneuvers all in which the computer can also choose from. Whichever player reaches 0 health or less, loses. I decided to build this program because on my free time I enjoy playing many different types of video games and I am really big on superheroes and combat movies as well. I decided to try and mix some of my favorite games and characters into a program that could be simple enough for my little sister to play.

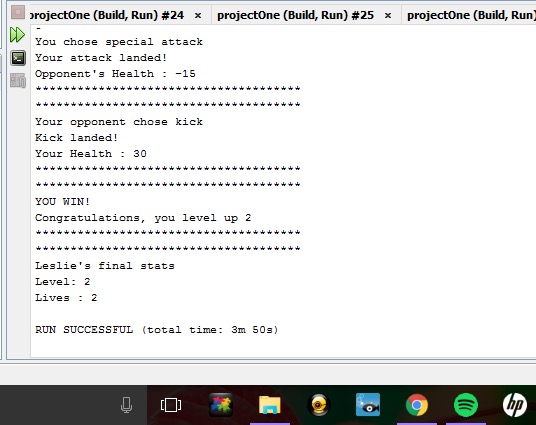
Game Description

The player must first insert “money,” each life costs 25 cents so the amount of tries are based on how much is given, once the amount is entered the player inputs his/her name. 

You then have the option to choose between 4 characters: Night Wing, Batman, Superman, and Spiderman. Once the character is picked the program will then randomly choose an opponent to go against you. The opponents in the program are also: Night Wing, Batman, Superman, and Spiderman. So yes it is possible to fight against the same character, its randomized. The player will start with 50 points of health (kind of like a health bar), however many lives purchased, and begin at level 1. The game will begin with a coin toss; it will ask the player to click "H" for heads "T" for tails. 

The program will "toss a coin," which is randomized, if the player’s choice is what the coin was "flipped" to then they make the first move, if not then the opponent attacks first. There are three attacks: punch, kick, and special attack. A punch takes 10 away from your life, a kick 15, and a special attack 20, that is however only if it lands. I made the attacks random so that the attacks may not always hit. So even if you do click, say punch, it does not necessarily mean that your opponent will lose 10 points, the punch could miss.

The attacks will go back and forth (after the player tries to hit, the opponent will then go, and so on and so forth) until either the player’s life or the opponent’s life is less than or equal to zero. The winner is then announced, if the player wins they go up a level, if they lose they lose a life. After the results are displayed, it will save the results in a text file, once the program is played again it will give the option to play or view any old data/games played.



|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Name** | **Description** | **Location** |
| Structure | PlayerOne | Holds players information and stats | PlayerOne.h |
| Int | Lives | Player’s total lives | PlayerOne.h |
|  | Level | Player’s starting level | PlayerOne.h |
|  | Health | Player’s starting health | PlayerOne.h |
|  | Health2 | Opponents starting health | PlayerOne.h |
|  | num | Total amount player inputs in money | PlayerOne.h |
| Char | C | To continue the game | Int main() |
|  | Choice | To choose a player | Void playerChoice() |
|  | Flip | Player choice in heads or tails | Void coinFlip() |
|  | Coin | The results of the random coin flip | Bool coinFlip() |
|  | Att | Players choice in attack | Void attack() |
|  | Attack[] | Randomized attack | Void oppAttack() |
| Bool | coinFlip | Returns true if flip is equal to coin | Int main() |

It was important to me because it really allowed me to incorporate different methods of programming that I actually had in the past. Many

of the problems that arose were not direct and I had to read deeper into functions that allowed me to master codes I was simply okay on. It was tedious, but it was an important part in helping me code better.

Summary

Lines of code: about 330

One structure

5 functions

The criteria include memory allocation, functions with structures, pointers with arrays and arrays of structures, character arrays, and reading and writing to binary files. I was able to incorporate each concept fairly easily however I did run into issues when it came to pointers and memory allocation. In the past I had trouble, mainly with pointers, I continued to get errors throughout my code when running my program and trying to incorporate pointers. It took me roughly 12 hours to finally get my program to run properly. I started with a simple program that was much quicker to write but it became extremely tricky for me to include the requirements.

Psuedo code

Struct {

Initialized the information (health, lives, and level)

}

Int main {

Input of money

Input of name

Display menu

Players choice

If chose to continue

{

Computers random choice in opponent

Coin flip is called

If you are right

You attack first

Then opponent attacks

Continues till someone gets to < 0

If you win you level up

If you lose you lose a life

If you are wrong

Computer attacks first

Then you attack

Continues till someone gets to < 0

If you win you level up

If you lose you lose a life

If you have more than one life you will continue to play

if you lose all your lives you current level and name is saved

The game ends

}

}

Code

/\*

\* File: PlayerOne.h

\* Author: Leslie Rodriguez

\*

\* Created on May 4, 2016, 11:20 PM

\*/

#ifndef PLAYERONE\_H

#define PLAYERONE\_H

using namespace std;

struct PlayerOne

{

string name;

int health;

int lives;

int level;

int health2;

};

#endif /\* PLAYERONE\_H \*/

Main.cpp

#include <cstdlib>

#include <iostream>

#include <string>

#include <ctime>

#include <fstream>

#include "PlayerOne.h"

using namespace std;

void playerChoice(int num, PlayerOne \*you);

void compPlayer(PlayerOne \*you);

bool coinFlip();

void attack(PlayerOne \*opp);

void oppAttack(PlayerOne \*you);

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

PlayerOne leslie;

char c;

int num;

cout << "Welcome to Leslie's Awesome Battle " << endl;

cout << "Each game costs 25 cents to play, enter amount that coresponds to how many times you would like to play" << endl;

cout << "Example: 25 = 1 game, 50 = 2 games, 75 = 3 games, etc.. " << endl;

cin >> num;

cout << "You entered " << num << " you may play up to " << (num/25) << " times" << endl;

cout << "Enter your name " << endl;

cin >> leslie.name;

cout << "Welcome " << leslie.name << endl;

leslie.lives=num/25;

leslie.level=1;

playerChoice(num, &leslie);

while(leslie.lives > 0 ){

cout << "Press C to continue " << endl;

cin >> c;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

if (c == 'c')

{

compPlayer(&leslie);

leslie.health=50;

leslie.health2=50;

if (coinFlip()==true)

{

do{

attack(&leslie);

oppAttack(&leslie);

}while((leslie.health>0)||(leslie.health2>0));

if (leslie.health<=0)

{

leslie.lives-=1;

cout << "Sorry you lost a life " << leslie.lives << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else if (leslie.health2<=0)

{

cout << "YOU WIN!" << endl;

leslie.level+=1;

cout << "Congratulations, you level up " << leslie.level << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

else

{

do {

oppAttack(&leslie);

attack(&leslie);

}while((leslie.health>0)||(leslie.health2>0));

if (leslie.health<=0)

{

leslie.lives-=1;

cout << "Sorry you lost a life " << leslie.lives << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else if (leslie.health2<=0)

{

cout << "YOU WIN!" << endl;

leslie.level+=1;

cout << "Congratulations you level up " << leslie.level << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

file.open("stats.txt", ios::out | ios::binary);

cout << leslie.name << "'s final stats " << endl;

cout << "Level: " << leslie.level << endl;

cout << "Lives : " << leslie.lives << endl;

file.write(leslie.name, leslie.level);

file.close();

cout << "Your stats have been recorded " << endl;

file.open("stats.txt", ios::in | ios::binary);

file.read(leslie.name, leslie.level);

cout << "Hope you enjoyed, have a nice day grading" << endl;

}

else

{

cout << "Invalid entry, Goodbye " << endl;

}

}

return 0;

}

void playerChoice(int num, PlayerOne \*you)

{

char choice;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "Choose your player " << endl;

cout << "1. Night Wing" << endl;

cout << "2. Batman " << endl;

cout << "3. Superman" << endl;

cout << "4. Spiderman " << endl;

cin >> choice;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

if (choice == '1')

{

cout << "You chose Night Wing" << endl;

}

else if (choice == '2')

{

cout << "You chose Batman" << endl;

}

else if (choice == '3')

{

cout << "You chose Superman" << endl;

}

else if (choice =='4')

{

cout << "You chose Spiderman" << endl;

}

else

{

cout << "Invalid entry!" << endl;

}

you->health=50;

cout << "Level: " << you->level << endl;

cout << "Health: " << you->health << endl;

cout << "Lives: " << you->lives << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

void compPlayer(PlayerOne \*opp)

{

srand(time(NULL));

string comPlayer[4] = { "Night Wing", "Batman", "Superman", "Spiderman" };

int ran = rand() % 4;

cout << "Your opponent is " << comPlayer[ran] << endl;

cout << comPlayer[ran] << " stats" << endl;

opp->health2=50;

cout << "Health : " << opp->health2 << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

bool coinFlip()

{

char flip;

char coin;

cout << "Pick heads or tails to see who will go first " << endl;

cin >> flip;

if (rand() % 2 == 0)

coin = 'H';

else

coin = 'T';

if (toupper(flip) == coin)

{

cout << "You chose right, you start" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

return true;

}

else

{

cout << "You chose wrong, sorry" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

return false;

}

}

void attack(PlayerOne \*opp)

{

char att;

srand(time(NULL));

cout << "Choice of attack " << endl;

cout << "1. Punch " << endl;

cout << "2. Kick " << endl;

cout << "3. Special " << endl;

cin >> att;

// opp->health=50;

if (att == '1')

{

cout << "You chose punch " << endl;

if (rand() % 6 != 0)

{

cout << "Your punch landed!" << endl;

opp->health2 -=10;

cout << "Opponent's Health : " << opp->health2 << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else

{

cout << "Your punch did not land " << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

else if (att == '2')

{

cout << "You chose kick " << endl;

if (rand() % 4 != 0)

{

cout << "Your kick landed!" << endl;

opp->health2 -= 15;

cout << "Opponent's Health : " << opp->health << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else

{

cout << "Your punch did not land " << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

else

{

cout << "You chose special attack " << endl;

if (rand() % 2 != 0)

{

cout << "Your attack landed!" << endl;

opp->health2 -= 20;

cout << "Opponent's Health : " << opp->health2 << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else

{

cout << "Your attack did not land " << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

}

void oppAttack(PlayerOne \*you)

{

srand(time(NULL));

char attack[3] = { '1', '2', '3' };

int index = rand() % 3;

if (attack[index] == '1')

{

cout << "Your opponent chose punch" << endl;

if (rand() % 6 != 0)

{

cout << "Punch landed!" << endl;

you->health -= 10;

cout << "Your Health : " << you->health << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else

{

cout << "Punch did not land " << you->health << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

else if (attack[index] == '2')

{

cout << "Your opponent chose kick" << endl;

if (rand() % 4 != 0)

{

cout << "Kick landed!" << endl;

you->health2 -= 15;

cout << "Your Health : " << you->health << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else

{

cout << "Kick did not land " << you->health << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

else

{

cout << "Your opponent chose special attack" << endl;

if (rand() % 2 != 0)

{

cout << "Attack landed!" << endl;

you->health -= 20;

cout << "Your Health : " << you->health << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

else

{

cout << "Attack did not land " << you->health << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout <<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

}

}

}