**Project 1**

Title

<Fighter Fighter>

Class:

CIS-5

Due Date:

May 4, 2016

By:

Bassam Elsaleh

Table of contents

Summary………………………………………………………………………………….Pg.3

Pseudo Code…………………..………………………………………………………….Pg.4

Flow Chart…….………………………………………………………………………….Pg.6

Constructs Used.………………………………………………………………………….Pg.8

Sources……………………………………………………………………………………Pg.9

Summary

Fighting games are always fun to play from the more famouse mortalcombat to the lesser known like skull girls. These games deal with a lot of skill and timing, but there were and still exsistgames that are not as complex and chalinging. This game requiers little to no skill, its a simple rock papper sicors match. Where the user can have fun with friends or play agenst the computer.

This porgram is not complex yet requiered over 900 lines because of one simple aspect of the game, special attacks, witch will be covered in this section. Each player will start in the menu and chose to either play one player, two players, look at the rules or quit. Now, as for rules, the player can use one of 4 moves, the first being a simple punch that will deal 2 damage to the opponat, now that doesn’t mean there is no way to prevent the attack, the players are allowed to block the damage by using the next ability Blocking. Blocing stops all the damage coming from the punch, now some players are strange and like to joke around by doing noting and just keep blocking so I created the specual attack. The special attack will deal damage to a caricter that isn blocking and 2 damge to a blocking charicter. Though, because the attack is so strong I had to ristrict the amount of special attacks to 4 times, this is why the code took up so manny lines, I had to re-write the code for every instance that one or both players might run out of special attacks. I then added one more move just for fun that would allow the charicter to be able to counter only the special attack, dealing all 4 damge back to the player, but wont block the puch attack.

Pseudo Code

//function for the menu

//function for the rules

//function for one player

//function for two players

//declare variables

//do while choice=4

//call menu function

//players pick

//go to the function player picked

//exit stage right

//menu function

//show menu

//rules function

//declare variables

//show the rules until player exits

//one player function

//declare variables

//computer random

//player picks move

//show health

//1 = punch

//2 = spc.

//3 = block

//4 = counter

//keep looping till one player has no health or quits

//player and computer both pick their movies as described in the rules

//describe to the player what happens and heath is removed where needed.

// show heath and specials

//same code but both players are out of specials

//same code only player one is out of specials

//same code only cpu is out of specials

//if both players die

//if player one dies

//if cpu dies

//if player one gives up

//go to file and show that player one wins.

//two player function

//declare variables

//player picks move

//player 2 picks

//show health

//1 = punch

//2 = spc.

//3 = block

//4 = counter

//keep looping till one player has no health or quits

//player and computer both pick their movies as described in the rules

//describe to the player what happens and heath is removed where needed.

// show heath and specials

//same code but both players are out of specials

//same code only player one is out of specials

//same code only cpu is out of specials

//if both players die

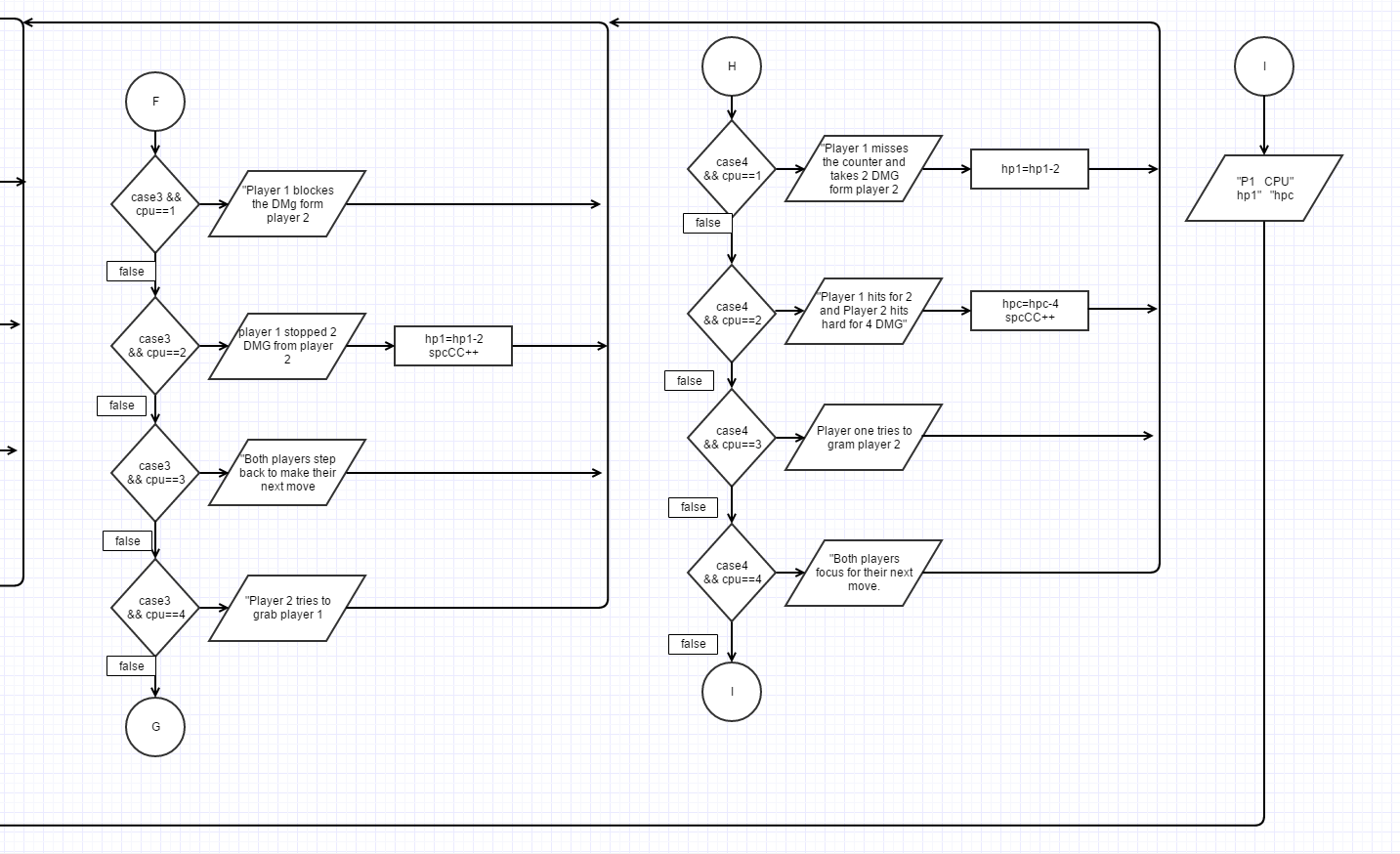
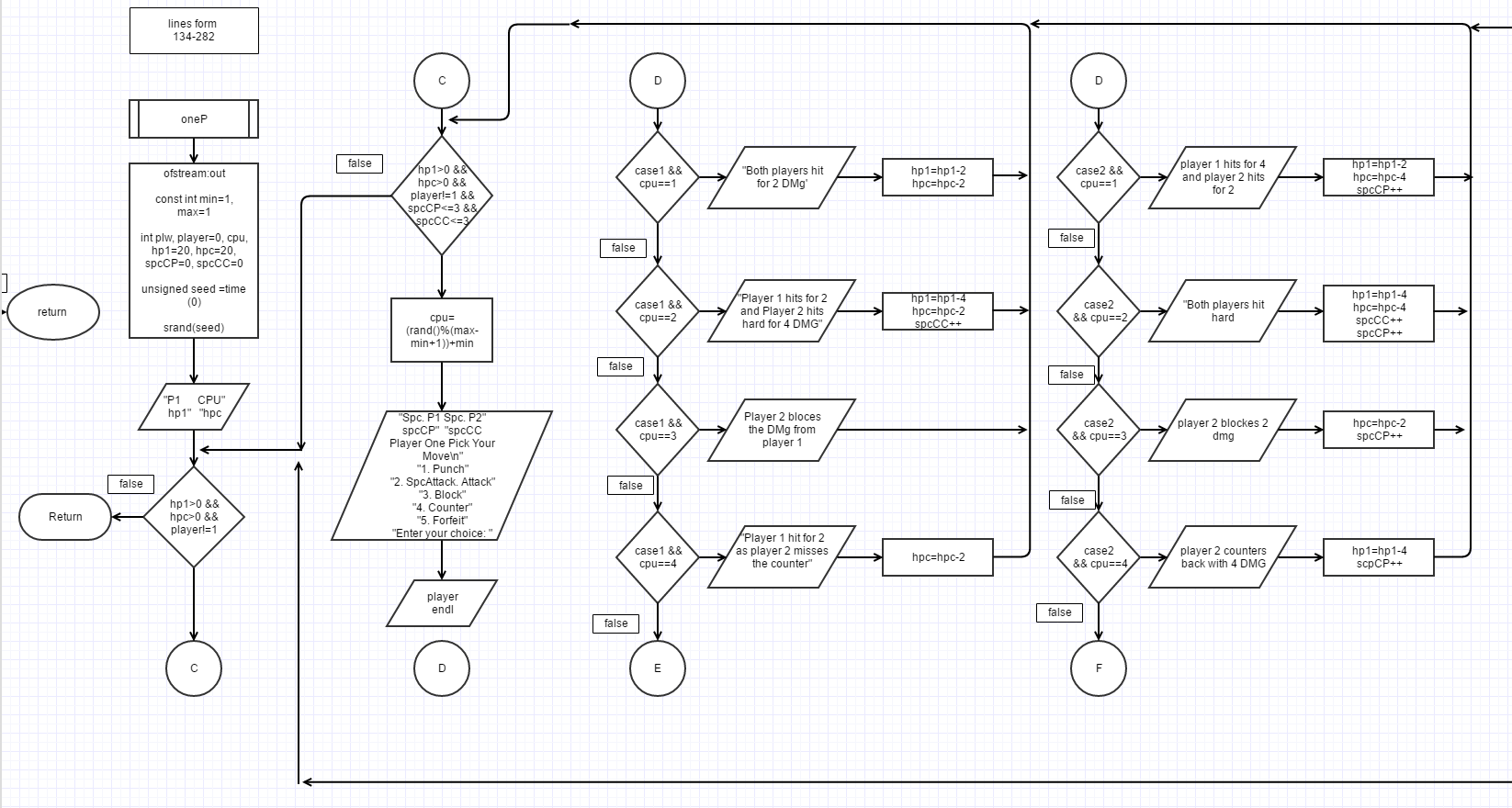
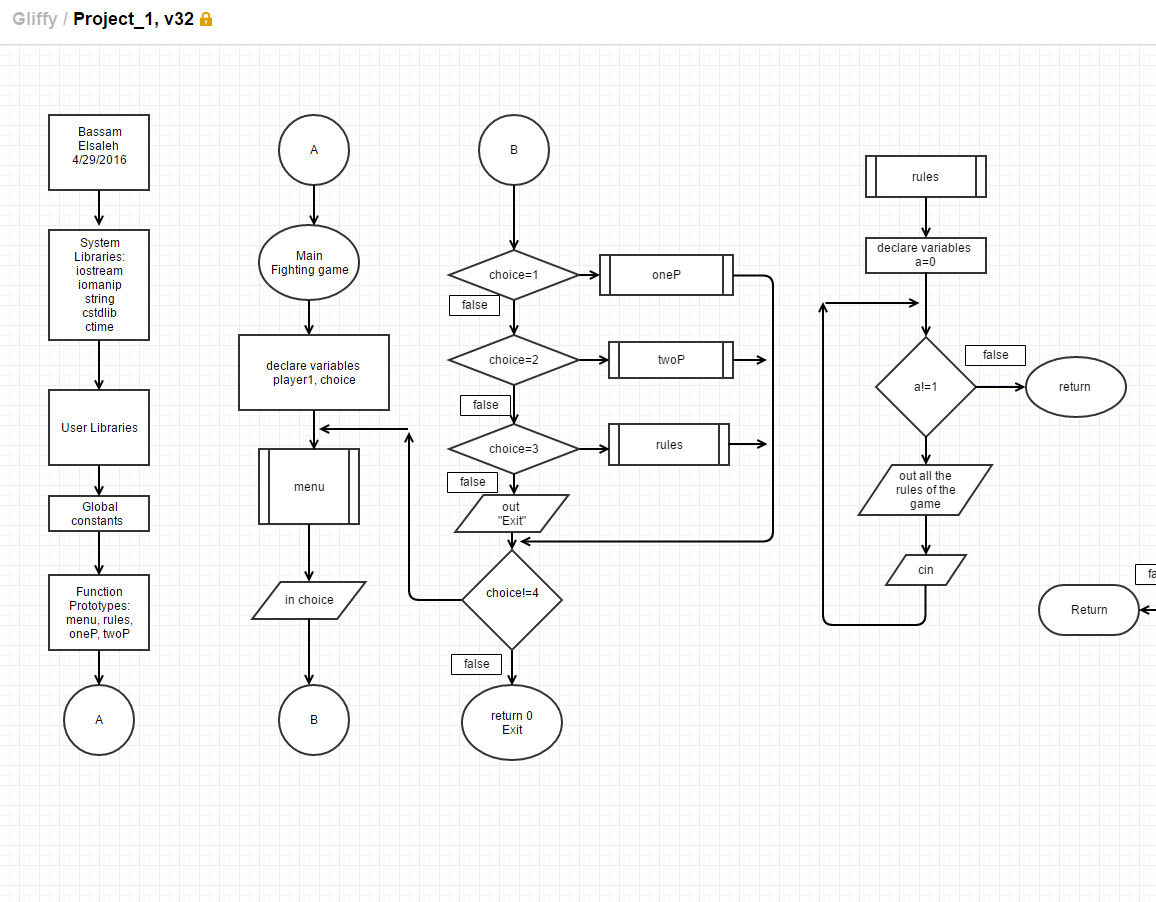
//if player one dies

//if player 2 dies

//if both players give up

//if player one gives up

//if player two gives up

Flow Chart

Constructs used

Do loops: I mainly used the do loop in the menu so that it runes right into the menu and alowing the player to pick their game mode.

For loops: I ued this in the rules section when I didn’t need to incrament but I set a veriable for what the user will input to exit the statement when they press 1

Switch statements: I belive that using the switch statement was the best way to code with the lest amount of lines, rather than using an outstanding amount of if and if else statements I decided to run a switch stament that only runs through the second players options that exsistes inside player ones switch statement.

If, else if and else statements: my code was riddled with this type of code to decide the outcome of the players hp and win lose.

Rand: This was only used in single player where the computer would need to pick its on moves. So I had to brin in cstdlip file and ctime to make sure the reandom generater wasn’t reporting the same random numbers.

Funcitons: I decided to use funciton in my project because doing one player allows for a whole set of rules and outcomes that I found were better to keep all on its on so that the in 2 palyer mode there wouldn’t be any misscomunication in the code.

fstream: I decided to send an out file that will occer if you win the computer in single player.

Data Types: I used primerly int’s in my codding no floats were needed because I didn’t have any decimals, i used ofstream to read out to a file, and had set the const in for a max and an min for the random generator. Along with the random generator I needed and unsigned seed and srand to make the random work.

Sources

Gaddis, T. (2012). Starting out with C. Boston: Pearson Addison-Wesley.

Rcc Class. Dr. Lehr.