

Adam Wright

CS-162

Project 3: Design and Reflection

Project Plan:

The problem in project three is to create a five player game where all of the characters are derived classes of an abstract base class. The abstract base class is named character and it will hold the private variables for strength amount, armor amount and whether the character is medusa for each character. The game's character selection menu will have two while loops with each allowing the choice of one through five to pick a pair of characters to do battle. Then the battle will run in a while loop which checks if both characters are still alive to continue running. Each loop will then have the two characters attack each other and if one dies then a message announcing the winner will be printed.

Pseudocode:

do while loop wrapping the game to allow restarting and exit

cout a greeting

entering 1 continues to the character selection menu

entering 2 will exit the application

choose 1-5 to pick the first character

print the stats for both characters during round

if character two dies print character one wins

choose 1-5 to pick the second character

print the stats for both characters during round

if character one dies print character two wins

while loop until a character dies

print the winner when the game ends

while loop returns to beginning

Testing Table:

| Testing Input | Expected Output | Actual Output |
|--------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| run program main | do while prints greeting | do while prints greeting |
| User enters 1 to enter game | prints character 1 menu | prints character 1 menu |
| User enters 2 to exit program | back to command line | back to command line |
| character menu 1 character menu 2 | must press 1-5 or loop continues must press 1-5 or loop continues | must press 1-5 or loop continues must press 1-5 or loop continues |
| Game runing | roundcount prints the current round | roundcount prints the current round |
| player one attack player one attack player two defense player two defense player two defense player two's life at 0 or less | player one attack prints player name and current roll prints player two name armor and strength prints player two name and roll prints player two strength-after prints player one wins prints | player one attack prints player name and current roll prints player two name armor and strength prints player two name and roll prints player two strength-after prints player one wins prints |
| player two attack player two attack player one defense player one defense player one defense player one's life at 0 or less | player two attack prints player name and current roll prints player one name armor and strength prints player one name and roll prints player one strength-after prints print player two wins | player two attack prints player name and current roll prints player one name armor and strength prints player one name and roll prints player one strength-after prints print player two wins |
| attack rolls | | |
| Vampire Barbarian blue Men Medusa Medusa Harry Potter | rolls 1-12 rolls 1-6 and 1-6 rolls 1-10 1-10 1-6 rolls 1-6 1-6 rolls 12 player dies rolls 1-6 1-6 | rolls 1-12 rolls 1-6 and 1-6 rolls 1-10 1-10 rolls 1-6 1-6 rolls 12 players dies rolls 1-6 1-6 |
| defense rolls | | |
| Vampire Vampire Barbarian blue Men blue Men blue Men Medusa Harry Potter Harry Potter Harry Potter start game menu | rolls 1-6 50percent no attack rolls 1-6 and 1-6 above 8 strength 1-6 1-6 1-6 above 4 strength 1-6 1-6 above 0 strength 1-6 rolls 1-6 rolls 1-6 1-6 1st life strength 10 2nd life strength 20 play again or quit | rolls 1-6 50 percent no attack rolls 1-6 and 1-6 above 8 strength 1-6 1-6 1-6 above 4 strength 1-6 1-6 above 0 strength 1-6 rolls 1-6 rolls 1-6 1-6 1st life strength 10 2nd life strength 20 play again or quit |
| User enters 1 User enters 2 | start a new game exit to command line | start a new game exit to command line |

Reflection:

This project had many features to it that we have used over the last month and so as a whole it didn't contain any new language feature which required a large logical jump. I wasn't sure if I wanted to create another class to put the game into, but it seemed like a waste to juggle another file and then just to turn the main function into a place with very little in it. This lead me to just use main for my menus and all of my game play loop running. The big difficulty was in thinking though the battle logic for the first player and then getting the combat to play out correctly and then making the necessary changes to that logic in the other character's

implementations. In practice this really just meant conditionals to check if the character doing battle was medusa or if the character was Harry Potter, that he was on life one or had been resurrected. In making this document I was able to catch that I hadn't added that isMedusa is true in the Medusa constructor. I just hope that I got all of my logic correct and that my testing was thorough enough to have caught all of my mistakes.

Class Diagram:

| | | | | |
|---------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| | | BASE CLASS | | |
| | | class Character int armor int strength bool isMedusa = false Character() Character(int inputArmor, int inputStrength) virtual int attack() = 0 virtual void defense(int inputAttack, bool isMedusa) = 0 virtual int getStrength() virtual bool getMedusa() | | |
| DERRIVED | DERRIVED | DERRIVED | DERRIVED | DERRIVED |
| class Vampire Vampire() int attack() void defense(int attack, bool inputMedusa) | class Barbarian Barbarian() int attack() void defense(int attack bool inputMedusa) | class Blue men Bluemen() int attack() void defense(int attack bool inputMedusa) | class Medusa Medusa() int attack() void defense(int attack bool inputMedusa) bool isMedusa = true | class Harry Potter Harry Potter() int attack() void defense(int attack bool inputMedusa) int timesKilled = 0 |