Project 3
Yucen Cao

Design description:

In this project, we will develop a fantasy combat game. This is going to be part 1 of the game development in which we develop the characters and a menu to test the characters.

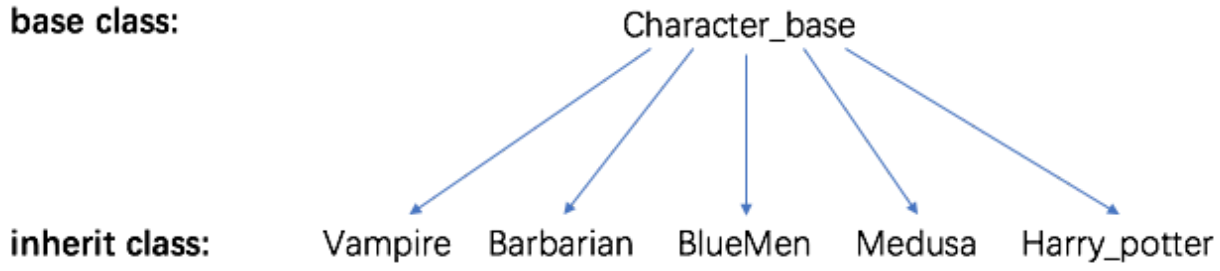
The game universe contains Vampire, Barbarian, Blue Men, Medusa, and Harry Potter. Each character has attributes of attack, defense, armor, and strength points.

- Each combat between 2 characters ends when one of the characters die.
- Each round consists of two attacks, one for each character. For each attack, attacker and defender both generate dice rolls.
- Damage = attacker's roll – defender's roll – defender's armor. Then the value of that damage is subtracted from the defender's strength points.
- The two attack will happen in the same time at the beginning of each combat. The two attacker can both die in the same combat.

Project Structure:

It has one base class called Character_base, and five inherit class called Vampire, Barbarian, Blue Men, Medusa, and Harry Potter.

class hierarchy diagram



Otherwise, I have main, game, menu, and input_validation.

They can split to 3 important parts, 1.user input and validation. 2. Character base class and five inherit class. 3. Game logic and print

My design

- In the menu, when user input 1-5 , choose character from 1 "Vampire", 2 "Barbarian", 3 "BlueMen", 4 "Medusa", 5"Harry_potter"
- After choosing the character, it will print result for each combat, till one or both character die.

Test table:

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Input float number	1.5	bool validation_int(string s) ;	Show enter wrong number, please enter again	Show enter wrong number, please enter again
Input character	A or bb or #	bool validation_int(string s) ;	Show enter wrong number, please enter again	Show enter wrong number, please enter again
Input negative number	-12	bool validation_int(string s) ;	Show enter wrong number, please enter again	Show enter wrong number, please enter again
Input 0 for steps and sides	0	bool validation_int(string s) ; int validation_positive_int(string temp) ;	Show enter wrong number, please enter again （for steps and sides it can not be 0)	Show enter wrong number, please enter again
Input not only 0 or 1 for the choose parts.	2	bool validation_int(string s) ; int	Show enter wrong number, please enter again （when user choose something, it has to be 0/1	Show enter wrong number, please enter again

		validation_one_zero(string temp);	according to the design.	
Input 0 for start the game	0	bool validation_int(string s) ; int validation_one_zero(string temp);	Exit the game	Exit the game
Vampire VS Vampire	1 1	fight();	print result for the combat, both vampire have the chance to use Charm	print result for the combat, both vampire have the chance to use Charm
Medusa VS Vampire	4 1	fight();	print result for the combat, vampire have the chance to use Charm, and medusa may use the Glare. The charm will trumps glare.	print result for the combat, vampire have the chance to use Charm, and medusa may use the Glare. The charm will trumps glare. Medusa win.
Blue Men VS Medusa	3 4	fight();	print result for the combat, medusa may use the Glare, and blue men's defence will change	print result for the combat, medusa may use the Glare, and blue men's defence will change. BlueMen win.
Medusa VS Medusa	4 4	fight();	print result for the combat, both medusa may use the Glare	print result for the combat, both medusa may use the Glare. Medusa win.
Harry Potter VS Barbarian	5 2	fight();	print result for the combat, harry potter use Hogwarts.	print result for the combat, harry potter use Hogwarts. Harry_potter win.
Medusa VS Harry Potter	4 5	fight();	print result for the combat, medusa may use the Glare, and the first time harry potter may use Hogwarts to against Glare	print result for the combat, medusa may use the Glare, and the first time harry potter may use Hogwarts to against Glare. Harry_potter win.

Reflection:

First time, I write the function in different way that put the attack function inside the defence function. In this way, the function will get which type of the Character, then will know the total reduce Strength point. I almost finish it, when I double check the requirement, I have to have the Attack and Defense function separate, and some of them have to Override.

So I change the logic, and rewrite the functions. Put Attack and Defense function separate, then get the total reduce Strength point.