Vannear Por
CS 162
May 13, 2017

Reflection Project 3

After a bit of practice with polymorphism prior to this project, creating the creature base class and the subclasses was not all that difficult. The project requirements listed what was basically needed for the member variables and what kind of functions they would need. So in my initial design I listed out the basic data needed that was common for every creature, which were: Creature Type, attack points, armor points, strength points, and defeat status; of course I tried to list out as much common data I could think of because having more is better than having missing variables. I set "get" functions for each variable and I felt like I was already halfway done through planning this project. So after planning out the base abstract class, I started to create the Creature class code file. After that I went on to create all 5 creature's .hpp files which mirrored the Creature base class. They all looked similar and I proceeded with the first .cpp implementation file; I started with Barbarian because he had no special abilities/requirements. This is where the problem begin in really implementing the functions to tie together smoothly. I realized soon that my attack and defense points had to come from a roll, so I decided to set both data to 0 in the default constructor, I also decided to give the attack and defense "get" functions to get it's data from a newly created "attackRoll" and "defenseRoll" function to get data from randomized rolls. I created 2 new functions here by solving this problem with getting attack and defense points. Everything started to become straight forward and I copied and pasted my finished "Barbarian" class into all the other 4 creatures and changed the names.

Before I started specializing the classes even more, I decided that maybe I should map out my "Main" file to lay out some plans before I decide to go in-depth with the special abilities of the characters. Issues started to occur when I was trying to figure out how to implement the "attack rounds" and how to apply damage and to apply the defenses. Of course, I knew I needed to do while loops to repeat the rounds, and each round needed to apply damage and attack points and that I needed to utilize nearly ALL created functions within one round. After trying to figure out how to manipulate defense and attack numbers from within main, I mapped out what I needed each side to do… when one side "Attacks" and gives damage, I realized that the "Defense" should TAKE the damage and utilize it, thus I was able to change my defense "get" function, into a Defend function which takes the attack as a parameter and the function would apply the math, create a new "Damage" variable and do the math regarding the strength points from within that function. This fixed a huge headache in where ALL the math regarding the defending creature would be contained into one simple function to be called from Main. This was the biggest roadblock, figuring out the flow of data throughout each round.

After figuring this out and creating a successfully compiling file, I tested the program and realize that the numbers were skewed, particularly in the Blue Men creature which had great defense. I saw that after each round where they had higher defense than the attack, their strength points was also growing! It made me realize that "strengthPoints - damage" if damage was negative, strengthPoints would increase! So I was able to create a validation where if damage was above 0, it would subtract from

strengthPoints, if it was 0 or lower, strengthPoints would stay the same. After testing the data came out exactly as needed.

Universe
- Vampires
- Barbarian
- Blue Men
- Medusa
- Harry Potter

} · Attack
· defense
· armor
· strength points

"3d6" = rolling three 6-sided dice, "2d10" rolling two 10-sided dice

Resolve attack - 2 dice rolls · 1 attacker roll · 1 defender roll
· subtract defense roll from attack roll.

Attack:
· Attack - defense = damage
· Apply damage → subtract from armor → subtract from str. point.
· str point = ∅ → loses

Base Class : Creature (abstract)

Need Menu:
- Display 5 characters & prompt user to select 2 (can be 2 same)
- Randomly to attack first. (then take turns)
- Print results of each round
- Display winner in end
- Ask user: play again or exit?

Creature :

protected:

- Name
- Atk
- Def
- Armor
- strength Pts
- die? (T or F)

functions:

- Name
- Atk
- Def
- Armor
- strength Pts
- death (T or F)

→ Atk Roll
→ Def Roll
Get functions

Main:

Menu - Choose P1
Input → P1

Menu - Chose P2
Input → P2

set P1 and P2 data after choosing

⇓

randomize (who goes first)

⇓

While loop attack and defends

⇓

end when str pts reach 0.

Vannear Par

Test Plan

Menus:

Input	Outcome
P1 chooses creature P2 chooses creature	Display correct creature name Display correct creature name
P1 chooses creature P2 chooses creature	→ Display creatures strength points to validate default constructors.
P1 str points = 0 P2 str points = 0	P2 victory P1 victory
Medusa glare	→ Medusa wins
Harry Potter HP = 0	→ Harry Potter str Point = 20
Blue Men str point = 12-9 = 8-5 = 4-0	→ 3 defense dice → 2 defense dice → 1 defense dice
Vampire drain	→ No change in strength points
Replay Game? Yes No	→ Go back to Menu. → Exit Program