Kimberly Tom

CS 162
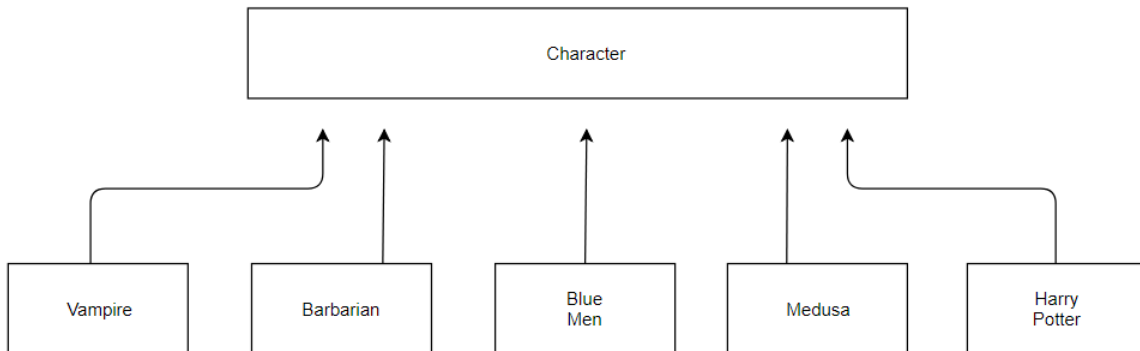
Project 3

Design for Fantasy Combat Game

1. Main
    a. Main is where we instantiate the menu
        i. Key functions and data members
            1. hold menu function
            2. set random time
2. Character base class – abstract class
    a. attack function - virtual
    b. attack die roll function
    c. defense die roll function
    d. defense function - virtual
    e. armor function
    f. damage taken function
    g. strength points function
3. Vampire – derived class of character
    a. attack roll function uses 1d12
    b. defense roll function uses 1d6
        i. charm ability
    c. attack function takes dice roll and implements attack
    d. defense function takes dice roll
    e. armor function initialized at 1
    f. strength points initialized at 18
4. Barbarian - derived class of character
    a. attack roll function uses 2d6
    b. defense roll function uses 2d6
    c. attack function takes dice roll and implements attack
    d. armor function initialized at 0
    e. strength points initialized at 12
5. Blue Men - derived class of character
    a. attack roll function uses 2d10
    b. defense roll function uses 3d6
    c. attack function takes dice roll and implements attack
    d. armor function initialized at 3
    e. strength points initialized at 12
        i. decrease defense die for every 4 points loss
6. Medusa - derived class of character
    a. attack roll function uses 2d6
        i. if rolls a 12, instant win

      b.   defense roll function uses 1d6

      c.   attack function takes dice roll and implements attack

      d.   armor function initialized at 3

      e.   strength points initialized at 8

7.   Harry Potter - derived class of character

      a.   attack roll function uses 2d6

      b.   defense roll function uses2d6

      c.   attack function takes dice roll and implements attack

      d.   armor function initialized at0

      e.   strength points initialized at 10, then 20 for second life

8.   Menu

      a.   Menu class displays options to the user

            i.   select two characters to battle

           ii.   play again after one character dies

         iii.   exit

      b.   displays the following for each attack

            i.   attack type

           ii.   defender type, armor, strength point

         iii.   attacker's attack dice roll

         iv.   defender's defend roll

          v.   total inflicted damage calculation

         vi.   defender's updated strength point amount after subtracting damage

9.   Gameplay

      a.   creates 2 characters to battle

      b.   game function to play game

            i.   character 1 attack function

                 1.   attack roll function

           ii.   character 2 defense function

                 1.   defense roll function

         iii.   character 2 attack function

                 1.   attack roll function

         iv.   character 1 defense function

                 1.   defense roll function

10. Input Validation

      a.   check user's menu choice to tell user to choose again if invalid choice is selected

Hierarchy

```
                          ┌─────────────────────────────────┐
                          │            Character            │
                          └─────────────────────────────────┘
                 ↑        ↑              ↑            ↑      ↑
        ┌──────────┐ ┌──────────┐ ┌──────────┐ ┌──────────┐ ┌──────────┐
        │ Vampire  │ │Barbarian │ │ Blue Men │ │  Medusa  │ │  Harry   │
        │          │ │          │ │          │ │          │ │  Potter  │
        └──────────┘ └──────────┘ └──────────┘ └──────────┘ └──────────┘
```

Test Plan and Results

TEST CASE	INPUT VALUES	EXPECTED OUTCOMES	OBSERVIED OUTCOMES
start menu, start	1	starts game	starts game
start menu. quit	2	quits game	quits game
start menu letter/number combinations	1a -2g b	invalid, prompt user for selection	invalid, prompt user for selection
character selection blue team, 1	1	you have selected Vampire	you have selected Vampire
character selection blue team, 2	2	you have selected Barbarian	you have selected Barbarian
character selection blue team, 3	3	you have selected Blue Men	you have selected Blue Men
character selection blue team, 4	4	you have selected Medusa	you have selected Medusa
character selection blue team, 5	5	you have selected Harry Potter	you have selected Harry Potter
character selection blue team, invalid	-5, 20, f, 6f, 0	ask user to enter integer between 1 and 5	ask user to enter integer between 1 and 5
character selection red team, 1	1	you have selected Vampire	you have selected Vampire
character selection red team, 2	2	you have selected Barbarian	you have selected Barbarian
character selection red team, 3	3	you have selected Blue Men	you have selected Blue Men
character selection red team, 4	4	you have selected Medusa	you have selected Medusa

character selection red team, 5	5	you have selected Harry Potter	you have selected Harry Potter
character selection red team, invalid	-5, 20, f, 6f, 0	ask user to enter integer between 1 and 5	ask user to enter integer between 1 and 5
select blue character: Vampire select red character: Vampire	1, 1	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender -show winner and loser after a character's strength points becomes 0	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender -show winner and loser after a character's strength points becomes 0
select blue character: Barbarian select red character: Barbarian	2, 2	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -show winner and loser after a character's strength points becomes 0	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -show winner and loser after a character's strength points becomes 0
select blue character: Blue Men select red character: Blue Men	3, 3	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is 5-8, and no higher than 6 if strength points left is 4 or less	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is 5-8, and no higher than 6 if strength points left is 4 or less

		-show winner and loser after a character's strength points becomes 0	-show winner and loser after a character's strength points becomes 0
select blue character: Medusa select red character: Medusa	4, 4	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -if medusa rolls 12 as attacker, then she instantly wins, unless vampire uses charm -show winner and loser after a character's strength points becomes 0	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -if medusa rolls 12 as attacker, then she instantly wins, unless vampire uses charm -show winner and loser after a character's strength points becomes 0
select blue character: Harry Potter select red character: Harry Potter	5, 5	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -if either/both Harry Potter strength points go down to 0 on first life, he should get 20 strength points for second life -show winner and loser after a character's strength points becomes 0	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -if either/both Harry Potter strength points go down to 0 on first life, he should get 20 strength points for second life -show winner and loser after a character's strength points becomes 0
select blue character: Vampire select red character: Harry Potter	1, 5	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender

		-if Harry Potter strength points go down to 0 on first life, he should get 20 strength points for second life -show winner and loser after a character's strength points becomes 0	-if Harry Potter strength points go down to 0 on first life, he should get 20 strength points for second life -show winner and loser after a character's strength points becomes 0
select blue character: Vampire select red character: Blue Men	1, 3	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is 5-8, and no higher than 6 if strength points left is 4 or less -show winner and loser after a character's strength points becomes 0	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is 5-8, and no higher than 6 if strength points left is 4 or less -show winner and loser after a character's strength points becomes 0
select blue character: Blue Men select red character: Medusa	3, 4	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is

		5-8, and no higher than 6 if strength points left is 4 or less -if medusa rolls 12 as attacker, then she instantly wins, unless vampire uses charm -show winner and loser after a character's strength points becomes 0	5-8, and no higher than 6 if strength points left is 4 or less -if medusa rolls 12 as attacker, then she instantly wins, unless vampire uses charm -show winner and loser after a character's strength points becomes 0
select blue character: Barbarian select red character: Blue Men	2, 3	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is 5-8, and no higher than 6 if strength points left is 4 or less -show winner and loser after a character's strength points becomes 0	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -for every 4 points of damage, blue men lose a defense die so their defense rolls cannot be higher than 12 of strength points left is 5-8, and no higher than 6 if strength points left is 4 or less -show winner and loser after a character's strength points becomes 0
select blue character: Medusa select red character: Vampire	4, 1	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender -if medusa rolls 12 as attacker, then she instantly wins, unless vampire uses charm	-shows round number, roll, damage of attacker, strength points, armor, defense, and strength points left of defender. -occasional use of vampire charm to prevent opponent attack if vampire is defender -if medusa rolls 12 as attacker, then she instantly wins, unless vampire uses charm

		-show winner and loser after a character's strength points becomes 0	-show winner and loser after a character's strength points becomes 0
after game ends, select 1	1	ask user to select character	ask user to select character
after game ends, select 2	2	quit game	quit game
after game ends, start menu letter/number combinations	1b -5g c	invalid, prompt user for selection	invalid, prompt user for selection
check for memory leaks	valgrind ./project3	no memory leaks	no memory leaks

Reflection

My program does not differ from my design much as I knew I needed to make the character class an abstract class and Vampire, Barbarian, BlueMen, Medusa, and Harry Potter derived classes. I decided to make the attack and defense functions pure virtual functions and to code it for each character, rather than just have only certain characters overriding the virtual function in the character class. I thought this would be easier because I wanted my attack rolls in the attack function and defense roll in the defense function, and each character has a different number of die count and die sides. So each character would need to have their own special attack and defense functions that override the one in character.hpp.

The only major difference is that for my gamePlay, I had to add more details to it. I needed this function to run the game simulation and I knew I needed to put the attack and defense functions for each monster in there. For example, I needed to make sure that the second monster did not lose all its healthpoints before implementing it's attack. My program required if statements to compare the strength points to zero to determine whether to continue with next round or end the game.

A problem I had early on was for each round, character 1 will attack first, and defend second, so character 2 will defend first and attack second. I did not have an if statement after character 1's attack and character 2's defense function. This created a problem because of character 2 died after character 1's attack, then character 2 should not attack because he is dead. To fix this, I created an if statement to say that if character 2 has greater than 0 strength points, then do the character 2 attack function and character 1 defense function. This solved my issue of having a dead character attack before the final results are shown.

One problem I found challenging was thinking about how to implement the special abilities of each character. For Harry Potter, it took me a while to think about how to do the Hogwarts ability. I was thinking initially if I could put something in the default constructor that would use if statements to take into account his strength points, such as using if else statements. I realized this wouldn't work as the default constructor is just called once at the beginning and not again. I then realized I can create a boolean for the Hogwarts ability and set it to true. Then in my defense function, I can say if Harry lost all his points and the ability was set to true, then set his strength points to 20, then set the boolean for Hogwarts ability to false. This made the program work as intended.

Another issue I had was with Medusa's glare ability. I was trying to figure out how to make sure that when she rolls a 12, the Medusa attack function would know to instantly kill the opponent, since we couldn't have the die roll function check if a roll of 12 was done by Medusa. I figured that I can do this by saying if her attack was equivalent to 12, then make her attack a really high value that would for sure deplete the defender's strength points. I made it so that if Medusa rolled a 12, then her attack was 500, which would for sure deplete the defender's strength points to below 0.

This lead me to a second issue I had with the Medusa glare ability. I needed to make sure that my program won't show a negative value for strength points after Medusa did the instant death (500

damage). To do this, I made use of my damageTaken parameter that my defense function takes for each character. I realized if I used an if statement and said if the damageTaken is equivalent to 500, then I can just set the strength points to zero. This way, the dead character will show 0 strength points and not a negative number.

After this project, the importance and usefulness of polymorphism is more apparent than before. It really made sense to me why we would want to use virtual functions. Even though my virtual functions were pure virtual functions and I had each character's code different for the attack and defense functions, in my gamePlay.cpp, I did not have to think about using if statements to determine which character and which character's function to call because polymorphism allowed me to just call the attack or defense function, and the program would know which character's function to call.

This program also helped ingrained the importance of inheritance. I did not need to create a getCharName() and setCharName() function for each character as all my characters inherited these functions from the character parent class.   This made things a lot easier as I didn't have to keep tying out repetitive code for each character when setting and getting their names.

This program also reinforced the importance of creating objects. My monster objects are of type character. Then, after the user selects which character they want to use, I set their input to equal my monster object. This is also another example of why inheritance is very useful, since all the "monsters" are part of the character class, they all can use the functions from the character class. Overall, I thought this project was interesting and fun to create, and was especially interesting when adding the different abilities to certain characters in the game.