

Assignment 4 Reflection:

Design:

The goals of the project was to take the creature hierarchy from Assignment 3 and use it cohesively with Assignment 4. It also incorporates the programs from lab 6 which introduces the data structures stacks and queues. The game pits two teams of characters (set by the user) against each other. Each round displays the characters fighting and the winner of the round.

When the triumphant emerge, the winner is put at the back of the lineup to keep fighting and the loser is sent to a loser pile and removed from the game. The players get awarded points for their victories. The main difference with the original creature class hierarchy will be the implementation of `creRestore()` member function. We will also use the stack and queue class, I had a death in the family last week so unfortunately I was not able to finish the lab, so these two programs I will have to make from scratch, so it will take a little longer than normal.

The winner of the game will be the one with creature(s) still standing, unless there is the event of a tie, in which both players have at least one creature left. The implementation of teams over individual players and housing them into a data class will prove to be challenging, but will add a new dimension to the game.

Stack and Queue()

- Both of these programs will be created and added to the system

Main()

- create two queues
- ask how many characters per team
- prompt team 1 leader
 - ask for name of team
 - ask for characters on team
- repeat process for team 2 leader
- run tournament
- print results
- delete all extra memory
- end program

Testing Plan:

Luckily and graciously, the derived creature classes for creature were pretty much unmodified, so that made a majority of the work already finished. The time is pretty much a wash though because of the need to create stack and queue for the lab and the assignment so I will need to get started on that

immediately, first and foremost. This puts one arm behind my back immediately because I need to build those up, test and run them and make sure there are no issues at all.

I plan on updating my main program as I go along so the basic skeleton of what is needing to be done is finished. I want to test the user input of types most of all to ensure that is down without any problems. As well as, adding their characters to the team and not just having them add it and having the program kick the character out, which is pretty much what would happen if I kept the program the same as two weeks ago. Most of the functions from the derived classes will be untouched for the most part, that makes me confident they are working without a problem as well.

Testing Results:

I immediately found out I underestimated the amount of work that would go into this program. I took into thought what you mentioned last week about bad habits and really focused on getting this finished. Had I not done that, this assignment would probably not be getting turned in. Stack was not so much of an issue to implement, the restore function was not either, however, deleting the memory for each node in the stack proved difficult until I stopped spacing out and set a pointer towards the “trash”.

The push and pop functions were a little difficult as well, but I got my head around it by telling myself it was pushing the value in and popping the value out. After that point it was easy. I also lost my creature & main programs half way through the week, which explains why this is late, I was able to restore my Linux laptop back but it only contained a few files here and there. Luckily again, it still had my derived classes because I backed those up in my Google Drive.

Testing the program:

Tested	Input	Expected Results	Results
Just ENTER	ENTER	Won't run without answer	didn't run
Entering with spaces	1 2 on team 1	Works	works
Entering invalid numbers	Rj space	Won't work	Wait for input

At this point I was excited that it was compiling and going through tests correctly, the tournament functions well without issue, and just like in the first game we ran, blue men dominate this game.