Project 4 Design Plan - Mariessa Logan

Requirements

Gameplay

This game is a **one-user-two-player** game, so one user should be able to play setup fighters for both teams. For simplicity, we will call the order of fighters in each team “lineup”, like a batting order in baseball or softball.

The **flow of gameplay** is as follows:

At the start of the tournament, it should prompt the user to **enter the number of fighters** for both team. After that, for each fighter, the game should ask the user to **choose the type of character, and enter a name for the fighter**. For example: assume the user chooses Harry Potter for player No.1’s team lineup, the user can name the Harry Potter character “Larry Snotter” or just “Harry Potter No.1”. The game should allow more than 1 of the same character for team lineups. For example, Team A has 3 members: 2 barbarians and 1 Harry Potter. The order of the lineup should be the same as the order user entered.

Function - start function

Pseudocode -

```
Ask user how many people they want in team 1
Ask user how many people they want in team2
Call Tourney class
```

After the user supplied the lineup of fighters in order, for Team No.1, and Team No.2, the tournament starts. In the tournament, the **fighter at the head/front of each lineup will battle**, in the same way they fight in project 3. The winner of each fight gets **put at the back of their lineup**, while **loser goes to the top of the loser container**(There should only be one containers to hold defeated fighters on both team).

If a fighter won, it may have taken damage, so the game should **restore some percentage of the damage** they took when they get back in line.

The lineup order cannot be changed, meaning that the game cannot modify the order of the lineup except when putting winner at the back of lineup and loser at the top of the loser container.

The tournament finishes when one of the team does not have any more fighters to fight. When the game ends, it should **print the result of the game**, and allow user to **have the choice to display the contents of the loser pile**. More information is shown below in the menu section

Containers

Create structures:

Structure names: Team1 Team2, Losers

Data types: String for name, integer for choice, pointers to a next and previous nodes.

Reflection

I followed my design plan as closely as possible, but I did have to make some key changes. By the time i was done with the Tourney class, it had grown to quite a sizeable number of lines. I had to do some reworking to get the number of lines down, so it wasn’t too big. However, I was still getting seg-faults on my program, even after hours of reworking the program to figure out the issues with the pointers. It was extremely frustrating.