For our final project, we will be simulating a Pokémon game and battle. Our problem is selecting the correct moves and attacks to defeat the boss with one Pokémon. In our initial thoughts of the problem, we will need two classes. One class will be the Pokémon class and the other will be the Boss class. For the Pokémon class, each player will be defined display their health, attacks. Similarly, the Boss class would randomly choose one of the two bosses and display their health, attacks, and impact of attacks. The players and their health, attacks and impact of attacks are listed below:

* Mewtow: psychic.
* Gastly: ghost and poison
* Rhyhorn ground and rock
* Pikachu: electric
* Venomoth: bug and poison
* ALL POKÉMON HAVE A HEALTH OF 300. THEIR IMPACT OF ATTACKS WILL RANGE (WITH A RANDOM NUMBER GENERATOR) FROM 20-40.

The Bosses all have a health of 400 with impact of attacks at 8-28, their names and attacks are as follows:

* Persian
* Tauros

The user’s wins will be saved in file input/output if they have won at the end of the game. At the start of the program, the user will be presented with all of the possible Pokémon that he/she can choose from. The user will need to choose one Pokémon that they wish to fight with. After the Pokémon have been chosen, a boss will be randomly chosen from the Boss class. The user will always be the first to attack the boss. During the battle, the user will choose an attack of that Pokémon and then the Boss will reciprocate and so on. This will be implemented using a loop until either the Pokémon or the Boss run out of health. The game ends with either the Pokémon or boss runs out of health and then the win will be tallied in the file I/O if the user has won.

With the help from functions, loops, and potentially pointers, we will be able to keep track of the health and the impact of the attacks on the players. We are also looking at setting up a file to save the amount of wins from a battle, so if the user can see how many game they have won. It will be challenging to fit all of these moving parts together and synchronize them in a way to work properly. We have not made any assumptions; we are still working on understanding what is needed to have our idea transfer appropriately into our code.

|  |
| --- |
| Pokémon |
| * hp : int   - attackPower: int |
| + getHealth: int  + setHealth ( health: int) : int  + getAttack: int  + setAttack: int  + showHp(hp:int, enemyattack:int): int  + Pokemon () |

|  |
| --- |
| Boss |
| * enemyhp:int * attackPower:int |
| + getHealth : int  + setHealth : (health : int) : void  +setAttackPower: void  + getAttackPower: int  + Boss()  +showEnemyHp (enemyhp:int, attack:int): int |