**Individual Report For**

**Turn-based battle game: FT DICT 422**

**by Kaung Myat San (Group 4)**

[**8580ls1a@psba.edu.sg**](mailto:8580ls1a@psba.edu.sg)

**Date: 22.11.2022**

**For the first time in my life, I wrote code systemically and created a game with my groupmates. It was fun to build that game together. I made a lot of memories with them. On the first day, since we were too far away, we worked online via messenger group video call. And day after day till 22nd Nov, we solved problems together. The one that I will never forget is we had to hardly figure out how to create AI with random module. It is easy in fact, but it was 1:00 and we were out of stamina and our brains weren’t functioning well and when we eventually solved it, we laughed and blamed ourselves that we should have done it easily.**

**Here are some of my snippets from group work. I mainly give effort in the part of manipulating player units.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**In source code Line 55:**

**while len(player\_unit) < 3:**

**unit\_type = input("Choose your champion..[Warrior(W) or Tank(T)] or Sorcerer(S). or**

**Return to main menu(R). \n\t->").upper()**

**if unit\_type == "W":**

**while True:**

**unit\_name = input("Give your champion a name. \n\t->")**

**if unit\_name not in unit\_namelist:**

**unit\_namelist.append(unit\_name)**

**troop = game.Game(unit\_type, unit\_name)**

**player\_unit.append(troop)**

**break**

**else:**

**print("\tTroops can't have Same name!")**

**elif unit\_type == "T":**

**while True:**

**unit\_name = input("Give your champion a name. \n\t->")**

**if unit\_name not in unit\_namelist:**

**unit\_namelist.append(unit\_name)**

**troop = game.Game(unit\_type, unit\_name)**

**player\_unit.append(troop)**

**break**

**else:**

**print("\nTroops can't have Same name!")**

**elif unit\_type == "S":**

**while True:**

**unit\_name = input("Give your champion a name. \n\t->")**

**if unit\_name not in unit\_namelist:**

**unit\_namelist.append(unit\_name)**

**troop = game.Game(unit\_type, unit\_name)**

**player\_unit.append(troop)**

**break**

**else:**

**print("\nTroops can't have Same name!")**

**elif unit\_type == "R":**

**return main()**

**else:**

**print("Invalid!")**

**if anger == 1:**

**print("Choose(W) or (T) or (S) Only!!!!")**

**elif anger == 3:**

**print("Cmon man! Choose from(W) or (T) or (S)")**

**elif anger == 5:**

**print("You are banned!!!!")**

**while len(player\_unit) != 0:**

**player\_unit.pop(0)**

**return main()**

**anger += 1**

**print("::::::::::::::::::::::::::::::::::::::::::::::::::::::\nYour Champions are here!!!!!!")**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_The codes above,**

* **Considering player input: input apart from (types of units [w, t, s], R(return to main menu)) will be neglected and output Invalid**
  + **Input name duplication : will not be allowed**
  + **Anger parameter: once an input got wrong, anger(variable) value +1**
  + **Anger point 1 = remainder with exclamation mark**
  + **Anger point 3 = Cmon! Man**
  + **Anger point 5 = banned from the game and returned to main menu and all data of remaining player\_unit would be removed**
* **After all units are set up:**
  + **Will print out all player units stats**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**In source code Line 144:**

**if rounds % 2 == 1: #odd**

**while loop == 0:**

**attunit\_ipt = input("Choose ur unit [1, 2, 3] : ")**

**print("++++++++++++++++++++++++++++++++++++++++++++++++++++++\n")**

**if attunit\_ipt.isdigit() == True:**

**attunit = (int(attunit\_ipt))-1**

**if attunit in range(len(player\_unit)) :**

**while loop1 == 0:**

**target\_ipt = input("Choose ur target unit [1, 2, 3] : ")**

**print("++++++++++++++++++++++++++++++++++++++++++++++++++++++\n")**

**if target\_ipt.isdigit() == True:**

**target = (int(target\_ipt))-1**

**if target in range(len(enemy)):**

**#player\_unit[attunit].attack(enemy[target])# Player attack to AI**

**filename.write(str(player\_unit[attunit].attack(enemy[target]))+ "\n\n")#**    **GameLog**

**filename.write(str(troop.exp\_gain(player\_unit[attunit], enemy[target]))+ "\n\n")#**  **Exp method #GameLog**

**filename.write(str(troop.rank(player\_unit[attunit], enemy[target]))+ "\n\n")#**  **Rank method ##GameLog**

**loop1 = 1**

**else:**

**print("Choose Remaining Unit!")**

**else:**

**print("Invalid:")**

**loop = 1**

**else:**

**print("Choose remaining unit!")**

**else:**

**print("Invalid:")**

**for i in enemy:**

**filename.write(str(troop.hp\_out(enemy, i))+ "\n\n")**

**for j in range(len(enemy)):**

**print(enemy[j])**

**rounds += 1**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_The codes above,**

* **Considering again the input: (-1) for the index so that input 1 2 3 will become 0 1 2 for index of list**
* **At that point, I encountered a problem that is other input aside integer won’t work with mathematical operation (-)**
* **So, I checked the input string with isdigit() method and if the user input not digits(1, 2, 3) it will output “Invalid” if the input digit it would be change to int type and –1**
* **The same way applied for the target unit........**

**And of Course, attack methods trigger and print out Game Message and unit remaining in list..**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Personal Learning Reflection Report**

**Since the program we had to develop was a game, it has median difficulty and due to our beginner skill, we encountered a lot of challenges. We strive to solve various problems with the skills we have. May be some problems could be solved easily, but we solved with harder way. Whenever we had to write a single line of code, we consulted together and chose opinions from each member and considered what would be a good choice. When we got an idea, we started to create smart code on code editor, at that point we faced with a lot of errors. But we could solve them by our cooperation. One of the critical incidents, when we were thinking and implementing AI smart targeting, we wasted 2 hours without any result, and I got my Shirt soaked with sweat. But when we could solve it, I would never forget such kind of happiness. At that point, I think my critical thinking skill and logical thinking skills might have improved. By looking at the code above, I think I got almost everything what IP module gives (By the help of Sir. Leong). Once problems seem could not be solved, I must figure out what kind solutions would work. At that time, I needed to revise what my teacher taught and if needed I had to learn more about relevant knowledge. I admit that I say Thanks to those difficulties because it makes me study more and it makes me get in touch with the lessons from the module. Where there is a will, there is a way. That is that......**

**After implementing our game. Happiness is the extra result that comes out together with Game. For the codes I mentioned above, they were written by me, but they contain a lot of advice from my teammates. All the codes in the source code are written by the combination of me and my groupmates. After each of a single problem was solved, we felt happy, and that kind of feeling is precious for me. On the other hand, something came into my mind that is “how could I solve such a problem! ". And I felt proud of myself, and I got potential that I could do better in the future. As a result, from the accomplishment of game, I got close friends and we promised that after submitting our work to my teacher, we won’t stop implementing the game and we would develop that game to a kind of level in the future.**

**From the side of mine, yes! I have learnt a lot from this module. And I applied what I have learnt in this turn base game development. Lab exercises are helpful to me. They recalled what I forgot about lessons in module. As the name of the book is “starting out with python”, I could use the basics of python in my future. Data manipulation is the one I am interested in from the module, ways to manipulate lists, tuples, dictionaries, etc.... are interesting for me. But I still have a lot to learn, and I have many weaknesses. Although I feel proud of myself, I still feel bad that my codes are long. I need to learn more and more to write code shorter and more efficiently. Sometimes, I felt guilty when we searched for shorter codes on the internet and copied to our source code. So that, I deleted that code and I try to encourage my team not to copy and even though our codes are long and not good, it is always better than copying others works.**