Homework 4 (major assignment: 10%)

Video demo link: https://youtu.be/01iEdhC3rss

Part 1 Task Description

In this assignment, you are given a **map** (see Figure 1) that contains various **nodes** (e.g., A, B, C, ….). Using this **map**, you are asked to travel from the **starting point** (A) to **the destination** (E). There are many possible routes that you can travel from the start (A) to the end (E). To travel on the map, you need to have a good **stamina**. In the beginning, you are given a full stamina cost, which is **25**. To be able to travel from one node to another, you need to **use up** your stamina. Based on which route you have chosen, the **cost** for the stamina will be **different**. For instance, if you travel from node *A* to *K*, the cost for the stamina is *3*, while the stamina cost from node *A* to *J* is **7**. The stamina **costs** for different routes are given in **Table 1**. While traveling on the map, if you have already **used up all your stamina**, the game will be **over**, hence, you fail in this attempt. If you arrived at the destination and you still have the stamina, then you **win** the game, and the game is ended successfully. Ideally, at the end of the game, the **higher** score you have, the **higher** achievement you gain in this game.

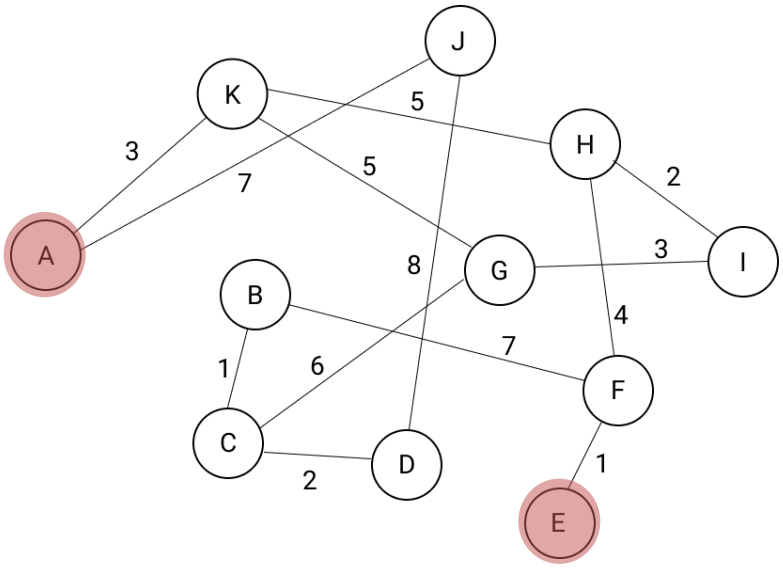

Figure 1. Map

Stamina Cost

A ⇔ K : 3 *A ⇔ J : 7* *K ⇔ H : 5* *K ⇔ G : 5* *J ⇔ D : 8* *H ⇔ I : 2* *H ⇔ F : 4*	*G ⇔ I : 3* *G ⇔ C : 6* *F ⇔ E : 1* *F ⇔ B : 7* *B ⇔ C : 1* *C ⇔D : 2*

Table 1. Stamina Cost

Implementation Requirements

- You need to implement it from scratch.
- You can apply the know-how you have gained from the previous lectures such as string, if-else, list, function, iteration, and dictionary.
- You are not allowed to use external libraries.
- You need to submit it to MyCourseVille.
- The codes must be yours, meaning that you should not copy any parts of your program from others, or you should not share your codes with others. There is a penalty applied for both parties.
- You need to make sure you submit it by the deadline; NO late submission is graded for the major assignment (HW1-HW4) since they are considered as take-home midterm exam.

Scoring Criteria (Full Score = **7**)

- Implement all nodes, routes, and correct stamina costs according to the given diagram (see Figure 1) (0.5)
- Game initialization that shows the full stamina, the current node, and route options & stamina costs for the next destination (1.5)
- User input that shows where the player wants to move next and the program should not accept the invalid input (e.g., case-sensitivity) (1)
- Update the game status that shows the correct current stamina, the current node, the route the player has traveled, and route options & stamina costs for the next destination (2)
- Game-Over that shows the total stamina left (i.e., stamina should be zero), the route the player has traveled, and a game-over message (1)
- Success status that shows the total stamina left, the correct route the player has traveled, and congratulatory message (1)

Output Examples

Game Initialization and Player's routes

```
Python 3.7.9 (bundled)
>>> %Run main.py
 Game has started.....................
 You have a full stamina of 25 ......
 You are currently at :A
 Choose your next destination........
 K: 3
 J: 7
 K

 Your current Stamina is : 22. You have used up 3 points from the total stamina.
 Your current route is : A >> K
 You are currently at :K
 Choose your next destination........
 A: 3
 H: 5
 G: 5
 H

 Your current Stamina is : 17. You have used up 5 points from the total stamina.
 Your current route is : A >> K >> H
 You are currently at :H
 Choose your next destination........
 K: 5
 I: 2
 F: 4
 F

 Your current Stamina is : 13. You have used up 4 points from the total stamina.
 Your current route is : A >> K >> H >> F
 You are currently at :F
 Choose your next destination........
 H: 4
 E: 1
 B: 7
```

Success

```
Your current Stamina is : 12. You have used up 1 points from the total stamina.

CONGRATULATIONS!! YOU REACHED THE DESTINATION.

Your stamina is 12
You have travelled : A >> K >> H >> F >> E
```

Fail

```
--

OUT OF STAMINA. BETTER LUCK NEXT TIME.

Your stamina is 0
You have travelled : A >> J >> A >> J >> A
```

Part 2 Task Description

In this assignment, you are given a CSV file that contains a list of students and their scores for the individual subjects, as well as the total scores. There are some incorrect total scores in the given file. To retrieve a list of students whose total scores are incorrect, you are required to create a python program and display a list of students whose total scores are incorrect, as well as score differences between the actual score and incorrect total score, as shown in Figure 2.

Implementation Requirements

- You need to create a Python program from scratch.
- You must use the **Numpy** array and its functions only.
- NO LOOP is allowed for the total score calculation; You can use only Numpy to calculate the total score.
- You are not allowed to use external libraries other than Numpy & (CSV if required).
- You need to submit it to MyCourseVille.
- The codes must be yours, meaning that you should not copy any parts of your program from others, or you should not share your codes with others. There is a penalty applied for both parties.
- You need to make sure you submit it by the deadline; NO late submission is graded for the major assignment (HW1-HW4) since they are considered as take-home midterm exam.

Scoring Criteria (Full Score = 3)

- Read the CSV file and process it correctly (0.5).
- Use the Numpy array and its functions to produce the desired outputs shown in the image below (1).
- Display the success message if all total scores are correct (0.5).
- If total scores are incorrect, display the list of students whose total scores are wrong with differences, incorrect total score, and the actual total score (1).

Output Sample

```
The following students' total scores are incorrect.
**************************************************
Jones's score is 58.0 different from the actual total score.
The incorrect total score is 211.0.
The actual total score should be 153.0.

Klein's score is 2.0 different from the actual total score.
The incorrect total score is 179.0.
The actual total score should be 177.0.

Lopez's score is 8.0 different from the actual total score.
The incorrect total score is 221.0.
The actual total score should be 229.0.

Mason's score is 7.0 different from the actual total score.
The incorrect total score is 201.0.
The actual total score should be 208.0.

Nalty's score is 5.0 different from the actual total score.
The incorrect total score is 180.0.
The actual total score should be 175.0.

Ochoa's score is 14.0 different from the actual total score.
The incorrect total score is 198.0.
The actual total score should be 184.0.

Patel's score is 6.0 different from the actual total score.
The incorrect total score is 221.0.
The actual total score should be 215.0.

Quinn's score is 22.0 different from the actual total score.
The incorrect total score is 167.0.
The actual total score should be 189.0.
```

When All scores are correct, the output should be displayed as follows:

```
############################
The total scores are correct.
############################
```

Submission

- You have to rename your file to be HW4_{ID_FirstName}_Task1.py and HW4_{ID_FirstName}_Task2.py; then zip them into HW4_{ID_FirstName}.zip. Finally, submit the zip file into MyCV.
- For example, if your id is 6438012521 and your first name is Kridbhume.
    - HW4_6438012521_Kridbhume_Task1.py
    - HW4_6438012521_Kridbhume_Task2.py
    - HW4_6438012521_Kridbhume.zip
- Remind:
    - You need to make sure you submit it by the deadline; NO late submission is graded for the major assignment (HW1-HW4) since they are considered as take-home midterm exam.
    - The codes must be yours, meaning that you should not copy any parts of your program from others, or you should not share your codes with others. There is a penalty applied for both parties.