

Final Assessment – Project Work

(50 total points)

(To be completed in groups (of up to three members). Please do submit your own original work. Should your work closely resemble some other groups or contents on some random page(s) on the Internet, you will be given an opportunity to explain the resemblance during your code review, and in case you fail to do so, you will be awarded a straight zero for that particular piece of work.)

Question 1 (3 X 5 points = 15 points)

In this question, we are trying to build a simple e-Wallet system which can handle basic monetary transactions and write the data accordingly into the respective **csv** files. There are two **csv** files which the system uses:

- a. **account_transactions.csv**, which contains all the transaction accounts' information.
- b. **user_records.csv**, which contains all the registered users' information.

There are three main Python classes (*Customer*, *Account*, *NewAccount*) which run this system, two of which (*Customer*, *NewAccount*) are already written completely and are included in the **wallet.py** file.

There is a **account.py** file included too, which contains partially completed code for the *Account* class.

For convenience, there is a *Jupyter Notebook* file included (**Operations.ipynb**), which is a duplicate of **account.py** file, laid out in a simplistic manner for you to get started. This is the file where you will be working on!

You must complete these three methods in the *Account* class contained in the given *Jupyter Notebook* file (or **account.py** file directly if you wish to do so!). The methods are **deposit**, **withdraw**, and **transfer_funds**.

In addition to performing the withdraw, deposit and fund transfer operations successfully, your methods also need to update the two csv files- which store the data- with the newer values.

[Note: For every method you write successfully, you get 5 points, so a total of 15 points for 3 methods.]

Question 2 (5 X 3 points = 15 points)

During our class, we have seen how NumPy is a great library that provides optimized functionality for different forms of array-based operations. In this question, we aim to understand one popular real-world application of NumPy, which is image manipulation. Please go through the included *Jupyter Notebook* file, and solve the following questions using NumPy:

- a. Slice an image vertically into half.
- b. Flip the image sideways (mirror copy).
- c. Plot just the green channel i.e., a green version of the image.
- d. Crop the 2D image into 80 percent of its original size, cropping uniformly from all sides.
- e. Can you recover the original image by removing the noise from the noisy image (which was obtained by taking a dot product with our noise matrix)? If yes, show the result. If no, please explain why that is not possible.

Note: The above questions (a-e) make sense only in the context of the included *Jupyter Notebook* file, so please make sure to go through it.

Question 3 (3 X 5 points = 15 points)

The file ***tourist_arrival.csv*** shows the number of tourists that visited Nepal each month from the year 1992 to the year 2013. Read the given file.

- a. Make a **bar plot** for the number of tourists that arrived each month (*Jan - Dec*). Make sure to take the average value from all the years (1992-2013).
- b. The seasons are defined as follows: Spring (*March, April, May*), Summer (*June, July, August*), Autumn (*September, October, November*), Winter (*December, January, February*). From the data, can you predict the **most popular season** for tourist arrival as well as the **least popular season**.
- c. Finally, can you predict the **five worst months** from the data where the tourist arrival was at its lowest.

[Hint: Your answer should be set like: June 2001, July 2002, ...]

Note: The remaining **5 points** will be given if the application has desirable features, appropriate input validation, well-commented code, modular design, and appropriate error handling.