SHENKAR שנקר

ENGINEERING. DESIGN. ART. הנדסה. עיצוב. אמנות

Final Project - Scientific computing & Algorithms with Python

Lecturer: Or Perets

T.A: Yossi Jerbi

Overview

This is your final project on advanced Python course!

After deep understanding of Data Structures (Stack, Queue, Linked-List, Hash-Table & Graphs), Data Science and little bit of Machine Learning, you are ready to build your main project.

All tools are allowed here! New libraries, internet, friends :)

Goals

1. Check your learning progress on scientific computing topics.
2. Use your abilities to build end-to-end project with python.
3. The most important: Have Fun!

Deadline

29/7/2019 - 08:30.

Requirements

I. Group size can be single or pairs only.

Each group will contains single student or pair.

- Recommendation: do it alone.

II. Report

Your group should write a report about your progress and results.
The report should be written in **English** and should contains **screenshots** of your results.

GOOD LUCK!

Part A - Data Structures and Algorithms

Q1

Write the class “SpecialQueue”.

“SpecialQueue” represent **Queue** data structure implemented by **Linked List.**

The class should contains the following methods:

function	description
is_empty(self)	Return True if the queue is empty, else False.
size(self)	Return the size of data structure.
enqueue(self, x)	Add x to data structure.
dequeue(self)	Return the first item in data structure.

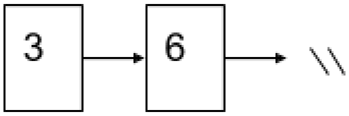
Guidelines

1. You are implements Queue, but from backend its Linked-List.
   Be aware of: insertion side & delete side.
2. Write the first function, check - and then you can go on.
   Don’t write all of your code at once. Do it step by step.

Code Example

```
sq = SpecialQueue()
sq.enqueue(3)
sq.enqueue(6)
```

```
# Current Linked List
```



```
print(sq.is_empty())   # False
print(sq.size())  # 2
```

Q2

Let A, B be lists with integers values which includes n elements.

You know the A is **unsorted** and B is **sorted**.

A = [a1, a2, …. , aN]

B = [b1, b2, …., bN]   **where**   b1 <= b2 <= … <= bN

1. Write “join_lists(A, B)” function.

   The function gets your 2 lists and return third list C.

   C includes all items that appears in A **and** B.

   Notice: Do not convert the lists to sets, Do not use “in” operator.

   Complexity Time: $O(n^2)$

2. Write “smart_join_lists(A, B)” function.

   The function returns the same input as (1).

   Notice: Do not convert the lists to sets, Do not use “in” operator.

   Complexity Time: $O(n \cdot logn)$

Part B - Data Analysis

You have 3 tables for this part: **flights, airports and airlines.**

Tip: Know your data, open the tables and read few rows just to be familiar with it.

Tables columns

Table	Column	Description
Airlines	IATA_CODE	Airline identifier
	AIRLINE	Airline`s name
Airports	IATA_CODE	Airport identifier
	AIRPORT	Airport`s name
	CITY	-
	STATE	-
	COUNTRY	-
	LATITUDE	Latitude of the airport
	LONGITUDE	Longitude of the airport
Flights	YEAR	-
	MONTH	-
	DAY	-
	DAY_OF_WEEK	Integer represent day of week. 1 is Monday, 2 is Tuesday and so on.
	AIRLINE	Airline identifier
	FLIGHT_NUMBER	Flight identifier
	TAIL_NUMBER	Aircraft identifier. Includes characters and integers.

	ORIGIN_AIRPORT	-
	DESTINATION_AIRPORT	-
	SCHEDULED_DEPARTURE	Planned Departure Time
	DEPARTURE_TIME	Integer number with 4 digits: HH:MM
	DEPARTURE_DELAY	Total Delay on Departure
	AIR_TIME	Actual flight air time
	DISTANCE	Distance between two airports
	ARRIVAL_TIME	-
	ARRIVAL_DELAY	-
	CANCELLED	Flight Cancelled (1 = cancelled)
	CANCELLATION_REASON	A - Airline/Carrier, B - Weather, C - National Air System, D - Security

Q1 - Intro

1. Print the number of rows in flights table.
2. Print the name of each column in flights table.
3. Repeat (1), (2) on airports table.
4. Repeat (1), (2) on airlines table.
5. Write conclusions: what is the **entity relationship** between the models?
   [You can add ERD (entity relationship diagram) if you want].

Q2 - Delayed flights

1. How many flights has been delayed in 2015 ?
2. What is the day with the most delayed flights in 2015 ?
   a. Display day name and not day number.
3. Find the largest distance between 2 airports.
   Print the following details:
   a. Distance.
   b. Airports full name.
   c. Day of the week.
      i. If there is more than one, print all.
      ii. Remember to convert number to string.
4. Display all tails numbers of “JetBlue Airways” that has been delayed in 2015.

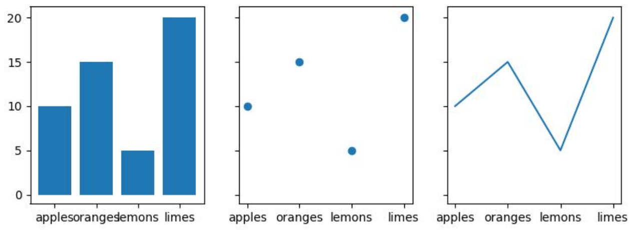
Q3 - Canceled flights

1. How many flights has been canceled in 2015 ?
2. What is the most common reason for flight cancellation?
   a. Plot bar diagram and proof your answer with comparison between common reasons.
3. What is the percentage of canceled flights across all flights ?
4. From all canceled flights, find the longest & shortest route.
   Print your answer with the following statement:
   print(“Shortest path: from {} to {}”.format(origin_airport, dest_airport))
   print(“Longest path: from {} to {}”.format(origin_airport, dest_airport))
   a. Your answer should include airport **full name.**

Q4 - Your own part

Let's assume that you are trying to get job on “Google”.
They like you but part of the interview is to prepare small presentation.

You need to describe your flights results (from Q1, Q2, Q3) with graphs and visualization.
Your final plot should be done with subplots, like:



(just for example)

Requirements

1. Title.
2. Visualization.
3. Labels & Ticks.
4. Legend (if needed).
5. Informative data.
6. **Creativity!**

Part C - Data Science & Business Intelligence

On this part, you need to build K-Nearest-Neighbors Classifier for Titanic problem.

Your dataset is “titanic_data.csv” and the file includes the following columns:

1. Pclass - Class of travel.
2. Sex - male / female.
3. Age
4. SibSp - Number of sibling / spouse aboard.
5. Parch - Number of parents / child aboard.
6. Fare
7. Embarked - The port each passenger has embarked.
8. Survived - 1/0 (True/False).

Your assignment is to build KNN Classifier for this problem and find the **optimal K.**

K value can be 1 up to (data length / 2).

Print your answer with the following statement:

```
print(“Best K: {}, Best Accuracy: {}”.format(optimal_k, acc))
```

Question

Why we do not want to use K=data-length?

Bonus (10 points)

Try to implement KNN algorithm yourself, without sklearn package.
Tip: split your code to small functions and split the responsibility.
If you want to implement “Bonus”, please talk with me to get some advice.