**(S2-18_DSECFZG519)**
(Data Structures and Algorithms Design)
Academic Year 2018-2019

Assignment 2 – PS3 - [SHIPMENT PROBLEM] - [Weightage 12%]

1. Problem Statement

Amazon Warehouse in Bangalore was expecting a freight shipment at 10 am on a Monday. However, the company responsible for the shipment mistook the destination as Amazon Development Centre, Bangalore. In order to save the extra cost of transportation back to the warehouse, the warehouse manager wants to help them find the fastest route from the DC to warehouse. He has the location map available which is given below (the weightage given is the distance in km between each location). However, as he did not take DSAD course he turns to you for help.
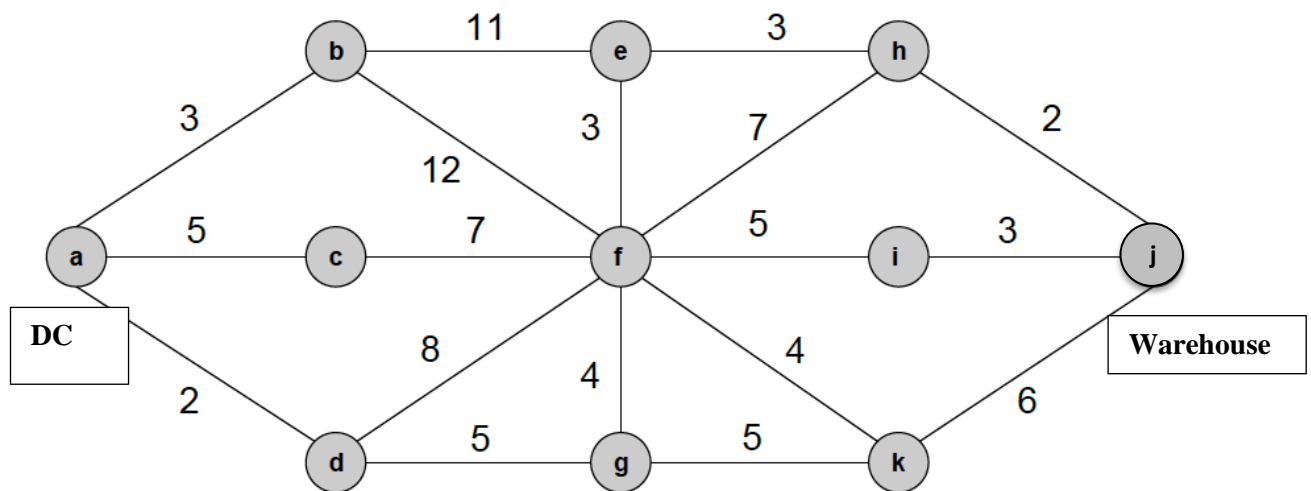
Your job is to help the manager with the following queries.

1. Which is the shortest route to reach Warehouse from the DC? The warehouse and DC should be taken as an input and is not fixed to the nodes mentioned in the graph above.

2. Find out the expected time of arrival of the shipment at the warehouse if the truck speed is 60 km/hr.

Requirements:

1. Formulate an efficient algorithm to perform the above task.
2. Provide a description about the design strategy used
3. Analyse the time complexity of the algorithm and show that it is an "efficient" one.
4. Implement the above problem statement using Python

Input:

Input should be taken in through a file called "inputPS3.txt" which has the fixed format mentioned below using the "/" as a field separator:

<location 1> / <location 2> / <distance in km>

Ex:

a / b / 3
a / c / 5
a / d / 2
…
DC Node: a

WH Node: j

Output:

Shortest route from DC 'a' to reach Warehouse 'j' is [a d g k j]
and it has minimum travel distance 18km
it will take him 18 minutes to reach
Expected arrival time at the warehouse is 10:18am

Display the output in **outputPS3.txt**.

2. **Deliverables**

- Word document **designPS3_<group id>.docx** detailing your algorithm design and time complexity of the algorithm.
- **Zipped AS2_PS3_SP_[Group id].py package folder** containing all the modules classes and functions for the employee node, binary tree and the main body of the program.
- **inputPS3.txt** file used for testing
- **outputPS3.txt** file generated while testing

3. Instructions

- Do not use inbuilt data structures available in Python. The purpose of these assignments is for you to lean how these data structures and algorithms work.
- It is compulsory to use Python for implementation.
- Ensure that all data structure insert and delete operations throw appropriate messages when their capacity is empty or full.
- For the purposes of testing, you may implement some functions to print the data structures or other test data. But all such functions must be commented before submission.
- Make sure that your read, understand, and follow all the instructions
- Ensure that the input and output file guidelines are adhered to. Deviations from the mentioned formats will not be entertained.
- Run time analysis is provided in asymptotic notations and not timestamp based runtimes in sec or milliseconds.

4. Deadline

- The strict deadline for submission of the assignment is **Sep 5th, 2019 EoD.**
- Late submissions won't be evaluated.

5. How to submit

- This is a group assignment.
- Each group has to make one submission (only one, no resubmission) of solutions.
- Each group should zip the deliverables and name the zipped file as below
- "ASSIGNMENT2_[BLR/HYD/DLH/PUN/CHE]_[B1/B2/…]_[G1/G2/…].zip"
- and upload in CANVAS in respective location under ASSIGNMENT Tab.
- Assignment submitted via means other than through CANVAS will not be graded.

6. Evaluation

- The assignment carries 12 Marks
- Grading will depend on
    - Efficiency of design (detailed in the design document)
        - Generic explanation copied off the internet will not be considered.
    - Every bug in the functionality will lead to negative marking.
    - Duplication of design document / code will be penalized.
    - Source code files which contain compilation errors will get at most 25% of the value of that question.
    - Fully executable code with all functionality.
- Late submissions will not be evaluated.

7. Readings

Text book: Algorithms Design: Foundations, Analysis and Internet Examples Michael T. Goodrich, Roberto Tamassia, 2006, Wiley (Students Edition). Chapters: 7.1, 7.2, 7.3