**Assignment 3. Transportation system**

Given a set of n cities, we would like to build a transportation system such that there is some path from any city to any other city. There are two ways to travel: by driving or by flying. Initially all of the cities are disconnected. It costs to build a road between city i and city j. It costs to build an airport in city i. For any two cities i and j, we can fly directly from i to j if there is an airport in both cities.

Design a O(mlog(n))-time algorithm for determining which roads and airports to build to minimize the cost of connecting the cities. Here, “connecting the cities" means that there should be some way to get from any city to any other. Your algorithm should take as input the costs and , and return a list of roads and airports to build.

1. **Introduction**.

Your assignment is to build the program to solve the above problem. For that, we will provide you a skeleton code for handling input arguments and sample data.

Inside of skeleton code, there exist function named development(). You need to implement this function with O(mlog(n)) as mentioned above.

**In this assignment you need to submit report along with codes.** In your report, you need to justify why your algorithm and implementation is correct.

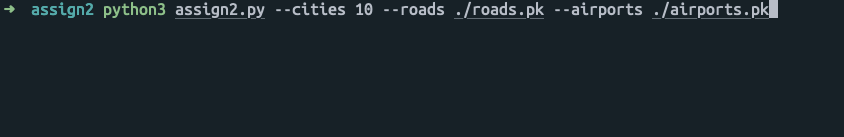
1. **Arguments**

When you execute the program there will be needed three arguments.

**--cities**: the number of cities

**--roads**: address of roads.pk which contains information about roads

**--airports**: address of airports.pk which contains information about airports



1. **Variables**

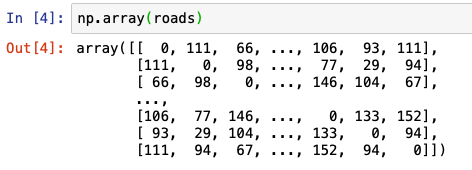
In this section, Short description for variables will be provided.

**cities** : the number of cities. Cities will be named with their index, which means there exist “city 0”, “city 1” ... “city n” (Starts with “0”)

**roads**: N by N symmetric matrix in form of list which contains cost for building a road between city a to city b.

(e.g. roads[0][2] : cost for building the road between “city 0” and “city 2”. )

(e.g. roads[i][i] : from “city i” to “city i” will contain 0. )



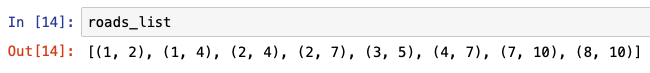
**airports** : 1D list which contains cost for building airports in each city.

(e.g airports[10] : cost for building the airport in “city 10”)

1. **Returns**

Function development() should returns list of roads and airport. With its return there exist a few rules

1. **roads\_list.**
2. road\_list is a list consists of tuples
3. If there exist road between “city i” and “city j”, road will be (i, j)
4. road(i, j) and road(j, i) are same. And only one of them should be in roads\_list
5. All roads(i,j) inside of roads\_list should be i < j
6. roads\_list should be sorted in ascending order



1. **airports\_list**
2. airports\_list is a list consist of int values
3. if “city i” need to build the airports airports\_list should contain “i”
4. airports\_list should be sorted in ascending order

개체이(가) 표시된 사진

자동 생성된 설명

1. **Additional information**

This section is not necessary for assignment.

In this assignment, for convenient data handling we use pickle modules.

Pickle is just one way to save data and re-loaded. And with given skeleton codes, tables information will be loaded correctly. If you want more information about pickle modules, go to <https://docs.python.org/3.6/library/pickle.html> .

1. **Requirements and Restriction.**
2. **No plagiarism at all. If we discover any plagiarism, we will regard all your assignment as 0. There will be no excuse at all.**
3. Given skeleton code is written with python3.6 . So this assignment will be restricted with **Python** with version above **3.6** .
4. Also inside of development() , you need to write **descriptive comments line by line**. Without proper comments, we will regard your assignment as 0.
5. **You need to submit a report to justify your implementation and algorithm.** Inside your report, you need to describe what kinds of algorithm you used, why that algorithm is correct, and why it’s time complexity is O(mlog(n)).
6. **Help**

If you need any help or have any questions about this assignment, feel free to contact us with e-mails.

강우성: [rkddntjddl@dgist.ac.kr](mailto:rkddntjddl@dgist.ac.kr)

김경택: [xor0925@dgist.ac.kr](mailto:xor0925@dgist.ac.kr)

1. **Submission**

Submit your codes in single compressed files which named as

“student-number\_name.zip” (e.g. 201912345\_GildongHong.zip) your name must be in English.

Must keep filename’s format (student-number\_name) and file type(.zip) otherwise, you will lose 30% score.

**Due date is *December 15st 11:59 PM in LMS***