Artificial Intelligence & Its Applications

Laboratory 01

Submission Due: 10-July-2020

Submission Procedure:
1. Use Pickle to save all variables into a file named "Lab01_XX_YY.py"
2. Compress the programs of each question into a file named "Lab01_XX_YY.zip"
2. Send "Lab01_XX_YY.py" and "Lab01_XX_YY.zip" to your monitor.

XX is your student ID.
YY is your first name + last name in small letter without spacing.

For example:
Chen Tao and ID 20190123456789
The file names should be
"Lab01_20190123456789_chentao.py" and
"Lab01_20190123456789_chentao.zip"

Networkx in Python is used in this laboratory.

Networkx
https://networkx.github.io

Installation
Command: "`pip install network`"

Graph Creation
https://networkx.github.io/documentation/stable/tutorial.html#creating-a-graph

Directed Graph Creation
https://networkx.github.io/documentation/stable/reference/classes/digraph.html?highlight=digraph#networkx.DiGraph

BFS and DFS
https://networkx.github.io/documentation/stable/reference/algorithms/traversal.html?highlight=bfs#

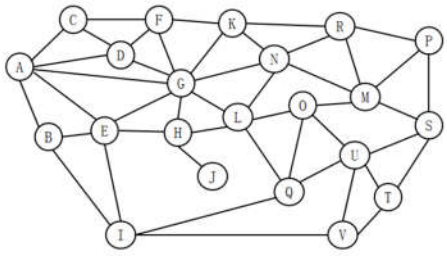
Shortest Path Algorithm
https://networkx.github.io/documentation/stable/reference/algorithms/shortest_paths.html

1. i) Store the following graphs in the format of Networkx.

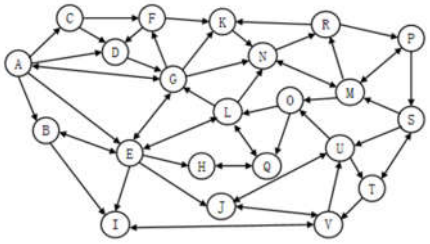
   ii) Show the sequence of visited edges using DFS and BFS for the following two graphs starting at node A.

   iii) Find the cost of the shortest path from node A to all nodes.

a)



b)



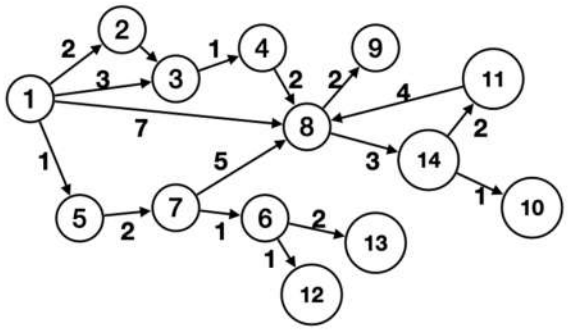
Answer Requirement:

1. Save the model of (a) and (b) in **`Q1_i_a`** and **`Q1_i_b`**
2. Save the sequences of (a) and (b) in **`Q1_ii_a_DFS`**, **`Q1_ii_a_BFS`**, **`Q1_ii_a_DFS`** and **`Q1_ii_a_BFS`** in this format:

   `[(1, 2), (1, 3), (3, 4), …]`

3. Save answer in **`Q1_iii`** in the format: `[(A, 10), (B, 20), …]`

2. Given the following graph, the weight of each edge represents the delay time of signal transmission. If we send a signal from node 1, what's the minimum time required for all nodes to receive the signal? If it's impossible, return -1.



Answer Requirement:

- Save the answer in **Q2** in the format: `[(1, 10), (2, 20), …]`

3. Count how many paths can reach the bottom-right cell from the top-left cell in the following maze. We are only allowed to move to cells downward and right, i.e. in the cell (i, j), we only can go to (i+1, j) and (i, j+1). The value of the cell in the maze is equal to 0 if it is an obstacle, otherwise the value is 1.

```
{{1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1},
 {1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1},
 {1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1},
 {0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0},
 {0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1},
 {0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 1},
 {1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1},
 {0, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1},
 {1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0},
 {1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1},
 {1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1},
 {1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0},
 {1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1},
 {0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1},
 {1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1},
 {0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0},
 {0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1},
 {1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1},
 {1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1},
 {0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1},
 {1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1},
 {1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0},
 {1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0},
 {1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0},
 {0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1}}
```

Example:

```
{{1, 0, 0, 1},
 {1, 1, 1, 1},
 {1, 0, 1, 1}}
```

Answer: 2

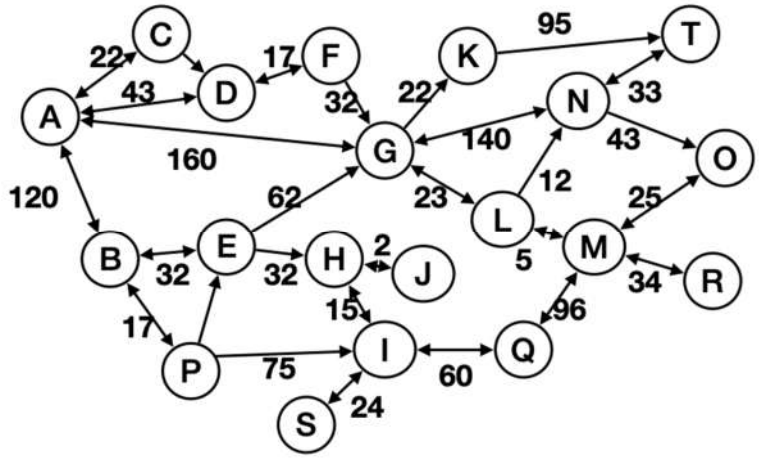
```
{{1, 1, 1, 1},
 {1, 0, 1, 1},
 {0, 1, 1, 1},
 {1, 1, 1, 1}}
```

Answer: 4

Answer Requirement:

- Save the answer in `Q3`

4. Given the train information of N cities with the following ticket prices, display the cheapest cost from each cities to every other city.



Answer Requirement:

- Save the answer in **Q4** in the format: `[(A, A, 10), (A, B, 20), …]`