# ACI Assignment 1
Question 1 A – Relief Supply Agent

Problem statement

Given below are the prominent cities of Ukraine. Now the country is in a crisis due to war, and as an expert in AI, you have been requested to help supply goods to various places using an intelligent agent. The picture below gives the map of the cities connected by paths that are safe to travel. You have been given the task to create an agent which can carry goods like medicines, food and other basic supplies to the required places and deliver them. The agent has restrictions of battery time and the agent is required to supply to all the places and return back to the starting point. You are required to find the shortest path for your work to go un-interrupted. The program should be able to take-in the start node dynamically from the user at run time. The edge costs depicted below is an approximation towards the transportation cost between any pair of cities in Ukraine. For heuristic design, consider all the possible paths between any arbitrary node n to the goal node (Starting point). The average of the total transmission cost across all these paths is the heuristic value h(n). Use the following search algorithms to find the minimum network connections and help the network agent.

a. Breadth First Search and
b. Uniform Cost Search
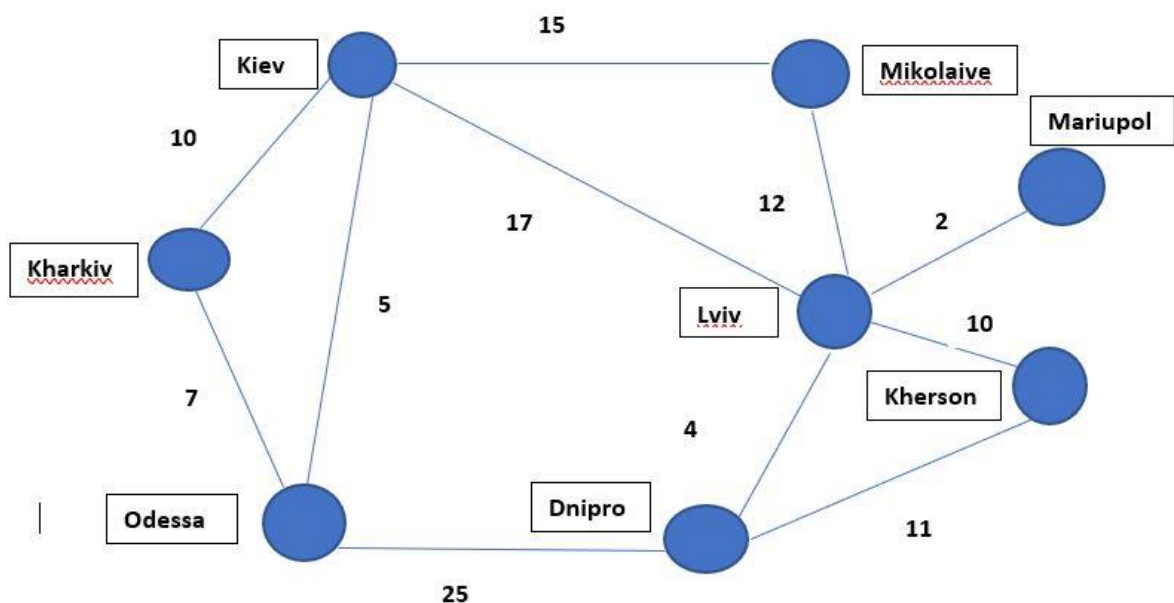
Evaluations will be based on the following.

1. Explain the PEAS (Performance measure, Environment, Actuator, Sensor.) for your agent. (20% marks)
2. Implement the given algorithms in PYTHON. The start node is to be obtained from the user at run time through prompt. (20% + 20% = 40% marks)
3. Print the minimum path taken by the agent that connects all the places. Print the cost of the path. (20% marks)
4. Include code in your implementation to calculate the space complexity and time complexity and print the same. For complexity calculations, refer to the prescribed text book for ACI. (20% marks)

Note 2:

- You are provided with the python notebook template which stipulates the structure of code and documentation. Use well intended python code.
- Use separate MS word document for explaining the theory part. Do not include theory part in the Python notebook except Python comments.
- The implementation code must be completely original and executable.
- Please keep your work (code, documentation) confidential. If your code is found to be plagiarized, you will be penalized severely. Parties involved in the copy will be considered equal partners and will be penalized severely.