National University of Computer & Emerging Sciences

AL2002 – Artificial Intelligence – Lab (Spring 2024)

BSCS-6B

Lab Work 7 (Hill Climbing Search, Simulated Annealling, Local Beam Search)

Lab Instructor	Momna Javaid
Department	Computer Science

Instructions:

1. You also have to submit .ipynb file.
2. Comments in the code explaining chunks of the code are important.
3. Plagiarism is strictly prohibited, 0 marks would be given to students who cheat.

Lab Tasks:

Task 1:

A salesman is required to visit a number of cities during a trip. The salesman needs to find the shortest route that takes him to each destination exactly once. With a map of streets and locations, the challenge is to plan the optimal route, minimizing travel time and distance. To tackle this challenge, the salesman decides to use any efficient algorithm.

Representing the cities by vertices and roads between them by edges there is a graph in figure 1.

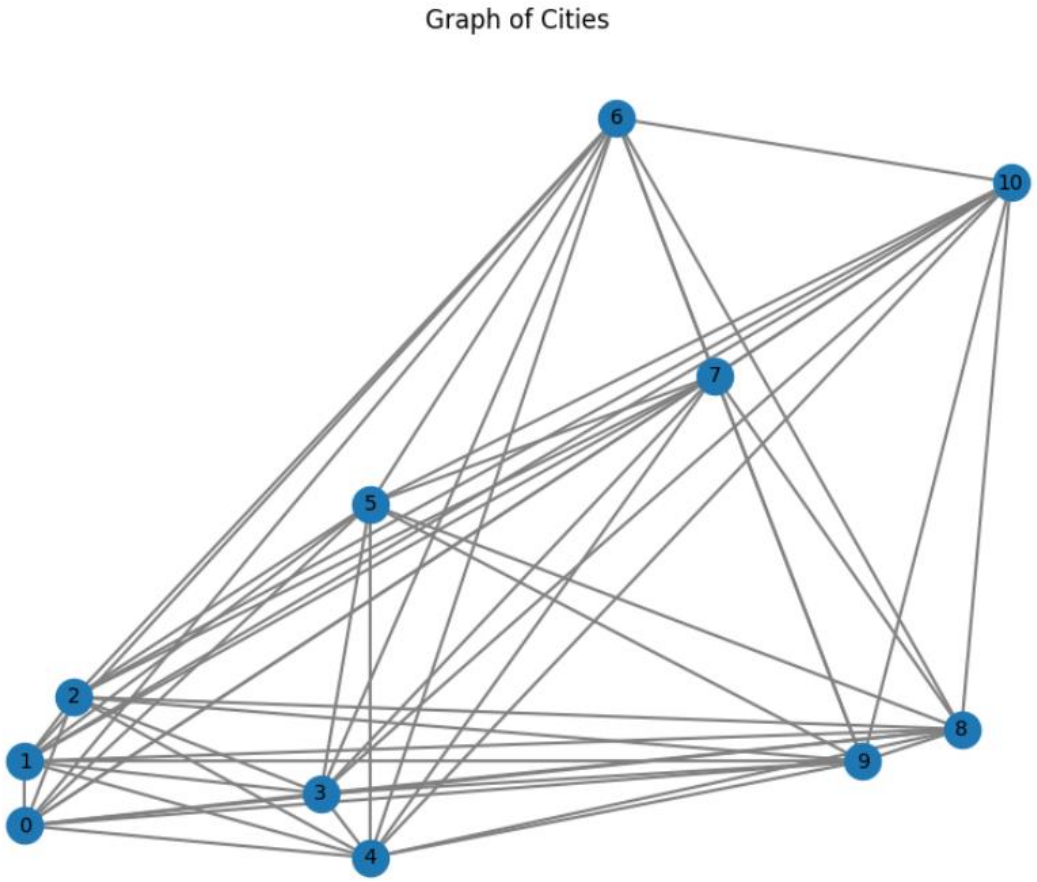

Figure 1. Graph of Cities

For this task you should create an array using numpy library, it should be a 2D array, the values should be:

```
# Coordinate of the points/cities
coordinate = np.array([[1, 2], [1, 4], [2, 6], [7, 3], [8, 1], [8, 12],
                       [13, 24], [15, 16], [20, 5], [18, 4], [21, 22]])
```

The graph in figure 1 is generated from these values. These values are the latitude and longitude for the cities' location.
Use this graph to solve the Travelling Salesman Problem using simple Hill Climbing, Hill Climbing variants (First Search, Random Restart, Stochastic), also apply simulated annealing and local beam search to analyze the difference between these search algorithms.
Calculate time and space complexity of each algorithm.
Find the complete path and create a graph for this path using networkx library.