

## Practical Class Exam Sample Question for Programmers : Neural Network and A\* Search Combined

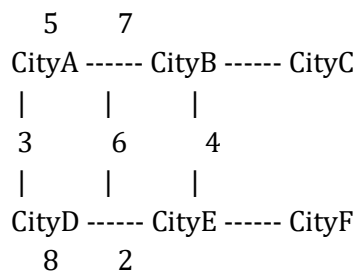
This practical question focuses on combining a simple feedforward neural network and the A\* search algorithm to solve a shortest path problem in a graph. You are required to use Python or any other programming language to implement both components and demonstrate their integration.

### Scenario

You are provided with a weighted graph representing a network of cities, where nodes are cities and edges are roads with weights representing distances (in kilometers). Your task is to implement a neural network to predict heuristic values for each city (straight-line distances to the goal city), and use these predictions in the A\* search algorithm to find the shortest path.

### Graph Representation

Consider the following graph:



### Dataset

The following table contains the heuristic values for each city, representing straight-line distances to the goal city (CityF). This dataset should be used to train the neural network.

City	Heuristic Value (h_manual)
CityA	12
CityB	10
CityC	4
CityD	15
CityE	7
CityF	0

## Tasks

### 1. **Neural Network Implementation:**

- Create a simple feedforward neural network using Python (e.g., TensorFlow or PyTorch).
- Train the neural network using the provided dataset to predict heuristic values ( $h(n)$ ) for cities.
- Test the neural network to ensure it can predict heuristic values for any city.

### 2. **A\* Search Algorithm Implementation:**

- Implement the A\* search algorithm in Python.
- Use the neural network's predicted heuristic values as input to the algorithm.
- Calculate the shortest path from CityA to CityF using the A\* algorithm.

### 3. **Integration and Testing:**

- Integrate the neural network and the A\* search algorithm.
- Test the integrated system with the provided graph to find the shortest path from CityA to CityF.
- Document the steps, the chosen path, and the total cost.