

# Lab Task 4

## UCS and Greedy Search

### Task 1: Network Routing

In a computer network, data packets must be transmitted efficiently from a source server to a destination server. Each link between routers has a different transmission cost depending on factors such as bandwidth, latency, congestion, or link quality. Your goal is to determine the most cost-efficient route for the data packet to travel from the source to the destination.

Problem Setup: The network can be modeled as a graph where:

Nodes represent routers in the network. Edges between nodes represent network links, with associated transmission costs. These costs reflect the real-world constraints, such as available bandwidth, latency, or congestion level.

Router 1	Router 2	Transmission Cost
A	B	4
A	C	2
B	D	3
C	D	1
C	E	7
D	F	5
E	F	3

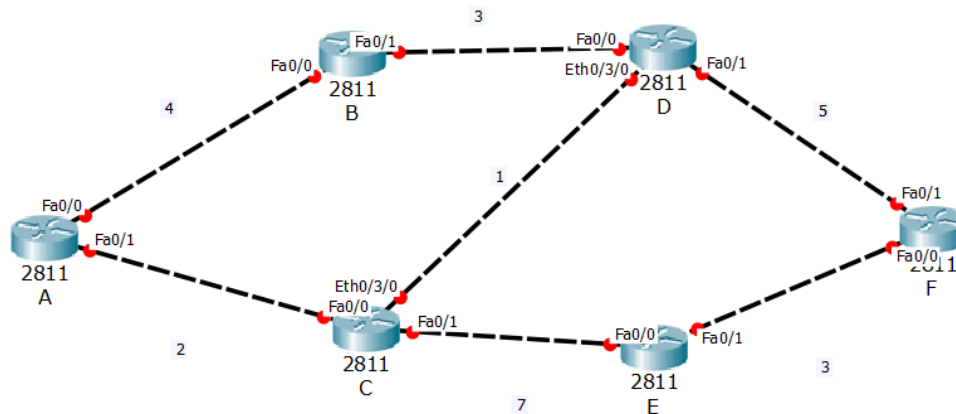
The task is to find the least costly path for the data packet to travel from the source server (Router A) to the destination server (Router F) using Uniform Cost Search (UCS).

Example Output:

Using UCS, the algorithm should explore paths such as:

A → C → D → F (total cost: 2 + 1 + 5 = 8)

A → B → D → F (total cost: 4 + 3 + 5 = 12)



## Task 2: Greedy Search: Word Ladder Puzzle

You are given a starting word and a goal word of the same length. You can change only one letter at a time, but each intermediate word must be a valid English word from a given dictionary.

Implement Greedy Search to find a transformation sequence that leads from the start word to the goal word. The heuristic function will be the number of letters that differ between the current word and the goal word (Hamming distance).

Example 1:

```
start = "hit"
goal = "cog"
word_list = ["hit", "hot", "dot", "dog", "cog", "lot", "log"]
```

```
Path: hit → hot → dot → dog → cog
```

## Heuristic Function (Hamming Distance)

Use this heuristic function. For each word, calculate the number of differing letters compared to the goal word.

Example for cog:

```
hot → cog   (Differs in 2 places: h→c, t→g) → Heuristic h("hot") = 2  
dot → cog   (Differs in 2 places: d→c, t→g) → Heuristic h("dot") = 2  
dog → cog   (Differs in 1 place: d→c)       → Heuristic h("dog") = 1
```