Department Of Computer Science and Engineering

Course Title: Artificial Intelligence and Expert Systems Lab

Course Code: CSE 404

Date of Submission: 04/13/2025

Submitted To:

Noor Mairukh Khan Arnob
Lecturer
Department Of Computer Science & Engineering

Submitted By:

Jannatun Saumoon
ID: 21201066
Sec: B2

Problem Title:

Implementation of a small Address Map (from my own home to UAP) using A* Search Algorithm.

Problem Description:

The objective of this problem is to determine the optimal path & the optimal path cost from Gandaria (home) to UAP(University of Asia Pacific) using the A* search algorithm.

A* search algorithm formula,

$f(n) = g(n) + h(n)$

Where,

$f(n)$ = Estimated cost from path n node to goal node

$g(n)$ = Actual Cost from start node to n-node

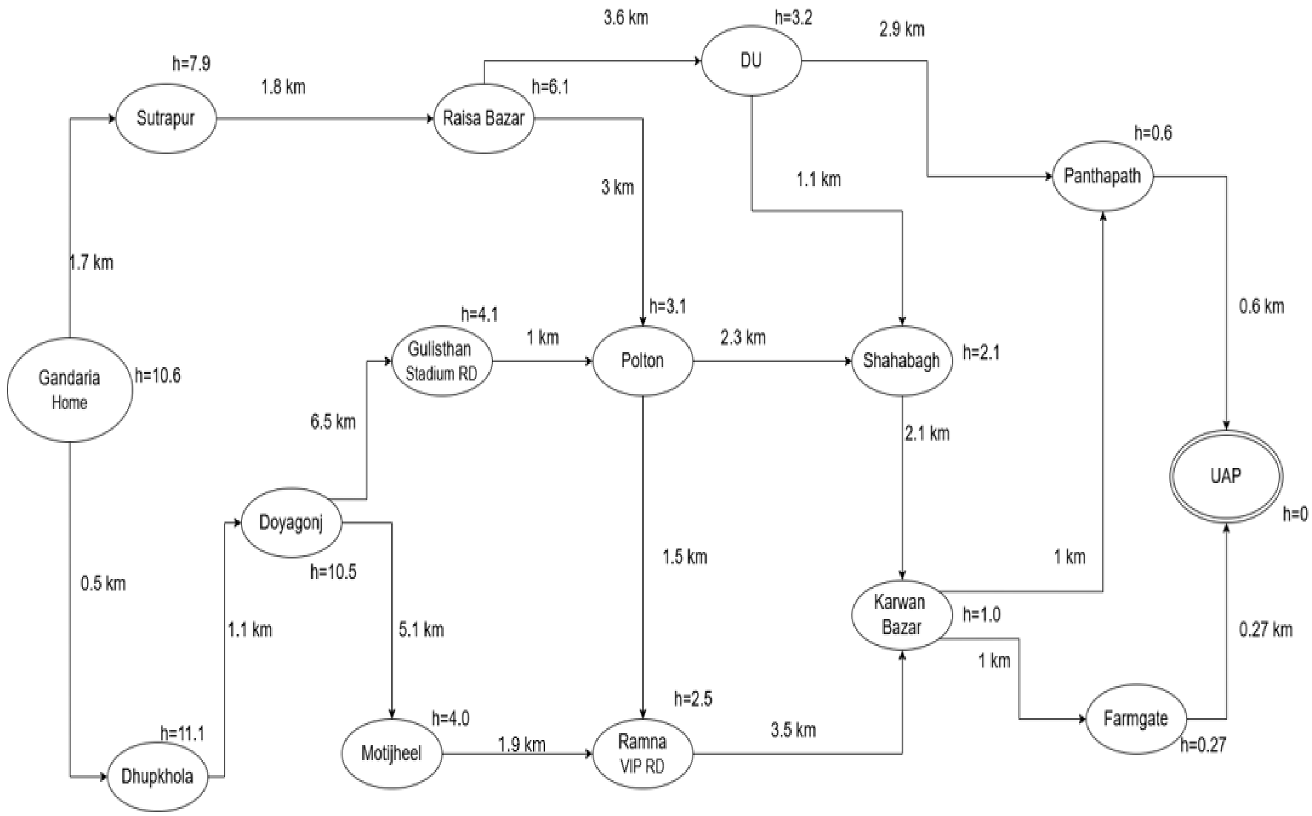
$h(n)$ = Estimated Cost from n-node to goal node

Tools and Languages Used:

- Programming Language: Python
- Tools: Google Colab

Diagram/Figure:

Designed Map:



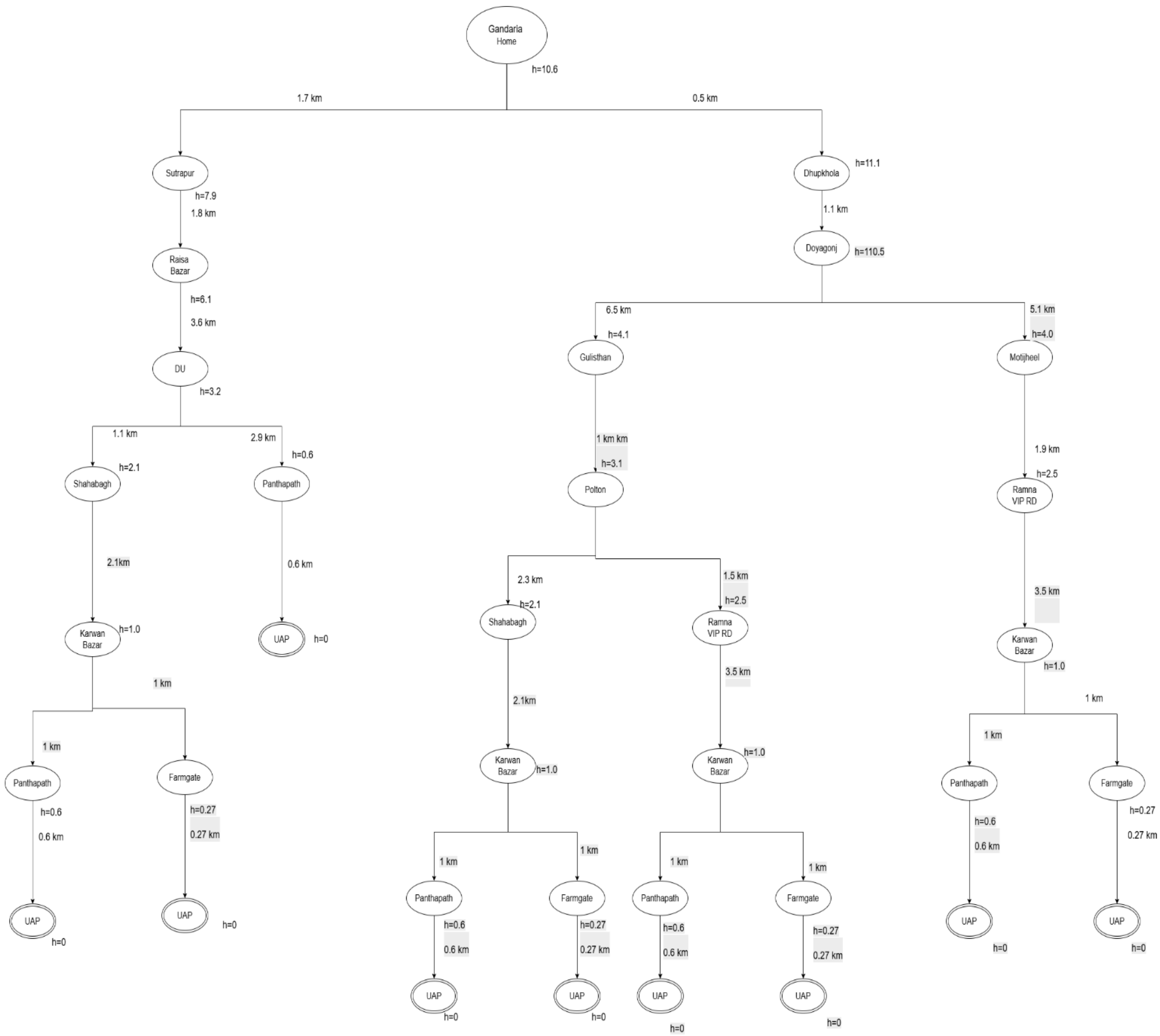
Here,

Start Node: Saumoon’s Home(Gandaria)

Goal Node: UAP

Cost in Distance: Kilometer(km)

Search tree of designed Map:



Sample Input/Output:

Input:

```
import heapq

# Graph: adjacency list with edge costs (g(n))
graph = {
    'Gandaria Home': [('Sutrapur', 1.7), ('Dhupkhola', 0.5)],
    'Sutrapur': [('Raisa Bazar', 1.8)],
    'Raisa Bazar': [('DU', 3.6)],
    'DU': [('Shahbagh', 1.1), ('Panthapath', 2.9)],
    'Shahbagh': [('Karwan Bazar', 2.1)],
    'Panthapath': [('UAP', 0.6)],
    'Karwan Bazar': [('Panthapath', 1.0), ('Farmgate', 1.0)],
    'Farmgate': [('UAP', 0.27)],
    'Dhupkhola': [('Doyagonj', 1.1)],
    'Doyagonj': [('Gulistan', 6.5)],
    'Gulistan': [('Polton', 1.0)],
    'Polton': [('Shahbagh', 2.3), ('Ramna VIP RD', 1.5)],
    'Ramna VIP RD': [('Karwan Bazar', 3.5)],
    'Motijheel': [('Ramna VIP RD', 1.9)],
    'UAP': []
}

# Heuristic h(n) to UAP
heuristics = {
```

```python
# Heuristic h(n) to UAP
heuristics = {
    'Gandaria Home': 10.6,
    'Dhupkhola': 11.1,
    'Doyagonj': 10.5,
    'Sutrapur': 7.9,
    'Raisa Bazar': 6.1,
    'Gulistan': 4.1,
    'Motijheel': 4.0,
    'Polton': 3.1,
    'Ramna VIP RD': 2.5,
    'DU': 3.2,
    'Shahbagh': 2.1,
    'Karwan Bazar': 1.0,
    'Panthapath': 0.6,
    'Farmgate': 0.27,
    'UAP': 0.0
}

def a_star_search(start, goal):
    open_set = []
    heapq.heappush(open_set, (heuristics[start], 0, start, [start]))
```

```python
def a_star_search(start, goal):
    open_set = []
    heapq.heappush(open_set, (heuristics[start], 0, start, [start]))

    visited = set()

    while open_set:
        f, g, current, path = heapq.heappop(open_set)

        if current in visited:
            continue
        visited.add(current)

        if current == goal:
            return path, g

        for neighbor, cost in graph.get(current, []):
            if neighbor not in visited:
                new_g = g + cost
                new_f = new_g + heuristics[neighbor]
                heapq.heappush(open_set, (new_f, new_g, neighbor, path + [neighbor]))

    return None, float('inf')
```

```python
    return None, float('inf')

# Run A* search
start = 'Gandaria Home'
goal = 'UAP'
path, total_cost = a_star_search(start, goal)

# Output result
print(" Optimal Path from Gandaria Home to UAP:")
print(" → ".join(path))
print(f" Total Cost: {total_cost:.2f} km")
```

```
 Optimal Path from Gandaria Home to UAP:
Gandaria Home → Sutrapur → Raisa Bazar → DU → Panthapath → UAP
 Total Cost: 10.60 km
```

<u>Output:</u>


```
 Optimal Path from Gandaria Home to UAP:
Gandaria Home → Sutrapur → Raisa Bazar → DU → Panthapath → UAP
 Total Cost: 10.60 km
```


Conclusion:

By implementing the A* search algorithm, we efficiently determined the most optimal path and the optimal path cost from Gandaria (Home)to UAP, minimizing travel distance. The algorithm effectively balances the actual travel cost (g(n)) with the estimated distance (h(n)), ensuring the shortest possible route while maintaining high computational efficiency.