

Itisha Desai

Sem – 5

Branch – Cyber Security

Batch – CSE54

Enrollment No. – 22162171006

Algorithm Analysis and Design

Practical-12

Question :-

“Rocket Singh: Salesman of the Year” is a travelling salesman, who sales good in various cities. One day in the morning, he decided to visit all the cities to sales good and come back to the starting city (from where he has started). Travelling Salesman Problem (TSP) is a touring problem in which n cities and distance between each pair is given. We have to help him to find a shortest route to visit each city exactly once and come back to the starting point.

Sample Input:

$[[\infty, 20, 30, 10, 11],$

$[15, \infty, 16, 4, 2],$

$[3, 5, \infty, 2, 4],$

$[19, 6, 18, \infty, 3],$

$[16, 4, 7, 16, \infty]]$

Sample Output:

Minimum Path

$1 - 4 = 10$

$4 - 2 = 6$

$2 - 5 = 2$

$5 - 3 = 7$

$3 - 1 = 3$

Minimum cost: 28

Path Taken: 1 - 4 - 2 - 5 - 3 - 1

Code :-

app.py :

```
from flask import Flask, render_template, request, redirect, url_for
import itertools
import numpy as np
app = Flask(__name__)
def calculate_tsp(matrix):
    n=len(matrix)
    min_cost = float('inf')
    min_path = []

    for perm in itertools.permutations(range(1, n)):
        current_cost = matrix[0][perm[0]] + sum(matrix[perm[i]][perm[i+1]]
            for i in range(len(perm)- 1)) + matrix[perm[-1]][0]

        if current_cost < min_cost:
            min_cost = current_cost
            min_path = [0] + list(perm) + [0]

    path_taken = '- '.join(str(x + 1) for x in min_path)
    min_cost_details = [(min_path[i] + 1, min_path[i + 1] + 1,
        matrix[min_path[i]][min_path[i + 1]]) for i in
        range(len(min_path)- 1)]
    return min_cost, path_taken, min_cost_details

@app.route('/', methods=['GET', 'POST'])

def input_matrix():
    if request.method == 'POST':
        size = int(request.form['size'])
        return redirect(url_for('get_matrix', size=size))
    return render_template('index3.html')

@app.route('/matrix/<int:size>', methods=['GET', 'POST'])

def get_matrix(size):
    if request.method == 'POST':
        matrix = []
        for i in range(size):
```

```

        row = []
        for j in range(size):
            value = request.form.get(f'cell_{i}_{j}')
            row.append(float('inf') if value.lower() == 'inf' else int(value))
        matrix.append(row)
    matrix = np.array(matrix)
    min_cost, path_taken, min_cost_details = calculate_tsp(matrix)
    return render_template('index1.html', min_cost=min_cost, path_taken=path_taken,
min_cost_details=min_cost_details)
    return render_template('index2.html', size=size)

if __name__ == '__main__':
    app.run(debug=True)

```

index1.html :

```

<!DOCTYPE html>

<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>TSP Solution</title>
<style>
body { font-family: Arial, sans-serif; margin: 20px; }
h1 { color: #333; }
ul { list-style-type: none; padding: 0; }
li { margin-bottom: 8px; }
</style>
</head>
<body>
<h1>Travelling Salesman Problem Solution</h1>
<p><b>Minimum Cost:</b> {{ min_cost }}</p>
<p><b>Path Taken:</b> {{ path_taken }}</p>

```

```

<ul>
{% for frm, to, cost in min_cost_details %}
<li>{{ frm }}- {{ to }} = {{ cost }}</li>
{% endfor %}
</ul>
</body>
</html>

```

index2.html:

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Input Distance Matrix</title>
</head>
<body>
<h1>Enter Distances Between Cities</h1>
<form action="" method="post">
<table>
{% for i in range(size) %}
<tr>
{% for j in range(size) %}
<td>
<input type="text" name="cell_{{ i }}_{{ j }}"
placeholder="{{ 'inf' if i == j else '0' }}"
value="{{ 'inf' if i == j else '' }}" required>

```

```
</td>
{% endfor %}
</tr>
{% endfor %}
</table>
<button type="submit">Calculate</button>
</form>
</body>
</html>
```

index3.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Input Matrix Size</title>
</head>
<body>
<h1>Enter the Size of Distance Matrix</h1>
<form action="/" method="post">
<label for="size">Number of cities:</label>
<input type="number" id="size" name="size" min="2" required>
<button type="submit">Next</button>
</form>
</body>
</html>
```

Output :-

Enter the Size of Distance Matrix

Number of cities:

Enter Distances Between Cities

inf	20	30	10	11
15	inf	16	4	2
3	5	inf	2	4
19	6	18	inf	3
16	4	7	16	inf

Travelling Salesman Problem Solution

Minimum Cost: 28.0

Path Taken: 1- 4- 2- 5- 3- 1

1- 4 = 10.0

4- 2 = 6.0

2- 5 = 2.0

5- 3 = 7.0

3- 1 = 3.0