**Institute of Computer Technology**

**B. Tech Computer Science and Engineering**

**Sub: Algorithm Analysis and Design**

**Practical 5**

You are working at the cash counter at a fun-fair, and you have three types of coins available to you in infinite quantities (coins are Rs. 1, Rs. 4 and Rs. 6). You are required to calculate the minimum numbers of coins required for changing the value of Rs. 9.

Design the algorithm for the same and implement using the programming language of your choice. Make comparative analysis for various use cases & input size.

Code:

App.py

from flask import Flask, render\_template, request

import matplotlib.pyplot as plt

import io

import base64

app = Flask(\_\_name\_\_)

def min\_coins(coins, value):

    dp = [float('inf')] \* (value + 1)

    dp[0] = 0

    for i in range(1, value + 1):

        for coin in coins:

            if i - coin >= 0:

                dp[i] = min(dp[i], dp[i - coin] + 1)

    return dp[value]

*# Index route (homepage)*

@app.route('/')

def index():

    return render\_template('index.html')

*# Route for Task 1 page*

@app.route('/task1', methods=['GET', 'POST'])

def task1():

    coins = [1, 4, 6]

    result = None

    values = []

    results = []

    if request.method == 'POST':

        value = int(request.form['value'])

        result = min\_coins(coins, value)

*# Comparative analysis for different input sizes*

        for i in range(1, value + 1):

            values.append(i)

            results.append(min\_coins(coins, i))

*# Plotting the graph with green line color*

        plt.figure(figsize=(10, 6))

        plt.plot(values, results, marker='o', color='green')

        plt.title('Minimum Coins Required for Different Values')

        plt.xlabel('Value')

        plt.ylabel('Minimum Coins')

        plt.grid(True)

*# Save plot to a string in base64 format*

        img = io.BytesIO()

        plt.savefig(img, format='png')

        img.seek(0)

        plot\_url = base64.b64encode(img.getvalue()).decode()

        return render\_template('task1.html', result=result, value=value, plot\_url=plot\_url)

    return render\_template('task1.html')

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(debug=True)

Task1.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Cash Counter - Minimum Coins</title>

    <link rel="stylesheet" href="{{ url\_for('static', filename='task1.css') }}">

</head>

<body>

    <div class="container">

        <h1>Cash Counter - Minimum Coins Calculation</h1>

        <form method="POST">

            <label for="value">Enter the value (in Rs.):</label>

            <input type="number" id="value" name="value" required>

            <button type="submit">Calculate</button>

        </form>

        {% if result is not none %}

        <div class="result">

            <h2>Result</h2>

            <p>Minimum number of coins required for Rs. {{ value }}: <strong>{{ result }}</strong></p>

            <h3>Comparative Analysis</h3>

            <img src="data:image/png;base64,{{ plot\_url }}" alt="Comparative Analysis Graph">

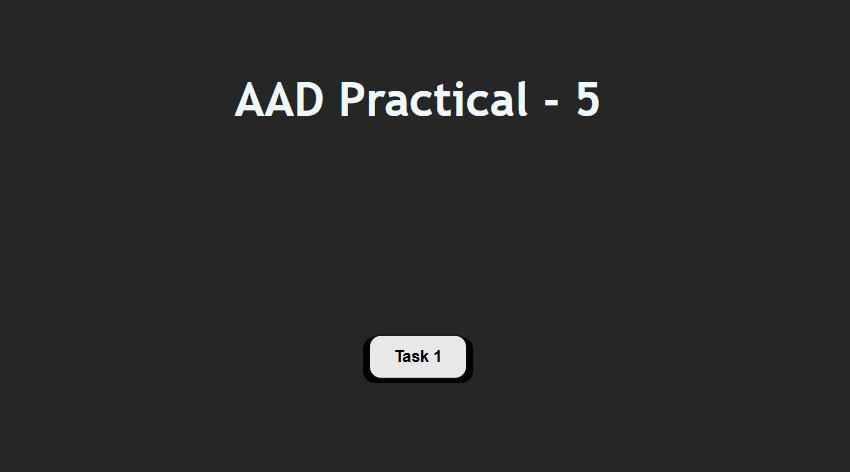
        </div>

        {% endif %}

    </div>

</body>

</html>

Output:

