

Name: Devanshi Jain
Enrollment-No: 22162101006
Batch: 51

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:

.py:

```
from flask import Flask, render_template, request
import matplotlib.pyplot as plt
import io
import base64

app = Flask(__name__)

# Function to calculate the minimum number of coins
def min_coins(coins, amount):
    dp = [float('inf')] * (amount + 1)
    dp[0] = 0

    for i in range(1, amount + 1):
        for coin in coins:
            if i >= coin:
                dp[i] = min(dp[i], dp[i - coin] + 1)
```

```

        return dp[amount] if dp[amount] != float('inf') else -1

@app.route('/', methods=['GET', 'POST'])
def index():
    coins = [1, 4, 6]
    amounts = list(range(1, 11))
    results = [min_coins(coins, amount) for amount in amounts]

    # Initialize variables
    min_coins_result = None
    user_amount = None

    if request.method == 'POST':
        # Get user input from form
        try:
            user_amount = int(request.form['amount'])
            min_coins_result = min_coins(coins, user_amount)
        except ValueError:
            min_coins_result = "Invalid input. Please enter a valid integer."

    # Generate the plot using Matplotlib
    fig, ax = plt.subplots()
    ax.plot(amounts, results, marker='o', linestyle='-', color='b')
    ax.set_xlabel('Amount (Rs.)')
    ax.set_ylabel('Number of Coins')
    ax.set_title('Minimum Coins Required for Various Amounts')

    # Save plot to a BytesIO object
    img = io.BytesIO()
    plt.savefig(img, format='png')
    img.seek(0)

    # Encode image to base64
    img_base64 = base64.b64encode(img.getvalue()).decode('utf-8')

    return render_template('p5.html',
                           image_data=img_base64,
                           min_coins_result=min_coins_result,
                           user_amount=user_amount)

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

```

.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Coin Change Analysis</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      text-align: center;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
    }
    h1 {
      color: #333;
      margin-top: 20px;
    }
    img {
      margin-top: 20px;
      border: 1px solid #ccc;
      background-color: #fff;
    }
    form {
      margin: 20px;
    }
    input[type="number"] {
      padding: 10px;
      font-size: 16px;
      width: 200px;
    }
    input[type="submit"] {
      padding: 10px 20px;
      font-size: 16px;
      cursor: pointer;
    }
    .result {
      margin: 20px;
      font-size: 18px;
      color: #333;
    }
  </style>
</head>
<body>
```

```

<h1>Minimum Coins Required for Various Amounts</h1>


<form method="POST" action="/">
  <label for="amount">Enter Amount (Rs.): </label>
  <input type="number" id="amount" name="amount" required>
  <input type="submit" value="Calculate">
</form>

{% if min_coins_result is not none %}
  <div class="result">
    {% if min_coins_result == "Invalid input. Please enter a valid
integer." %}
      {{ min_coins_result }}
    {% else %}
      Minimum number of coins required for Rs. {{ user_amount }}: {{
min_coins_result }}
    {% endif %}
  </div>
{% endif %}
</body>
</html>

```

OUTPUT:

