NAME: MAURYA R PATEL
ER-NO: 22162101014
BATCH: 51
BRANCH:CBA

Institute of Computer Technology
B. Tech Computer Science and Engineering

Sub: Algorithm Analysis and Design

Practical 1

(1) There are 2 chefs, namely chef 1 and chef 2 in the MasterChef competition. The judge is going to judge on the basis of 3 categories: presentation, taste and hygiene to prepare the dishes. The marking is scaling from 1 to 100. The rating for chef 1 challenge is the triplet a = (a[0], a[1], a[2]), and the rating for Chef 2 challenge is the triplet b = (b[0], b[1], b[2]), where 0 index is presentation, 1 index is taste and 2 index is hygiene.
The task is to find their comparison points by comparing a[0] with b[0], a[1] with b[1], and a[2] with b[2].

- If a[i] > b[i], then Chef 1 is awarded 1 point.
- If a[i] < b[i], then Chef 2 is awarded 1 point.
- If a[i] = b[i], then neither person receives a point.

Comparison points are the total points a person earned.
Given a and b, determine their respective comparison points.
Design the algorithm for the same and implement using the programming language of your choice. Make comparative analysis for various use cases & input size.

Sample Input 1
27 48 70
89 26 7
Sample Output 1
2 1
Explanation 1
Comparing the 0th elements, 27<89 so Chef 2 receives a point.
Comparing the 1st and 2nd elements, 48>26 and 70>7 so Chef 1 receives two points.
The return array is [2,1].

CODE:

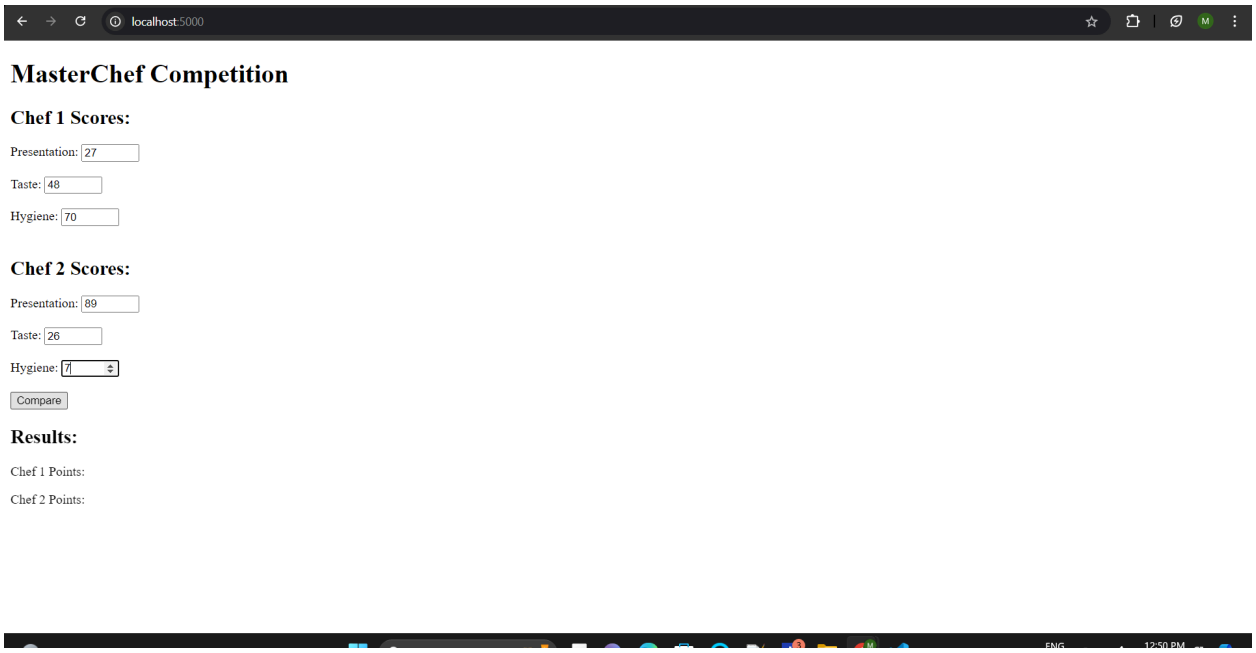
```
from flask import Flask, request, jsonify, render_template

app = Flask(__name__)

def compare_chefs(a, b):
    chef1_points = 0
    chef2_points = 0

    for i in range(3):
        if a[i] > b[i]:
            chef1_points += 1
        elif a[i] < b[i]:
            chef2_points += 1

    return chef1_points, chef2_points

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/compare', methods=['POST'])
def compare():
    try:
        a = list(map(int, request.form.getlist('a')))
        b = list(map(int, request.form.getlist('b')))

        if len(a) != 3 or len(b) != 3:
                return render_template('index.html', error='Invalid input
length. Expected triplets.')

        chef1_points, chef2_points = compare_chefs(a, b)
           return render_template('index.html', chef1_points=chef1_points,
chef2_points=chef2_points)
    except Exception as e:
        return render_template('index.html', error=str(e))

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

.html file:

```
<!DOCTYPE html>
<html>
<head>
    <title>MasterChef Competition</title>
</head>
<body>
    <h1>MasterChef Competition</h1>
    <form method="POST" action="/compare">
        <h2>Chef 1 Scores:</h2>
        <label for="a_presentation">Presentation:</label>
        <input type="number" id="a_presentation" name="a" min="1"
max="100" required>
        <br><br>
        <label for="a_taste">Taste:</label>
        <input type="number" id="a_taste" name="a" min="1" max="100"
required>
        <br><br>
        <label for="a_hygiene">Hygiene:</label>
        <input type="number" id="a_hygiene" name="a" min="1" max="100"
required>
        <br><br>

        <h2>Chef 2 Scores:</h2>
        <label for="b_presentation">Presentation:</label>
        <input type="number" id="b_presentation" name="b" min="1"
max="100" required>
        <br><br>
        <label for="b_taste">Taste:</label>
        <input type="number" id="b_taste" name="b" min="1" max="100"
required>
        <br><br>
        <label for="b_hygiene">Hygiene:</label>
        <input type="number" id="b_hygiene" name="b" min="1" max="100"
required>
        <br><br>

        <button type="submit">Compare</button>
    </form>
```

```
    {% if chef1_points is not none and chef2_points is not none %}
    <h2>Results:</h2>
    <p>Chef 1 Points: {{ chef1_points }}</p>
    <p>Chef 2 Points: {{ chef2_points }}</p>
    {% endif %}

    {% if error %}
    <p style="color:red;">{{ error }}</p>
    {% endif %}
</body>
</html>
```

OUTPUT:



Results:

Chef 1 Points: 2

Chef 2 Points: 1

(2) Let us suppose that you are having an array containing both positive and negative numbers. Given the numbers you are supposed to find 2 such elements such that the sum of those numbers is closest to zero.

Sample Input 1
15, 5, -20, 30, -45

Sample Output 1
15, -20

Explanation 1
In all the comparison, the sum of 15 and -20 is smallest amount among all other comparison.

Sample Input 2
15, 5, -20, 30, 25

Sample Output 2
15, -20 & -20, 25

Explanation 2
In all the comparison, the sum of 15,-20 & -20, 25 is smallest amount among all other comparison.

CODE:

.py file:

```python
from flask import Flask, request, render_template
import itertools

app = Flask(__name__)

def find_closest_pairs(nums):
    nums.sort()
    closest_sum = float('inf')
    closest_pairs = []

    for i in range(len(nums) - 1):
        for j in range(i + 1, len(nums)):
            current_sum = nums[i] + nums[j]
            if abs(current_sum) < abs(closest_sum):
                closest_sum = current_sum
                closest_pairs = [(nums[i], nums[j])]
            elif abs(current_sum) == abs(closest_sum):
                closest_pairs.append((nums[i], nums[j]))

    return closest_pairs

@app.route('/', methods=['GET', 'POST'])
def index():
    result = None
    if request.method == 'POST':
        numbers = request.form.get('numbers')
        num_list = list(map(int, numbers.split(',')))
        result = find_closest_pairs(num_list)
    return render_template('result.html', result=result)

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

.html file:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Find Closest Pairs</title>
</head>
<body>
    <h1>Find Pairs with Sum Closest to Zero</h1>
    <form method="POST">
        <label for="numbers">Enter numbers :</label>
        <input type="text" id="numbers" name="numbers" required>
        <button type="submit">Find Pairs</button>
    </form>
    {% if result %}
        <h2>Closest Pairs</h2>
        <ul>
            {% for pair in result %}
                <li>{{ pair[0] }}, {{ pair[1] }}</li>
            {% endfor %}
        </ul>
    {% endif %}
</body>
</html>
```

OUTPUT:

Find Pairs with Sum Closest to Zero

Enter numbers (comma-separated): 15,5,-20,30,25 Find Pairs

Closest Pairs

- -20, 15
- -20, 25

Find Pairs with Sum Closest to Zero

Enter numbers : 15,5,-20,30,-45 Find Pairs

Closest Pairs

- -20, 15