Name : Harpalsinh Bhati

Enr. No.: 22162171010

Sub: Algorithm Analysis & Design

Branch: CS

Batch: 54

Sem: 5-B

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, render_template_string, request

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
        # No points are awarded if a[i] == b[i]

    return [chef1_points, chef2_points]

@app.route('/', methods=['GET', 'POST'])
def index():
    result = None
    if request.method == 'POST':
        a = list(map(int, request.form['a'].split(',')))
        b = list(map(int, request.form['b'].split(',')))
        result = compare_chefs(a, b)

    return render_template_string('''
        <!doctype html>
        <title>Compare Chefs</title>
        <h1>Compare Chefs</h1>
        <form method="post">
            Chef A Scores: <input type="text" name="a" placeholder="e.g.
27,48,70"><br><br>
            Chef B Scores: <input type="text" name="b" placeholder="e.g.
89,26,7"><br><br>
            <input type="submit" value="Compare">
        </form>
        {% if result is not none %}
            <h2>Result: {{ result }}</h2>
        {% endif %}
    ''', result=result)

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

Output:

Compare Chefs

Chef A Scores: 27, 48, 70

Chef B Scores: 89, 26, 7

Compare

Result: [0, 3]

Compare Chefs

Chef A Scores: e.g. 27,48,70

Chef B Scores: e.g. 89,26,7

Compare

Result: [2, 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:

```
from flask import Flask, render_template_string, request

app = Flask(__name__)

def find_closest_sum_pair(arr):
    arr.sort()

    left = 0
    right = len(arr) - 1
    closest_sum = float('inf')
    closest_pair = (0, 0)

    while left < right:
        current_sum = arr[left] + arr[right]

        if abs(current_sum) < abs(closest_sum):
            closest_sum = current_sum
            closest_pair = (arr[left], arr[right])

        if current_sum < 0:
            left += 1
        else:
            right -= 1

    return closest_pair

@app.route('/', methods=['GET', 'POST'])
def index():
    pair = None
    if request.method == 'POST':
        arr = list(map(int, request.form['array'].split(',')))
        pair = find_closest_sum_pair(arr)

    return render_template_string('''
        <!doctype html>
        <title>Find Closest Sum Pair</title>
        <h1>Find the Pair Whose Sum is Closest to Zero</h1>
        <form method="post">
```

```
            Array (comma-separated): <input type="text" name="array"
placeholder="e.g. 1, 60, -10, 70, -80, 85"><br><br>
            <input type="submit" value="Find Pair">
        </form>
        {% if pair is not none %}
            <h2>The pair whose sum is closest to zero is: {{ pair }}</h2>
        {% endif %}
    ''', pair=pair)

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

Output:

Find the Pair Whose Sum is Closest to Zero

Array (comma-separated): 15, 5, -20, 30, -45

Find Pair

Find the Pair Whose Sum is Closest to Zero

Array (comma-separated): e.g. 1, 60, -10, 70, -80, 85

Find Pair

The pair whose sum is closest to zero is: (-20, 15)