**Batch: SY-IT(B3)** **Experiment Number:4**

**Roll Number: 16010423076** **Name:Ritesh Jha**

**Aim of the Experiment:** To study Greedy Programming approach for implementation of problem statement to obtain optimal solution.

---

**Program/ Steps:**

```
#include <iostream>
using namespace std;

int main() {
    int T;
    cin >> T;

    while (T--) {
        string num;
        cin >> num;

        // Step 1: Count the frequency of each digit (0 to 9)
        int freq[10] = {0};
        for (int i = 0; i < num.length(); i++) {
            freq[num[i] - '0']++;  // convert character to integer
and update frequency
        }

        // Step 2: Store digits in sorted order
        char digits[20];
        int index = 0;  // Index to track position in digits array
        for (int i = 0; i < 10; i++) {
            while (freq[i] > 0) {  // If this digit appears in the
input
```

```
                digits[index++] = '0' + i;  // Store it in the
array
                freq[i]--;
            }
        }

        // Step 3: Distribute digits into two numbers alternately
        string num1 = "", num2 = "";  // Two numbers as strings
        for (int i = 0; i < index; i++) {
            if (i % 2 == 0) {  // Even index goes to num1
                num1 += digits[i];
            } else {  // Odd index goes to num2
                num2 += digits[i];
            }
        }

        //Convert strings to integers and calculate sum
        int sum = (num1.empty() ? 0 : stoi(num1)) + (num2.empty()
? 0 : stoi(num2));



        cout << sum << endl;
    }
    return 0;
}
```

---

**Output/Result:**

Log ID: 270607084 / Feb 18, 2025 02:04 AM PST (America/Los_Angeles)

**RESULT:** Sample Test Cases Passed

Refer judge environment

**Note:** When you **Compile & Test code**, the code is run against sample inputs. When you **Submit code**, the code is run against sample input as well as multiple hidden test cases. In order to solve the problem, your code must pass all of the test cases.

| Time (sec) | Memory (KiB) | Language |
|---|---|---|
| 0.008695 | 2 | C++14 |

**Input**

```
2
1321
42255
```

**Output**

```
25
270
```

**Expected Correct Output**

```
25
270
```

Submission ID: 107528870

**RESULT:** Accepted

Refer judge environment

| Score | Time (sec) | Memory (KiB) | Language |
|---|---|---|---|
| 20 | 0.25724 | 356 | C++14 |

| Input | Result | Time (sec) | Memory (KiB) | Score | Your output | Correct output | Diff |
|---|---|---|---|---|---|---|---|
| Input #1 | Accepted | 0.0656 | 356 | 4 | | | |
| Input #2 | Accepted | 0.098288 | 356 | 20 | | | |
| Input #3 | Accepted | 0.065873 | 356 | 20 | | | |
| Input #4 | Accepted | 0.009338 | 2 | 20 | | | |
| Input #5 | Accepted | 0.009365 | 2 | 20 | | | |
| Input #6 | Accepted | 0.008778 | 2 | 16 | | | |

**Outcomes** : CO2. Understand the fundamental concepts for managing the data using different data structures such as lists, queues, trees etc.

**Conclusion (based on the Results and outcomes achieved):**

From this experiment, I learned how to efficiently process and manipulate numerical data using arrays and strings in C++. By implementing a digit frequency count and sorting mechanism, I understood how to systematically distribute numbers to achieve a desired outcome.

**References:**

1. https://tutorialspoint.dev/algorithm/greedy-algorithms/greedy-algorithm-to-find-minimum-number-of-coins
2. https://www.baeldung.com/cs/min-number-of-coins-algorithm