

# Task 11: STL Vectors in C++

## Objective:

To understand and demonstrate the use of the STL vector container in C++. This task involves storing integers in a vector and writing separate functions to calculate the sum and average of the elements.

## 1. Concept Overview

The Standard Template Library (STL) vector is a dynamic array that can automatically resize itself. It provides fast random access and is widely used for storing and manipulating collections of data.

## 2. Program Code

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

int findSum(const vector<int>& v) {
    int sum = 0;
    for (int num : v)
        sum += num;
    return sum;
}

double findAverage(const vector<int>& v) {
    int sum = findSum(v);
    return static_cast<double>(sum) / v.size();
}

int main() {
    vector<int> numbers;

    numbers.push_back(10);
    numbers.push_back(20);
    numbers.push_back(30);
    numbers.push_back(40);
    numbers.push_back(50);

    cout << "Vector elements: ";
    for (int n : numbers)
        cout << n << " ";
    cout << endl;
```

```

cout << "Sum = " << findSum(numbers) << endl;
cout << "Average = " << findAverage(numbers) << endl;

return 0;
}

```

### 3. Compilation Instructions

`g++ vector_demo.cpp -o vector_demo`

### 4. Sample Output

```

student@student-virtual-machine:~/25SUB4508_LSP/25SUB4508_56133/ClassWork/day21$ g++ vector_demo.cpp -o vector_demo
student@student-virtual-machine:~/25SUB4508_LSP/25SUB4508_56133/ClassWork/day21$ ./vector_demo
Vector elements: 10 20 30 40 50
Sum = 150
Average = 30
student@student-virtual-machine:~/25SUB4508_LSP/25SUB4508_56133/ClassWork/day21$ █

```

### 5. Observations & Explanation

1. A `vector<int>` is used to store integer values dynamically.
2. Elements are added using the `push_back()` function.
3. The `findSum()` function calculates the total of all elements.
4. The `findAverage()` function calculates the average using the sum and size of vector.
5. Range-based for loops simplify iteration over vector elements.

### 6. Advantages of STL Vector

- Dynamic resizing
- Efficient random access
- Easy integration with STL algorithms
- Safer than traditional arrays

### 7. Conclusion

This task demonstrates how STL vectors can be used to store and manipulate collections of data efficiently. Writing separate functions for sum and average improves modularity and code readability.