Task 14: Range-based Loops in C++

Objective:
To understand and demonstrate the use of range-based for loops in C++. This task iterates over a container using a range-based loop and prints each element.

1. Concept Overview

A range-based for loop provides a simple and readable way to iterate over elements of a container such as arrays, vectors, or other STL containers. It eliminates the need for index variables and makes code cleaner.

2. Program Code

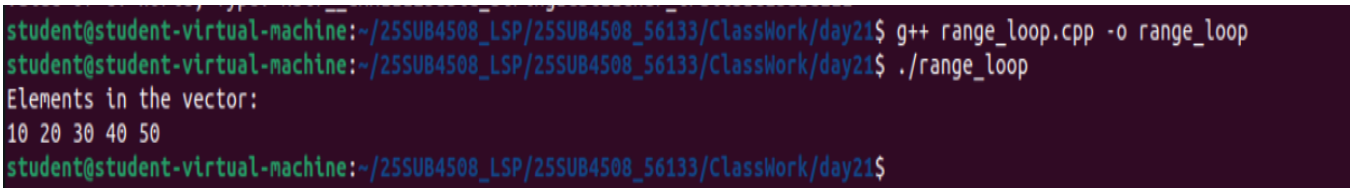
```
#include <iostream>
 #include <vector>
 using namespace std;

 int main() {
        vector<int> numbers = {10, 20, 30, 40, 50};

        cout << "Elements in the vector:" << endl;
        for (int num : numbers) {
        cout << num << " ";
        }
        cout << endl;

        return 0;
}
```

3. Compilation Instructions

Compile the program using a C++ compiler:

```
 g++ range_loop.cpp -o range_loop
```

4. Sample Output



5. Observations & Explanation

1. A vector<int> is created with integer elements.
2. The range-based for loop iterates over each element directly.
3. The loop variable 'num' takes the value of each element in sequence.
4. No index or size calculation is required.

6. Advantages of Range-based Loops

- Cleaner and more readable code
- Avoids index-related errors
- Works with arrays and STL containers

7. Conclusion

This task demonstrates the simplicity and effectiveness of range-based loops in C++. They are widely used in modern C++ for safe and concise iteration over containers.