Institute of Computer Technology

B. Tech. Computer Science and Engineering

Sub: ESFP – II

Course Code: 2CSE203

Practical – 16

Name: Jaymin Gondaliya

Enrollment No: 23162171007

Sem - 2

Branch: CS

Class: B

Batch: 25

Objective:

To learn about STL, that is vector, deque, list, set, Multiset, map, multimap,stack and queue.

Problem Definition-1: Complete the code for the object assigned to you to satisfy the following specifications.

Code:

```cpp
#include <iostream>
#include <vector>
#include <deque>
#include <list>
#include <set>
#include <map>
#include <stack>
#include <queue>
#include <algorithm>
using namespace std;

void vectorModule() {
    vector<int> vec;
    int element;
    cout << "Enter elements to add (enter -1 to stop): ";
    while (cin >> element && element != -1) {
```

```
        vec.push_back(element);
    }

    cout << "Vector size: " << vec.size() << endl;
    cout << "Vector elements: ";
    for (auto it = vec.begin(); it != vec.end(); ++it) {
        cout << *it << " ";
    }
    cout << endl;

    vec.pop_back();
    cout << "After pop_back(), Vector size: " << vec.size() << endl;
}

void dequeModule() {
    deque<int> dq;
    int element;
    cout << "Enter elements to add (enter -1 to stop): ";
    while (cin >> element && element != -1) {
        dq.push_back(element);
    }

    cout << "Deque size: " << dq.size() << endl;
    cout << "Deque elements: ";
    for (auto it = dq.begin(); it != dq.end(); ++it) {
        cout << *it << " ";
    }
    cout << endl;

    dq.pop_back();
    cout << "After pop_back(), Deque size: " << dq.size() << endl;
}

void listModule() {
    list<int> li;
    int element;
    cout << "Enter elements to add (enter -1 to stop): ";
    while (cin >> element && element != -1) {
        li.push_back(element);
    }

    cout << "List size: " << li.size() << endl;
    cout << "List elements: ";
    for (auto it = li.begin(); it != li.end(); ++it) {
        cout << *it << " ";
    }
    cout << endl;
```

```
    li.pop_back();
    cout << "After pop_back(), List size: " << li.size() << endl;
}

void setModule() {
    set<int> s;
    int element;
    cout << "Enter elements to add (enter -1 to stop): ";
    while (cin >> element && element != -1) {
        s.insert(element);
    }

    cout << "Set size: " << s.size() << endl;
    cout << "Set elements: ";
    for (auto it = s.begin(); it != s.end(); ++it) {
        cout << *it << " ";
    }
    cout << endl;

    cout << "Enter element to erase: ";
    cin >> element;
    s.erase(element);
    cout << "After erase(), Set size: " << s.size() << endl;
}

void mapModule() {
    map<string, int> mp;
    string key;
    int value;
    cout << "Enter key-value pairs (enter 'exit' to stop): ";
    while (cin >> key && key != "exit") {
        cin >> value;
        mp[key] = value;
    }

    cout << "Map size: " << mp.size() << endl;
    cout << "Map elements: ";
    for (auto it = mp.begin(); it != mp.end(); ++it) {
        cout << it->first << ":" << it->second << " ";
    }
    cout << endl;

    cout << "Enter key to erase: ";
    cin >> key;
    mp.erase(key);
    cout << "After erase(), Map size: " << mp.size() << endl;
}
```

```cpp
void stackModule() {
    stack<int> s;
    int element;
    cout << "Enter elements to push (enter -1 to stop): ";
    while (cin >> element && element != -1) {
        s.push(element);
    }

    cout << "Stack size: " << s.size() << endl;
    cout << "Stack top: " << s.top() << endl;
}

void queueModule() {
    queue<int> q;
    int element;
    cout << "Enter elements to push (enter -1 to stop): ";
    while (cin >> element && element != -1) {
        q.push(element);
    }

    cout << "Queue size: " << q.size() << endl;
    cout << "Queue front: " << q.front() << endl;
    cout << "Queue back: " << q.back() << endl;
}

int main() {
    int choice;
    do {
        cout << "\n1. Vector Module\n";
        cout << "2. Deque Module\n";
        cout << "3. List Module\n";
        cout << "4. Set Module\n";
        cout << "5. Map Module\n";
        cout << "6. Stack Module\n";
        cout << "7. Queue Module\n";
        cout << "8. Exit\n";
        cout << "Enter your choice: ";
        cin >> choice;

        switch (choice) {
            case 1:
                vectorModule();
                break;
            case 2:
                dequeModule();
                break;
            case 3:
                listModule();
```

```
                break;
            case 4:
                setModule();
                break;
            case 5:
                mapModule();
                break;
            case 6:
                stackModule();
                break;
            case 7:
                queueModule();
                break;
            case 8:
                cout << "Exiting...";
                break;
            default:
                cout << "Invalid choice!" << endl;
        }
    } while (choice != 8);

    return 0;
}
```

Output:

```
PS C:\ICT\SEM-2\ESFP-II\Practical-16\output> .\main.exe

1. Vector Module
2. Deque Module
3. List Module
4. Set Module
5. Map Module
6. Stack Module
7. Queue Module
8. Exit
Enter your choice: 6
Enter elements to push (enter -1 to stop): 1 3 4 5
-1
Stack size: 4
Stack top: 5
```

```
3. List Module
4. Set Module
5. Map Module
6. Stack Module
7. Queue Module
8. Exit
Enter your choice: 7
Enter elements to push (enter -1 to stop): 5 3 12 4
-1
Queue size: 4
Queue front: 5
Queue back: 4

1. Vector Module
2. Deque Module
3. List Module
4. Set Module
5. Map Module
6. Stack Module
7. Queue Module
8. Exit
Enter your choice:
```