Assignment - 37 **A Job Ready Bootcamp in C++, DSA and IOT**

STL: vector

1. Declare a vector with Initialization and print the elements.

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

int main()
{
    vector <int> v;
    for(int i = 0; i < 10; i++)
        v.push_back(i+1);

    for(int i = 0; i < 10; i++)
        cout << v.at(i) << " ";

    return 0;
}
================================================================================
Output:
1 2 3 4 5 6 7 8 9 10
```

2. Declare a vector without initialization, insert some elements and print

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

int main()
{
    vector<int> v;
    int n;
    cout << "Enter 10 numbers: ";
    for (int i = 0; i < 10; i++)
    {
        cin >> n;
        v.push_back(n);
    }

    for (int i = 0; i < 10; i++)
        cout << v.at(i) << " ";

    return 0;
}
================================================================================
Output:
Enter 10 numbers: 10 20 30 40 50 60 70 80 90 100
10 20 30 40 50 60 70 80 90 100
```

3. Write a function to print the element of a vector and take input from the user.

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

void printElement(vector<int> v1)
{
    cout << "Vector elements are : ";
    for (int i = 0; i < v1.size(); i++)
        cout << v1.at(i) << " ";
}
int main()
{
```

```
    vector<int> v;
    int n;
    cout << "Enter 10 vector elements: ";
    for (int i = 0; i < 10; i++)
    {
        cin >> n;
        v.push_back(n);
    }
    printElement(v);

    return 0;
}
```

```
Output:
Enter 10 vector elements: 10 20 30 40 50 60 70 80 90 100
Vector elements are : 10 20 30 40 50 60 70 80 90 100
```

4. Write a program to Copy one vector's elements to another vector.

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

int main()
{
    vector<int> v{50, 40, 30, 20, 10};
    cout << "Vector v" << endl;
    for (int i = 0; i < v.size(); i++)
        cout << v.at(i) << " ";
    vector<int> v1;
    for (int i = 0; i < v.size(); i++)
        v1.push_back(v[i]);
    cout << "\nVector v1" << endl;
    for (int i = 0; i < v1.size(); i++)
        cout << v1.at(i) << " ";
    return 0;
}
```

```
Output:
Vector v
50 40 30 20 10
Vector v1
50 40 30 20 10
```

5. Find largest and smallest elements in a vector

```
#include <iostream>
#include <vector> // for create vector
#include <algorithm> // for minimum and maximum element find
using namespace std;

int main()
{
    vector<int> v{24, 15, 3, 50, 10};

    cout << "Vector v" << endl;
    for (int i = 0; i < v.size(); i++)
        cout << v.at(i) << " ";
    cout << endl;
    cout << "Smallest: " <<*min_element(v.begin(), v.end()) << endl;
    cout << "Largest: " <<*max_element(v.begin(), v.end()) << endl;
    return 0;
}
```

```
Output:
```

```
Vector v
24 15 3 50 10
Smallest: 3
Largest: 50
```

6. Write a program to reverse vector elements

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

int main()
{
    vector<int> v = {100, 90, 80, 70, 60, 50, 40, 30, 20, 10};
    for (int i = 0; i < v.size(); i++)
        cout << v.at(i) << " ";

    reverse(v.begin(), v.end());
    cout << endl;
    for (int i = 0; i < v.size(); i++)
        cout << v.at(i) << " ";

    return 0;
}
```

```
================================================================================
Output:
100 90 80 70 60 50 40 30 20 10
10 20 30 40 50 60 70 80 90 100
```

7. Write a program to find sum of vector elements

```cpp
#include <iostream>
#include <vector>
#include <numeric> // using accumulate function for sum of elements
using namespace std;

int main()
{
    vector<int> v = {100, 90, 80, 70, 60, 50, 40, 30, 20, 10};

    for (int i = 0; i < v.size(); i++)
        cout << v.at(i) << " ";

    cout << endl;

    cout << "Sum is " << accumulate(v.begin(), v.end(), 0) << endl;

    return 0;
}
```

```
================================================================================
Output:
100 90 80 70 60 50 40 30 20 10
Sum is 550
```

8. Write a program to find common elements between two vectors.

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

int main()
{
    vector<int> v = {100, 90, 80, 70, 60, 50, 40, 30, 20, 10};
```

```
    vector<int> v1 = {1, 90, 45, 94, 60, 78, 40, 97, 24, 10};

    for (int i = 0; i < v.size(); i++)
        cout << v.at(i) << " ";

    cout << endl;

    for (int i = 0; i < v1.size(); i++)
        cout << v1.at(i) << " ";

    sort(v.begin(), v.end());
    sort(v1.begin(), v1.end());

    vector<int> v2(v.size() + v1.size());
    vector<int>::iterator it, st;
    it = set_intersection(v.begin(), v.end(), v1.begin(), v1.end(),
v2.begin());

    cout << "\nCommon elements: ";
    for (st = v2.begin(); st != it; st++)
        cout << *st << ", ";

    return 0;
}
================================================================================
Output:
100 90 80 70 60 50 40 30 20 10
1 90 45 94 60 78 40 97 24 10
Common elements: 10, 40, 60, 90,
```

9. Write a program to Push and print elements in a float vector

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

int main()
{
    vector<float> v;
    float n;
    cout << "Enter 10 numbers: ";
    for (int i = 0; i < 10; i++)
    {
        cin >> n;
        v.push_back(n);
    }
    cout << "\nElements are: ";
    for (int i = 0; i < 10; i++)
        cout << v.at(i) << " ";

    return 0;
}
================================================================================
Output:
Enter 10 numbers: 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8 9.9 10.10
```

```
Elements are: 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8 9.9 10.1
```

10. Write a program to check whether an element exists in a vector or not.

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

int main()
{
    vector<int> v;
    int n, number;
    cout << "Enter 10 numbers: ";
    for (int i = 0; i < 10; i++)
    {
        cin >> n;
        v.push_back(n);
    }
    cout << "Enter a number to check whether an element exists in a vector or
not: ";
    cin >> number;

    vector<int>::iterator it = find(v.begin(), v.end(), number);

    if (it != v.end())
        cout << "Element exists in a vector" << endl;
    else
        cout << "Element does not found" << endl;

    return 0;
}
```

```
================================================================================
Output:
Enter 10 numbers: 1 2 3 4 5 6 7 8 9 10
Enter a number to check whether an element exists in a vector or not: 25
Element does not found
```