Vector

- C++ vector is a dynamic array that stores collection of elements of same type in contiguous memory.
- It has the ability to resize itself automatically when an element is inserted or deleted.

Inserting elements

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

int main() {
    vector<char> v = {'a', 'f', 'd'};

        // Inserting 'z' at the back
        v.push_back('z');

        // Inserting 'c' at index 1
        v.insert(v.begin() + 1, 'c');

        for (int i = 0; i < v.size(); i++)
      cout << v[i] << " ";
    return 0;
}
```

Output

a c f d z

Finding vector size

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

int main() {
    vector<char> v = {'a', 'c', 'f', 'd', 'z'};

    // Finding size
    cout << v.size();

    return 0;
}
```

Output

5

Updating elements

```
#include <iostream>
#include <vector>
using namespace std;
int main() {
    vector<char> v = {'a', 'c', 'f', 'd', 'z'};

    // Accessing and printing values
            cout << v[3] << endl;
            cout << v.at(2) << endl;

  // Updating values using indexes 3 and 2
            v[3] = 'D';
            v.at(2) = 'F';

            cout << v[3] << endl;
            cout << v.at(2);
    return 0;
}
```

Output

d f D F

Traversing

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

int main() {
    vector<char> v = {'a', 'c', 'f', 'd', 'z'};

    // Traversing vector using range based for loop
            for (int i = 0; i < v.size(); i++)
        cout << v[i] << " ";
    return 0;
}
```

Output

a c f d z

Delete elements

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

int main() {
    vector<char> v = {'a', 'c', 'f', 'd', 'z'};

    // Deleting last element 'z'
            v.pop_back();

            // Deleting element 'f'
            v.erase(find(v.begin(), v.end(), 'f'));

            for (int i = 0; i < v.size(); i++) {
        cout << v[i] << " ";
    }
    return 0;
}
```

Output

a c d

C++ STL Sorting Algorithm

```
#include <iostream>
#include <algorithm>
#include <vector>

using namespace std;

int main() {

   // initialize vector of int type
   vector<int> v = {3, 2, 5, 1, 4};

   // sort vector elements in ascending order
   sort(v.begin(), v.end());

   // print the sorted  vector
   for (int i=0;i<v.size();i++) {
      cout << v[i] << "  ";
   }
   return 0;
}
```

Output

1 2 3 4 5

C++ STL Sorting Algorithm

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

int main() {
    // initialize vector of int type
    vector<int> v = {3, 2, 5, 1, 4};

    // sort vector elements in descending order
    sort(v.begin(), v.end(), greater<int>());

    // print the sorted vector
    for (int i = 0; i < v.size(); i++) {
        cout << v[i] << " ";
    }

    return 0;
}
```

Output

5 4 3 2 1

Inserting in 2D vector

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    vector<vector<int>> v = {{1, 2, 3},{4, 5, 6}};

    // Insert a new row at the end
    v.push_back({7, 8, 9});

    // Insert value in 2nd row at 2nd position
    v[1].insert(v[1].begin() + 1, 10);

    for (int i = 0; i < v.size(); i++) {
        for (int j = 0; j < v[i].size(); j++) {
            cout << v[i][j] << " ";
        }
        cout << endl;
    }
    return 0;
}
```

Output

```
1 2 3
4 10 5 6
 7 8 9
```