

Vector

1

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    //Demo vector (array representation)
    //vector<int> arr = {1,2,3,4,5};

    //fill constructor
    std::vector<int> arr(10,7); //here we will get 10 value vector with
    initialized with 7

    //pop back => it is going to remove element 5
    arr.pop_back();

    // Push_back 0(1)
    arr.push_back(15);

    //print all the elements
    for (int i = 0; i < arr.size(); ++i)
    {
        cout << arr[i] << " ";
    }
    cout << endl;
    for(int x : arr){
        cout << x << " ";
    }

    //size of the vector
    cout << arr.size() << endl;

    //capacity of the vector
    cout << arr.capacity() << endl;
}
```

2

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    //2D array
    std::vector<std::vector<int>> arr = {
        {1,2,3},
        {1,2,3,4},
        {1,2,3},
        {1,2}
    };
}
```

```

        //vector<vector<int>> v;

        arr[0][0] += 10; //update

//printing the element
    for (int i = 0; i < arr.size(); ++i)
    {
        for(int number :arr[i]){
            cout << number << " ";
        }
        cout << endl;
    }
}

```

3

```

//vector data structure
#include <bits/stdc++.h>
#include "vector.h"
using namespace std;

/*
class Vector{
    //data members
    int *arr;
    int cs; //currently we have only 5 element
    int ms; //maximum size

    //Methods, constructor , destructor

public:
    Vector(int max_size = 1){
        cs = 0;
        ms = max_size;
        arr = new int[ms];
    }
    void push_back(int d){
        //Two cases
        if(cs==ms){
            //create a new array and delete the old one,double the capacity
            int *oldArr = arr;
            ms = ms*2;
            arr = new int[ms];
            //copy the elements
            for (int i = 0; i < cs ; ++i)
            {

```

```

        arr[i] =oldArr[i];
    }
    //delete the old array
    delete [] oldArr;
}
arr[cs] = d;
cs++;
}

void pop_back(){
    if(cs >= 0){
        cs--;
    }
}

//Front , Back ,At(i) ,[]
int front(){
    return arr[0];
}

int back(){
    return arr[cs-1];
}

int at(int i){
    return arr[i];
}

int size(){
    return cs;
}

int capacity(){
    return ms;
}

int operator[](int i){ //operator overloading
    return arr[i];
}

}; */

int main(){
    Vector<char> vc;
    vc.push_back('a');
    vc.push_back('b');
    for (int i = 0; i < vc.size(); ++i)

```

```

{
    cout << vc[i] << endl;
}

Vector<int> v;
v.push_back(1);
v.push_back(2);
v.push_back(33);
v.push_back(44);
v.push_back(55);
v.pop_back();

cout << v.front() << endl;
cout << v.back() << endl;

cout << v.size() << endl;
cout << v.capacity() << endl;

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

```

4

```

//find or search function in the stl vector
//this is linear search not a binary search
//visit cplusplus.com for more knowledge about the method
#include <bits/stdc++.h>
using namespace std;
int main(){
    std::vector<int> v = {10,12,2,12,3,32,33};

    int key;
    cin >> key;

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

    //cout << it; //returning the address
    //it is possible that element is not present

    if(it!=v.end()){
        cout <<"Present at index " <<  it - v.begin();
    }
    else{
        cout << "Element is not found";
    }
}

```

```
}  
}
```

5

```
//sorting a complex vector  
#include <bits/stdc++.h>  
using namespace std;  
int calTotalMarks(vector<int> marks){  
    return marks[0] +marks[1] + marks[2];  
}  
bool compare(pair<string,vector<int>> s1, pair<string,vector<int>> s2){  
    vector<int> m1 = s1.second;  
    vector<int> m2 = s2.second;  
    return calTotalMarks(m1) > calTotalMarks(m2);  
}  
int main(){  
    vector<pair<string,vector<int>>> student_marks = {  
        {"ashish",{90,80,85}},  
        {"mayank",{88,99,77}},  
        {"dip",{97,87,86}}  
    };  
  
    sort(student_marks.begin(),student_marks.end(),compare); //by default it  
    is going to take first para as sort  
  
    for(auto s: student_marks){  
        cout << s.first << " " << calTotalMarks(s.second) << endl;  
    }  
}
```

6

```
#include <bits/stdc++.h>  
using namespace std;  
int main(){  
    std::vector<int > v (10,2);  
    v.push_back(22);  
    for (int i = 0; i < v.size(); ++i)  
    {  
        cout << v[i] << " ";  
    }  
}
```