

VECTOR:

Vectors are the same as dynamic arrays with the ability to resize itself automatically when an element is inserted or deleted, with their storage being handled automatically by the container. Vector elements are placed in contiguous storage so that they can be accessed and traversed using iterators. In vectors, data is inserted at the end. Inserting at the end takes differential time, as sometimes the array may need to be extended. Removing the last element takes only constant time because no resizing happens. Inserting and erasing at the beginning or in the middle is linear in time.

Example 1:

```
#include<iostream>

using namespace std;

#include<vector>
#include<iterator>

int main()
{
    vector<int>v;
    v.push_back(101);
    v.push_back(102);
    v.push_back(103);
    v.push_back(104);
    v.push_back(105);
    cout<<"size="<<v.size()<<endl;
    cout<<"capacity="<<v.capacity()<<endl;
    for(int i=0;i<v.size();i++)
    {
        cout<<v[i]<<endl;
    }
    cout<<"front="<<v.front()<<endl;
    cout<<"back="<<v.back()<<endl;
    vector<int>::iterator it=v.end();
    v.pop_back();
    cout<<"pop"<<endl;
    for(int i=0;i<v.size();i++)
```