STL Vectors in C++

Arrays: A collection of variables of same data-type. It has a **fixed (or static) size**

```
int arr[100];
// an array of size 100
```

Vectors are arrays with **dynamic size.** Vectors are provided by STL library of C++

When you use vectors ?
When you don't know the size of the array required beforehand, you use vectors.

In #include<bits/stdc++.h> , vectors are already included. So, you don't need to include any extra header file for using vectors.

Syntax for declaring vector:

vector <data-type> name; // for an empty
vector <data-type> name(size);
Eg.

```
vector<int> vec; // vec is empty
```

```
vector<int> vec(1000); // vec is a vector of size 1000
```

.size()
Time complexity of O(1)

```
cout<<vec.size();
```

Accessing a particular element of vector

0-based indexing

vec[0] => 1st element of vector
vec[1] => 2nd element of vector
vec[2] => 3rd element of vector

…..
vec[n-1] => nth element of vector

Declaring vector with some default value

vector<data-type> name(size, defaultValue);

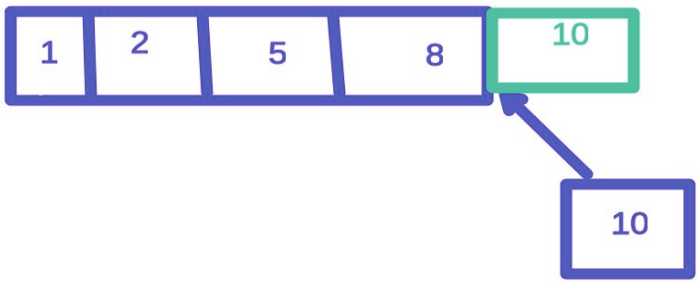
Eg. `vector<int> vec(5,2);`
{ 2 , 2, 2, 2, 2 }

Printing all the values of a vector

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

.push_back()

Pushes a value at the back of vector and increases size of vector by 1



Syntax:

```
vec.push_back(value);
```

Time complexity:

O(1)

Eg. 1 A code using push_back():

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

int32_t main()
{
    vector<int> vec(3, 8);
    vec.push_back(22);
    int n=vec.size();
    for(int i=0; i<n; i++)
    {
       cout<<vec[i]<<" ";
    }

    return 0;
}
```

Eg. 2 Difference between changing value using vec[i] and push_back():

```
vector<int> vec={2, 3, 4, 6, 9};
```

Case I: (Changing value using vec[i])

```
vec[2]=99;
```

// New value of vec: { 2, 3, 99, 6, 9}

Case II: (Inserting element using push_back())

```
vec.push_back(100);
```

// New value of vec: { 2, 3, 4, 6, 9, 100);

.resize()

Resize the vector to a different size

Eg.

```
vec.resize(100);
```

// Make the size of vector 100

.sort()

// Used to arrange all elements of vector in increasing or ascending order

// O(N log N)

```
sort(vec.begin(), vec.end());
```

.reverse()

// Used to reverse a vector

// O(N)

```
reverse(vec.begin(), vec.end());
```

Q. Suppose, the input format is given like this:
All elements are given as space separated integers and last element is -1. Write program to take input of these integers.

```
int num=100;
vector<int> vec;
while(num != -1)
{
 cin>>num;
 vec.push_back(num);
}
```

For auto loop to print all elements:

```
for(auto num: vec)
{
   cout<<num<<" ";
}
```

Note: But using this syntax, you can can't change any value of the original array. (Similar to call by value)

If you want to change all the elements of the array, use reference variable by **putting & sign before variable name** in loop (Similar to call by reference)

Example

```
for(auto &num: vec)
{
    num=100;
}
```