Some simple ways to define the size and initialize a vector

Define the size of a vector like array

We can define the size of a vector while declaring a vector. We can simply write this:

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

int main()
{
   int n = 5;
   vector<int> v(n);
   for ( int i = 0; i < v.size(); i++ ) cout << v[i] << " "; // 0 0 0 0 0
   cout << endl;
}
```

We will simply declare a vector and in parentheses we will define the size. After declaring our vector will be like this **v = { 0, 0, 0, 0, 0 }.**

We defined the size of the vector 5. So it has created 5 32bit spaces dynamically for indexes and then filled every index with 0. Now can access and change any index we want. For example we can increment every index by 1.

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

int main()
{
   int n = 5;
   vector<int> v(n);
   for ( int i = 0; i < v.size(); i++ ) v[i]++;
   for ( int i = 0; i < v.size(); i++ ) cout << v[i] << " "; // 1 1 1 1 1
   cout << endl;
}
```

Look we have incremented all elements by 1 and our vector look like this **v = { 1, 1, 1, 1, 1 }.**

Initialize a vector with n elements of any value using assign() function

Previously we have initialized a vector with n number of 0's. But we can initialize a vector with n number of any value. For example :

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

int main()
{
   int n = 5;
   vector<int> v;
   v.assign ( n, 3 );
   for ( int i = 0; i < v.size(); i++ ) cout << v[i] << " "; // 3 3 3 3 3
   cout << endl;
}
```

Look assign() function takes 2 parameters. First is the size of the vector, second is the target value. We have passed 5 as size and 3 as target value. So it created 5 32bit spaces for the indexes and then filled every index with 3. So our vector looks like this **v = { 3, 3, 3, 3, 3 }.**

Initialize a vector with n elements of any value without assign() function

We can initialize a vector with n elements in a more simple way without using assign() function. We can simply do this:

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

int main()
{
   int n = 5;
   vector<int> v(n, 3);
   for ( int i = 0; i < v.size(); i++ ) cout << v[i] << " "; // 3 3 3 3 3
   cout << endl;
}
```

Yes we will simply declare a vector and in parenthesis we will first define the size of the vector and then the target value. We have passed 5 as size and 3 as target value. So it created 5 32bit spaces for the indexes and then filled every index with 3. So our vector looks like this **v = { 3, 3, 3, 3, 3 }.**

Initialize a vector like array

We can also initialize a vector the same as array. For example:

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

int main()
{
   int a[] = {1, 2, 3, 4};

   vector<int> v;
   v = { 1, 2, 3, 4 };

   //or

   vector<int> v1 = {1, 2, 3, 4};

   //or

   vector<int> v2 {1, 2, 3, 4};
}
```

We have shown 3 types of initialization above code. For vector **v** we have first declared it and then assigned a list of numbers. For vector **v1** we have assigned a list of numbers while declaring the vector. For vector **v2** we have passed a list of numbers to initialize a vector. You can use any type you want.

So these are some simple initialization and declaration of vectors. There are some other complex and powerful initialization or assignment operations. In Sha ALLAH we will cover them in upcoming lessons.

So that’s all for today's lesson. Hope you enjoyed the lesson. See you in next lesson.