**ASSIGNMENT NUMBER 6**

**STATEMENT**: **Create a class template to represent a generic vector. Include following member functions:  
i. To create the vector.  
ii. To modify the value of a given element   
iii. To multiply by a scalar value  
iv. To display the vector in the form (10,20,30,…)**

**AIM**:To create a class template to represent a generic vector.

**DESCRIPTION: Generics is the idea to allow type (Integer, String, … etc and user-defined types) to be a parameter to methods, classes and interfaces. For example, classes like an array, map, etc, which can be used using generics very efficiently.**

**OOP CONCEPT USED**:

1. Class Vector : **A vector is a sequence container class that implements dynamic array, means size automatically changes when appending elements. A vector stores the elements in contiguous**

**memory locations and allocates the memory as needed at run time.**

**SOURCE CODE:**

#include<iostream>

#include<iterator>

#include<vector>

using namespace std;

void create(vector<int> &v)

{

int x,y;

cout<<"Enter the number of elements in the vector "<<endl;

cin>>x;

for(int i=0;i<x;i++)

{

cout<<"Enter values"<<endl;

cin>>y;

v.push\_back(y);

}

}

vector<int> modify(vector<int> &v)

{

int i,j;

cout<<"Enter the position of the element you want to modify :"<<endl;

cin>>i;

cout<<"Enter Replacement value:"<<endl;

cin>>j;

vector<int>::iterator itr;

itr = v.begin();

itr = itr + i - 1;

\*itr = j;

cout<<"Value Modified!"<<endl;

return v;

}

vector<int> del(vector<int> &v)

{

int p;

cout<<"Enter the position of the element you want to delete"<<endl;

cin>>p;

vector<int>::iterator it = v.begin();

it = it + p -1;

v.erase(it);

cout<<"Vector element deleted!";

return v;

}

void display(vector<int> &v)

{

cout<<"('";

for(int i=0;i<v.size()-1;i++)

{

cout<<v[i]<<",";

}

cout<<v[v.size()-1];

cout<<"')\n";

}

vector<int> multiply(vector<int> &v)

{

int m;

cout<<"Enter the value to multiply: "<<endl;

cin>>m;

for(int k=0;k<v.size();k++)

{

v[k] \*= m;

}

cout<<"Vector Multiplied!\n"<<endl;

return v;

}

int main()

{

vector<int> v;

int ch;

while(ch != 7)

{

cout<<"1.Create\n2.Modify\n3.Multiply\n4.Display\n5.Delete\n6.Exit\n";

cout<<"\nEnter a choice\n";

cin>>ch;

switch(ch)

{

case 1:create(v);

break;

case 2:v = modify(v);

break;

case 3:v = multiply(v);

break;

case 4:display(v);

break;

case 5:v = del(v);

break;

//case 6:v.clear();

// cout<<"All the elements of the vector have been destroyed!"<<endl;

// v.assign(0);

// break;

case 6:cout<<"Program Terminated!";

break;

}

}

}

**CONCLUSION**: In this assignment, we learned the template class and generic vector.