**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.**

**Source Code:**

**#include<iostream>**

**using namespace std;**

**template<class T>**

**class vector**

**{**

**T v[20];**

**int size;**

**public:**

**void create();**

**void modify();**

**void mult();**

**void display();**

**};**

**template<class T>**

**void vector<T>::create()**

**{**

**int i;**

**T value;**

**char ans;**

**size=0;**

**do**

**{**

**cout<<"\nEnter the index position & value:";**

**cin>>i>>value;**

**v[i]=value;**

**size++;**

**cout<<"\nDo you want to insert more elements (y/n)?";**

**cin>>ans;**

**}**

**while(ans=='y'||ans=='Y');**

**}**

**template<class T>**

**void vector<T>::modify()**

**{**

**int key;**

**T newval;**

**cout<<"\nEnter index you want to modify:";**

**cin>>key;**

**cout<<"\nEnter new value:";**

**cin>>newval;**

**v[key]=newval;**

**}**

**template<class T>**

**void vector<T>::mult()**

**{**

**int i;**

**int scalarval;**

**cout<<"\nEnter scalar value for multiplication :";**

**cin>>scalarval;**

**for(i=0;i<size;i++)**

**v[i]=v[i]\*scalarval;**

**}**

**template<class T>**

**void vector<T>::display()**

**{**

**int i;**

**cout<<"\nSize of vector is:"<<size;**

**cout<<"\nElements in vector are:";**

**cout<<"(";**

**for(i=0;i<size;i++)**

**{**

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

**}**

**cout<<")";**

**}**

**int main()**

**{**

**int ch;**

**vector<int>obj;**

**do**

**{**

**cout<<"\n1.Create";**

**cout<<"\n2.Display";**

**cout<<"\n3.Mult";**

**cout<<"\n4.Modify";**

**cout<<"\n0.Exit";**

**cout<<"\nEnter your choice:";**

**cin>>ch;**

**switch(ch)**

**{**

**case 1:**

**obj.create();**

**break;**

**case 2:**

**obj.display();**

**break;**

**case 3:**

**obj.mult();**

**break;**

**case 4:**

**obj.modify();**

**break;**

**case 0:**

**cout<<"\nExit\n";**

**break;**

**default:**

**cout<<"\nInvalid choice";**

**break;**

**}**

**}**

**while(ch!=0);**

**return 0;**

**}**

**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.**

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