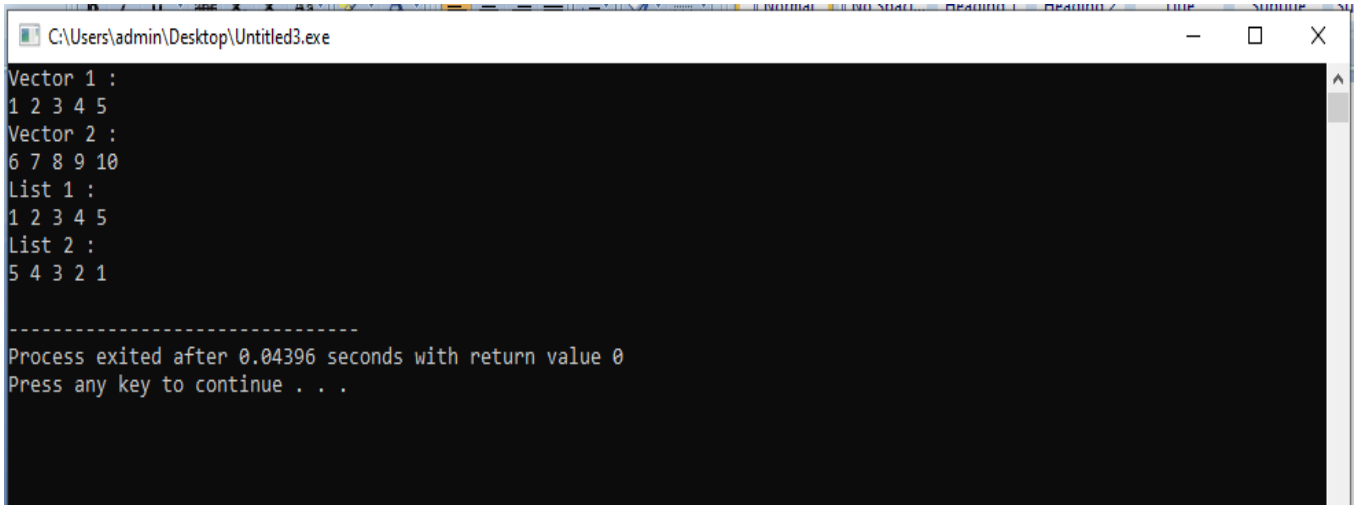
Develop a C++ program to implement List, Vector and its Operations.

PROGRAM:

```
#include<iostream>
#include<vector>
#include<list>
using namespace std;
int main()
{
	vector<int>ivec1,ivec2;
	list<int>l1,l2;
	for(int i=1;i<=5;i++)
	{
		ivec1.push_back(i);
	}
	for(int i=6;i<=10;i++)
	{
	   ivec2.push_back(i);
	}
	vector<int>::iterator it1;
	cout<<"Vector 1 :"<<endl;
	for(it1=ivec1.begin();it1!=ivec1.end();++it1)
	{
		cout<<*it1<<" ";
	}
	cout<<endl;
	vector<int>::iterator it2;
	cout<<"Vector 2 :"<<endl;
	for(it2=ivec2.begin();it2!=ivec2.end();++it2)
	{
		cout<<*it2<<" ";
	}
	cout<<endl;
	for(int i=1;i<=5;i++)
	{
		l1.push_back(i);
	}
	for(int i=1;i<=5;i++)
	{
	   l2.push_front(i);
	}
	list<int>::iterator it3;
```

```
        cout<<"List 1 :"<<endl;
        for(it3=l1.begin();it3!=l1.end();++it3)
        {
                cout<<*it3<<" ";
        }
        cout<<endl;
        list<int>::iterator it4;
        cout<<"List 2 :"<<endl;
        for(it4=l2.begin();it4!=l2.end();++it4)
        {
                cout<<*it4<<" ";
        }
        cout<<endl;
        cout<<"Mering of Two lists"<<endl;
        l1.merge(l2);
        for(it3=l1.begin();it3!=l1.end();++it3)
        {
                cout<<*it3<<" ";
        }
        cout<<endl;
        return 0;
}
```

OUTPUT:

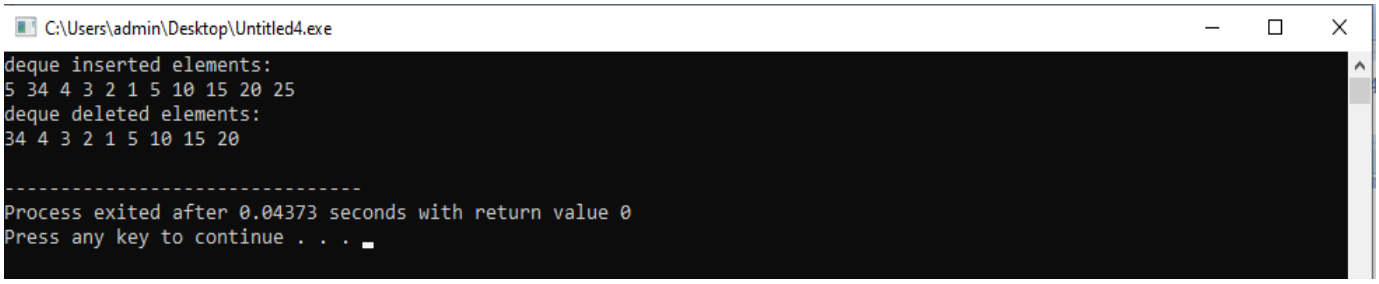


Develop a C++ program to implement Deque and Deque Operations.

PROGRAM:

```
#include<iostream>
#include<deque>
using namespace std;
int main()
{
	deque<int>deq;
	for(int i=1;i<=5;i++)
	{
		deq.push_front(i);
		deq.push_back(i*5);
	}
	deque<int>::iterator d;
	d=deq.begin();
	++d;
	deq.insert(d,1,34);
	cout<<"deque inserted elements: "<<endl;
	for(d=deq.begin();d!=deq.end();d++)
	{
		cout<<*d<<" ";
	}
	cout<<endl;
	deq.pop_front();
	deq.pop_back();
	cout<<"deque deleted elements: "<<endl;
	for(d=deq.begin();d!=deq.end();d++)
	{
		cout<<*d<<" ";
	}
	cout<<endl;
	return 0;
}
```

OUTPUT:



Develop a C++ program to implement Map and Map Operations.

PROGRAM:

```
#include<iostream>
#include<map>
#include<string>
using namespace std;
int main()
{
	map<int,string>mp;
	mp.insert(make_pair(2805,"JAMES"));
	mp.insert(make_pair(2802,"SMITH"));
	mp.insert(make_pair(2801,"SCOTT"));
	mp.insert(make_pair(2807,"MILLER"));
	mp.insert(make_pair(2804,"JOY"));
	mp.insert(make_pair(2806,"JOHN"));
	mp.insert(make_pair(2801,"SAM"));
	mp.insert(make_pair(2803,"BOB"));
	map<int,string>::iterator pos;
	cout<<"EMPID"<<" "<<"NAME"<<endl;
	for(pos=mp.begin();pos!=mp.end();++pos)
	{
		cout<<pos->first<<" "<<pos->second<<endl;
	}
	pos=mp.find(2803);
	cout<<"Value of key 2803 is "<<pos->second<<endl;
	mp.erase(2805);
	cout<<"Map after deletion "<<endl;
	for(pos=mp.begin();pos!=mp.end();++pos)
	{
		cout<<pos->first<<" "<<pos->second<<endl;
	}
	return 0;
}
```

OUTPUT:


C:\Users\admin\Desktop\Untitled5.exe
EMPID NAME
2801 SCOTT
2802 SMITH
2803 BOB
2804 JOY
2805 JAMES
2806 JOHN
2807 MILLER
Value of key 2803 is BOB
Map after deletion
2801 SCOTT
2802 SMITH
2803 BOB
2804 JOY
2806 JOHN
2807 MILLER