```
//DMA case(Dangling pointer)
#include <iostream>
using namespace std;
int main()
{
   int n;
   cout<<"\nEnter number of values you want to work with:";
   cin>>n;
   int *ptr=new int[n];
   cout<<"\nEnter" <<n <<"values:";
   for(int i=0;i<n;i++)
   {
      cin>>*(ptr+i);
   }
   cout<<"\nEntered values are:";
   for(int i=0;i<n;i++)
   {
      cout<<*(ptr+i)<<" ";
   }
   cout<<"\nAddress containted in pointer before memory deallocation:"<<ptr;
   delete [] ptr;
   cout<<"\nAddress contained in pointer after memory deallocation:"<<ptr;//Same address is
printed(Dangling pointer)
   return 0;
}
//DMA case(Dangling pointer)-Solution
#include <iostream>
using namespace std;
int main()
{
   int n;
   cout<<"\nEnter number of values you want to work with:";
   cin>>n;
   int *ptr=new int[n];
   cout<<"\nEnter" <<n <<"values:";
   for(int i=0;i<n;i++)
   {
      cin>>*(ptr+i);
   }
   cout<<"\nEntered values are:";
   for(int i=0;i<n;i++)
   {
      cout<<*(ptr+i)<<" ";
   }
   cout<<"\nAddress containted in pointer before memory deallocation:"<<ptr;
   delete [] ptr;
   ptr=NULL;
   cout<<"\nAddress contained in pointer after memory deallocation:"<<ptr;//0 will be printed(No longer
a dangling pointer)
```

```
   return 0;
}
//Pointer to pointer
#include<iostream>
using namespace std;
int main()
{
	int *ptr,**ptr1;
	int a=10;
	ptr=&a;
	cout<<*ptr<<endl;//It will display value of a
	ptr1=&ptr;
	cout<<**ptr1<<endl;//It will display value of a
	cout<<"Address of a is:"<<*ptr1<<" "<<ptr<<" "<<&a<<endl;//It will display address of a
	cout<<"Address of ptr is:"<<ptr1<<" "<<&ptr<<endl;//It will display address of ptr
	cout<<"Address of ptr1 is:"<<&ptr1;
	return 0;
}
#include<iostream>
using namespace std;
int main()
{
	int a[100],n,i,max;
	int *p=a;
	cout<<"\nEnter n:";
	cin>>n;
	cout<<"\nEnter array elements:";
	for(int i=0;i<n;i++)
	{
	   cin>>*(p+i);
	}
	max=*(p+0);
	for(int i=1;i<n;i++)
	{
	   if(*(p+i)>max)
	   {
	      max=*(p+i);
	   }
	}
	cout<<"\nLargest element is:"<<max;
	return 0;
}
#include<iostream>
using namespace std;
int main()
{
	int a[100],n,i,max;
	int *p=a;
	cout<<"\nEnter n:";
```

```
    cin>>n;
    cout<<"\nEnter array elements:";
    for(int i=0;i<n;i++)
    {
       cin>>p[i];//i[p] or *(i+p) or *(p+i)
    }
    max=p[0];
    for(int i=1;i<n;i++)
    {
       if(p[i]>max)
       {
          max=p[i];
       }
    }
    cout<<"\nLargest element is:"<<max;
    return 0;
}
```