```
//WAP to allocate/ or deallocate memory for one memory location
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
   int *ptr=NULL;
   ptr=new int;
   if(!ptr)
   {
      cout<<"\nMemory allocation failure";
      exit(1);
   }
   else
   {
      cout<<"\nMemory allocation was successful";
      cout<<"\nEnter integer value:";
      cin>>*ptr;
      cout<<"\nSquare of integer value is:"<<(*ptr)*(*ptr);
      delete ptr;
      cout<<"\nMemory deallocated successfully";
   }
   return 0;
}
```

```
//WAP to allocate/ or deallocate memory for one memory location
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
   int *ptr=NULL;
   ptr=new int(12);//We are initializing 12 to one integer memory location
   if(!ptr)
   {
      cout<<"\nMemory allocation failure";
      exit(1);
   }
   else
   {
      cout<<"\nMemory allocation was successful";
      //cout<<"\nEnter integer value:";
      //cin>>*ptr;
      cout<<"\nSquare of integer value is:"<<(*ptr)*(*ptr);
      delete ptr;
      cout<<"\nMemory deallocated successfully";
   }
   return 0;
}
```

```
-----------------------------------
//WAP to allocate/ or deallocate memory for one memory location
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
   float *ptr=NULL;
   ptr=new float;
   if(!ptr)
   {
      cout<<"\nMemory allocation failure";
      exit(1);
   }
   else
   {
      cout<<"\nMemory allocation was successful";
      cout<<"\nEnter float value:";
      cin>>*ptr;
      cout<<"\nArea of the circle is:"<<3.14*(*ptr)*(*ptr);
      delete ptr;
      cout<<"\nMemory deallocated successfully";
   }
   return 0;
}
--------------------------------------
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
   int *a=new int;
   int *b=new int;
   int *temp=new int;
   if(!a||!b||!temp)
   {
      cout<<"\nMemory allocation failure";
      exit(1);
   }
   else
   {
      cout<<"\nMemory allocation was successful";
      cout<<"\nEnter value of a:";
      cin>>*a;
      cout<<"\nEnter value of b:";
      cin>>*b;
      *temp=*a;
      *a=*b;
      *b=*temp;
```

```
        cout<<"\nValues after swapping are:"<<*a<<" "<<*b;
        delete a;
        delete b;
        delete temp;
        cout<<"\nMemory deallocated successfully";
    }
    return 0;
}
---------------------------------------
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
    int *n=new int;
    if(!n)
    {
        cout<<"\nMemory allocation failure";
        exit(1);
    }
    else
    {
        cout<<"\nMemory allocation was successful";
        cout<<"\nEnter integer value:";
        cin>>*n;
        if(*n==0)
        {
            cout<<"\nEntered integer is zero";
        }
        else if(*n>0)
        {
            cout<<"\nEntered integer is +ve";
        }
        else
        {
            cout<<"\nEntered integer is -ve";
        }
        delete n;
        cout<<"\nMemory deallocated successfully";
    }
    return 0;
}
-------------------------------------
//Allocating/ deallocating array of memory locations
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
```

```
    int size;
    cout<<"\nEnter number of integer values to work with:";
    cin>>size;
    int *ptr=new int[size];
    if(!ptr)
    {
        cout<<"\nMemory allocation failure";
        exit(1);
    }
    else
    {
        cout<<"\nMemory allocation was successful";
        cout<<"\nEnter array values:";
        for(int i=0;i<size;i++)
        {
            cin>>*(ptr+i);
        }
        cout<<"\nEntered values are:";
        for(int i=0;i<size;i++)
        {
            cout<<*(ptr+i)<<" ";
        }
        delete [] ptr;
        cout<<"\nMemory deallocated successfully";
    }
    return 0;
}
-------------------------------------
//Allocating/ deallocating array of memory locations
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
    int size,sum=0;
    cout<<"\nEnter number of integer values to work with:";
    cin>>size;
    int *ptr=new int[size];
    if(!ptr)
    {
        cout<<"\nMemory allocation failure";
        exit(1);
    }
    else
    {
        cout<<"\nMemory allocation was successful";
        cout<<"\nEnter array values:";
        for(int i=0;i<size;i++)
        {
```

```
            cin>>*(ptr+i);
            sum=sum+*(ptr+i);
        }
        cout<<"\nSum is:"<<sum;
        delete [] ptr;
        cout<<"\nMemory deallocated successfully";
    }
    return 0;
}
```

```
//Allocating/ deallocating array of memory locations
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
    int size;
    double avg=0.0,sum=0.0;
    cout<<"\nEnter number of double values to work with:";
    cin>>size;
    double *ptr=new double[size];
    if(!ptr)
    {
        cout<<"\nMemory allocation failure";
        exit(1);
    }
    else
    {
        cout<<"\nMemory allocation was successful";
        cout<<"\nEnter array values:";
        for(int i=0;i<size;i++)
        {
            cin>>*(ptr+i);
            sum=sum+*(ptr+i);
        }
        avg=sum/size;
        cout<<"\nAverage is:"<<avg;
        delete [] ptr;
        cout<<"\nMemory deallocated successfully";
    }
    return 0;
}
```

```
//Allocating/ deallocating array of memory locations
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
```

```
   int size,max,min;
   cout<<"\nEnter number of integer values to work with:";
   cin>>size;
   int *ptr=new int[size];
   if(!ptr)
   {
      cout<<"\nMemory allocation failure";
      exit(1);
   }
   else
   {
      cout<<"\nMemory allocation was successful";
      cout<<"\nEnter array values:";
      for(int i=0;i<size;i++)
      {
         cin>>*(ptr+i);
      }
      max=*ptr;
      min=*ptr;
      for(int i=1;i<size;i++)
      {
         if(ptr[i]>max)
         {
            max=ptr[i];
         }
         else if(ptr[i]<min)
         {
            min=ptr[i];
         }
      }
      cout<<"\nLargest element is:"<<max;
      cout<<"\nSmallest element is:"<<min;
      delete [] ptr;
      cout<<"\nMemory deallocated successfully";
   }
   return 0;
}
---------------------------------------
//Allocating/ deallocating array of memory locations
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
   int size,rev=0,digit;
   cout<<"\nEnter number of integer values to work with:";
   cin>>size;
   int *ptr=new int[size];
   if(!ptr)
```

```
    {
        cout<<"\nMemory allocation failure";
        exit(1);
    }
    else
    {
        cout<<"\nMemory allocation was successful";
        cout<<"\nEnter array values:";
        for(int i=0;i<size;i++)
        {
            cin>>*(ptr+i);
        }
        for(int i=0;i<size;i++)
        {
            if(ptr[i]%5==0)
            {
                cout<<"\nReversed multiples of 5 elements are:";
                while(ptr[i]!=0)
                {
                    digit=ptr[i]%10;
                    rev=rev*10+digit;
                    ptr[i]=ptr[i]/10;
                }
                cout<<rev<<" ";
            }
        }
        delete [] ptr;
        cout<<"\nMemory deallocated successfully";
    }
    return 0;
}
---------------------------------------
//Allocating/ deallocating dynamic memory inside class
#include <iostream>
#include <stdlib.h>
using namespace std;
class example
{
    int *ptr,size;
    public:
    void input()
    {
        cout<<"\nEnter number of integer values to work with:";
        cin>>size;
        ptr=new int[size];
        if(!ptr)
        {
            cout<<"\nMemory allocation failure";
            exit(1);
```

```
        }
        else
        {
           cout<<"\nMemory was successfully allocated";
           cout<<"\nEnter group of integer values:";
           for(int i=0;i<size;i++)
           {
              cin>>ptr[i];
           }
           cout<<"\nEntered values are:";
           for(int i=0;i<size;i++)
           {
              cout<<ptr[i]<<" ";
           }
        }
    }
    ~example()
    {
       delete []ptr;
       cout<<"\nMemory deallocated successfully";
    }
};
int main()
{
    example obj;
    obj.input();
    return 0;
}
-----------------------------------
//Allocating/ deallocating dynamic memory inside class
#include <iostream>
#include <stdlib.h>
using namespace std;
class example
{
    double *ptr,avg=0.0,sum=0.0;
    int size;
    public:
    void input()
    {
       cout<<"\nEnter number of double values to work with:";
       cin>>size;
       ptr=new double[size];
       if(!ptr)
       {
          cout<<"\nMemory allocation failure";
          exit(1);
       }
       else
```

```
        {
            cout<<"\nMemory was successfully allocated";
            cout<<"\nEnter group of double values:";
            for(int i=0;i<size;i++)
            {
                cin>>ptr[i];
            }
        }
    }
        double average()
        {
            for(int i=0;i<size;i++)
            {
                sum=sum+ptr[i];
            }
            avg=sum/size;
            return avg;
        }
    ~example()
    {
        delete []ptr;
        cout<<"\nMemory deallocated successfully";
    }
};
int main()
{
    example obj;
    obj.input();
    cout<<obj.average();
    return 0;
}
```