**/\*1. C++ Program to find sum of individual digits of a positive integer \*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int main()**

**{**

**int n, m=0, sum=0;**

**clrscr();**

**cout<<"Enter a number:";**

**cin>>n;**

**while(n!=0)**

**{**

**m=n%10;**

**sum=sum+m;**

**m=n/10;**

**n=m;**

**}**

**cout<<"Sum of the digits:"<<sum;**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter a number**

**145**

**sum of digits=10**

**/\*2. C++ Program to generate first n terms of fibonacci sequence.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int main()**

**{**

**int i,n,a=0,b=1,c;**

**clrscr();**

**cout<<"Enter n value:";**

**cin>>n;**

**cout<<a<<"\t"<<b<<"\t";**

**for(i=0;i<n-2;i++)**

**{**

**c=a+b;**

**cout<<c<<"\t";**

**a=b;**

**b=c;**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter n value**

**5**

**0 1 1 2 3**

**/\* 3. C++ Program to generate prime numbers between 1 and n.\*/**

**//PROGRAM**

**#include <iostream.h>**

**#include <conio.h>**

**int main()**

**{**

**int n,i,j,count=0;**

**clrscr();**

**cout<<"enter any postive integer";**

**cin>>n;**

**for(i=2;i<=n;i++)**

**{**

**for(j=1;j<=i;j++)**

**{**

**if(i%j==0)**

**count++;**

**}**

**if(count==2)**

**cout<<i<<"\t";**

**count=0;**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter any positive number**

**9**

**2 3 5 7**

**/\* 4.a.i. C++ Program to find recursive factorial of a given integer\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int recursive(int);**

**int main()**

**{**

**int n,f;**

**clrscr();**

**cout<<"ENTER A NUMBER:";**

**cin>>n;**

**f=recursive(n);**

**cout<<"FACTORIAL is:"<<f;**

**getch();**

**return 0;**

**}**

**int recursive(int n)**

**{**

**int a;**

**if(n>0)**

**{**

**a=n\*recursive(n-1);**

**return a;**

**}**

**else**

**return 1;**

**}**

**//OUTPUT**

**ENTER A NUMBER**

**5**

**FACTORIAL is 120**

**/\* 4.a.ii. C++ Program to find no-recursive factorial of a given integer\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int non\_recursive(int);**

**int main()**

**{**

**int n,f;**

**clrscr();**

**cout<<"ENTER N VALUE:";**

**cin>>n;**

**f=non\_recursive(n);**

**cout<<"FACTORIAL is:"<<f;**

**getch();**

**return 0;**

**}**

**int non\_recursive(int n)**

**{**

**int a=1;**

**while(n>0)**

**{**

**a=a\*n;**

**n--;**

**}**

**return a;**

**}**

**//OUTPUT**

**ENTER N VALUE**

**4**

**FACTORIAL is 24**

**/\* 4.b.i. C++ Program to find recursive gcd of a given integer\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int gcd(int,int);**

**int main()**

**{**

**int a,b,g1;**

**clrscr();**

**cout<<"ENTER TWO INTEGERS :";**

**cin>>a>>b;**

**g1=gcd(a,b);**

**cout<<"GCD is:"<<g1;**

**getch();**

**return 0;**

**}**

**int gcd(int a,int b)**

**{**

**int g;**

**g=a%b;**

**if(g>0)**

**return (gcd(b,g));**

**else**

**return b;**

**}**

**//OUTPUT**

**ENTER TWO INTEGERS**

**7 3**

**GCD is 1**

**/\* 4.b.ii. C++ Program to find non-recursive gcd of a given integer\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int gcd(int,int);**

**int main()**

**{**

**int a,b,g;**

**clrscr();**

**cout<<"ENTER TWO INTEGERS:";**

**cin>>a>>b;**

**g=gcd(a,b);**

**cout<<"GCD is:"<<g;**

**getch();**

**return 0;**

**}**

**int gcd(int a,int b)**

**{**

**int r;**

**while(r!=0)**

**{**

**r=a%b;**

**a=b;**

**b=r;**

**}**

**return a;**

**}**

**//OUTPUT**

**ENTER TWO INTEGERS**

**10 5**

**GCD is 5**

**/\* 4.c.i. C++ Program to find nth fibonacci number usind recursive function\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int non\_recursive(int);**

**int main()**

**{**

**int r,n;**

**clrscr();**

**cout<<"Enter n value:";**

**cin>>n;**

**r=non\_recursive(n);**

**cout<<n<<"th fibonacci number is"<<r;**

**getch();**

**return 0;**

**}**

**int non\_recursive(int n)**

**{**

**int a=0,b=1,c;**

**for(int i=0;i<n-2;i++)**

**{**

**c=a+b;**

**a=b;**

**b=c;**

**}**

**return c;**

**}**

**//OUTPUT**

**Enter n value:**

**5**

**5th fibonacci number is 3**

**/\* 4.c.ii. C++ Program to find nth fibonacci number usind recursive function\*/**

**//PROGRAM**

**#include<conio.h>**

**#include<iostream.h>**

**int recursive(int);**

**void main()**

**{**

**int n,r;**

**clrscr();**

**cout<<"enter n";**

**cin>>n;**

**r=recursive(n-1);**

**cout<<n<<"th feb no."<<r;**

**getch();**

**}**

**int recursive(int n)**

**{**

**if(n==0)**

**{**

**return 0;**

**}**

**else if(n==1)**

**{**

**return 1;**

**}**

**else**

**{**

**return (recursive(n-1) + recursive(n-2));**

**}**

**}**

**//OUTPUT**

**Enter n :**

**5**

**5th fibonacci number is 3**

**/\* 5.C++ Program that uses recursive functions to solve Towers of Hanoi problem \*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**void tower(int,char,char,char);**

**int main()**

**{**

**int n;**

**clrscr();**

**cout<<"ENTER NUMBER OF DISCS";**

**cin>>n;**

**cout<<"TOWER OF HANOI INVOLVES FOLLOWING MOVES";**

**tower(n,'A','C','B');**

**getch();**

**return 0;**

**}**

**void tower(int n,char frompeg,char topeg,char auxpeg)**

**{**

**if(n==1)**

**{**

**cout<<endl<<"MOVE DISK 1 FROM "<<frompeg<<" TO "<<topeg;**

**}**

**else**

**{**

**tower(n-1,frompeg,auxpeg,topeg);**

**cout<<"MOVE DISK "<<n<<" FROM "<<frompeg<<" TO "<<topeg<<endl;**

**tower(n-1,auxpeg,topeg,frompeg);**

**}**

**}**

**//OUTPUT**

**ENTER NUMBER OF DISCS:3**

**TOWER OF HANOI INVOLVES FOLLOWING MOVES**

**MOVE DISK 1 FROM A TO C**

**MOVE DISK 2 FROM A TO B**

**MOVE DISK 1 FROM C TO B**

**MOVE DISK 3 FROM A TO C**

**MOVE DISK 1 FROM B TO A**

**MOVE DISK 2 FROM B TO C**

**MOVE DISK 1 FROM A TO C**

**/\*6. C++ Program that uses overloaded functions to swap (i)two integers (ii)two characters (iii)two reals. \*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**void swap(int ,int);**

**void swap(char,char);**

**void swap(float,float);**

**int main()**

**{**

**clrscr();**

**int a,b;**

**cout<<"Enter two integers";**

**cin>>a>>b;**

**cout<<"Before swaping: "<<a<<" , "<<b<<endl;**

**swap(a,b);**

**char c,d;**

**cout<<"Enter two characters";**

**cin>>c>>d;**

**cout<<"Before swaping: "<<c<<" , "<<d<<endl;**

**swap(c,d);**

**float e,f;**

**cout<<"Enter two float";**

**cin>>e>>f;**

**cout<<"Before swaping: "<<e<<" , "<<f<<endl;**

**swap(e,f);**

**getch();**

**return 0;**

**}**

**void swap(int a,int b)**

**{**

**int t;**

**t=a;**

**a=b;**

**b=t;**

**cout<<"After swaping : "<<a<<" , "<<b<<endl;**

**}**

**void swap(float e,float f)**

**{**

**float t;**

**t=e;**

**e=f;**

**f=t;**

**cout<<"After swaping : "<<e<<" , "<<f<<endl;**

**}**

**void swap(char c,char d)**

**{**

**char t;**

**t=c;**

**c=d;**

**d=t;**

**cout<<"After swaping : "<<c<<" , "<<d<<endl;**

**}**

**//OUTPUT**

**Enter two integers**

**33 66**

**Before swaping 33 66**

**After swaping 66 33**

**Enter two characters**

**h u**

**Before swaping: h u**

**After swaping : u h**

**Enter two float**

**8.79 9.77**

**Before swaping:8.79 9.77**

**After swaping :9.77 8.79**

**/\*7. C++ Program to find largest and smallest numbers of a list of integers\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int main()**

**{**

**int n,i,min,max,a[50];**

**cout<<"Enter the number of integers in list";**

**cin>>n;**

**cout<<"\n Enter the values in list";**

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

**{**

**cin>>a[i];**

**}**

**max=min=a[0];**

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

**{**

**if(a[i]<min)**

**min=a[i];**

**if(a[i]>max)**

**max=a[i];**

**}**

**cout<<"\n Minimum value is : "<<min<<"\n Maximum value is : "<<max;**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter the number of integers in list:**

**5**

**Enter the values in list:**

**77**

**11**

**32**

**43**

**56**

**Minimum value is :11 Maximum value is :77**

**/\*8. C++ Program to sort a list of numbers in ascending order.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**void sort(int arr[],int );**

**int n,arr[50];**

**void main()**

**{**

**int i;**

**clrscr();**

**cout<<"Enter the number of elements:";**

**cin>>n;**

**cout<<"Enter the values one by one";**

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

**{**

**cin>>arr[i];**

**}**

**cout<<"\n Array before sorting:\n";**

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

**{**

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

**}**

**cout<<"\n Array after sorting:\n";**

**sort(arr,n);**

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

**{**

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

**}**

**getch();**

**}**

**void sort(int arr[],int n)**

**{**

**int i,j,temp;**

**for(i=0;i<n-1;i++)**

**{**

**for(j=i;j<n;j++)**

**{**

**if(arr[i]>arr[j])**

**{**

**temp=arr[i];**

**arr[i]=arr[j];**

**arr[j]=temp;**

**}**

**}**

**}**

**}**

**//OUTPUT**

**Enter the number of elements:**

**5**

**Enter the values one by one**

**6**

**33**

**85**

**74**

**12**

**Array before sorting:**

**6 33 85 74 12**

**Array after sorting:**

**6 12 33 74 85**

**/\*9. C++ Program to find largest and smallest and sort a list using templates.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**template<class T>**

**void maxmin(T a[],int n)**

**{**

**int i;**

**T temp;**

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

**for(int j=i+1;j<n;j++)**

**{**

**if(a[i]>a[j])**

**{**

**temp=a[i];**

**a[i]=a[j];**

**a[j]=temp;**

**}**

**}**

**cout<<"max="<<a[n-1]<<"\n"<<"min="<<a[0]<<"\n";**

**cout<<"sorted list is: \n";**

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

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

**}**

**int main()**

**{**

**int a[50],i,ch,n;**

**double d[50];**

**float f[50];**

**char c[50];**

**cout<<"Enter corresponding Index"<<endl;**

**cout<<"1.integer"<<endl;**

**cout<<"2.characters"<<endl;**

**cout<<" 3.float numbers"<<endl;**

**cout<<" 4.double numbers"<<endl;**

**cin>>ch;**

**cout<<"Enter the n value\n";**

**cin>>n;**

**switch(ch)**

**{**

**case 1: cout<<"Enter integers\n";**

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

**cin>>a[i];**

**maxmin(a,n);**

**break;**

**case 2:**

**cout<<"Enter characters\n";**

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

**cin>>c[i];**

**maxmin(c,n);**

**break;**

**case 3:**

**cout<<"Enter floatnumbers\n";**

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

**cin>>f[i];**

**maxmin(f,n);**

**break;**

**case 4:**

**cout<<"Enter doublenumbers\n";**

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

**cin>>d[i];**

**maxmin(d,n);**

**break;**

**default:**

**cout<<"Invalid choice entered...";**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter corresponding Index**

**1.integer**

**2.characters**

**3.float numbers**

**4.double numbers**

**2**

**Enter the n value**

**5**

**Enter characters**

**d j a l v**

**max=v**

**min=a**

**sorted list is:**

**a d j l v**

**/\*10.C++ Program to sort list of names in alphabetical order.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<stdio.h>**

**#include<string.h>**

**#include<stdlib.h>**

**char arr[40][20];**

**void sort(char arr[40][20],int );**

**int n;**

**int main()**

**{**

**int i;**

**clrscr();**

**cout<<"Enter the number of elements: \n";**

**cin>>n;**

**cout<<"Enter the values one by one \n";**

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

**{**

**gets(arr[i]);**

**}**

**cout<<"List in alphabetical order:\n";**

**sort(arr,n);**

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

**{**

**puts(arr[i]);**

**}**

**getch();**

**return 0;**

**}**

**void sort(char arr[40][20],int n)**

**{**

**int i,j;char temp[15];**

**for(i=0;i<=n-1;i++)**

**{**

**for(j=i;j<n;j++)**

**{**

**if(strcmp(arr[i],arr[j])>0)**

**{**

**strcpy(temp,arr[i]);**

**strcpy(arr[i],arr[j]);**

**strcpy(arr[j],temp);**

**}**

**}**

**}**

**}**

**//OUTPUT**

**Enter the number of elements:**

**5**

**Enter the values one by one**

**ram**

**uma**

**siva**

**radha**

**krishna**

**List in alphabetical order**

**krishna**

**radha**

**ram**

**siva**

**uma**

**/\* 11.C++ Program to implement the matrix ADT using a classs.The operations supported by this ADT are:**

**a)Reading a matrix**

**b)Printing a matrix**

**c)Addition of matrices**

**d)Subtraction of matrices**

**e)Multiplication of matrices\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<stdlib.h>**

**class Matrix**

**{**

**public:**

**int t[10][20],d[10][10],e[10][10],i,j,k,m,n,a[10][10],b[10][10],c[10][10];**

**void read();**

**void sub();**

**void add();**

**void mul();**

**void print(int t[10][10]);**

**};**

**class Mato:public Matrix**

**{**

**public:**

**void read()**

**{**

**cout<<"Enter number of rows,columns";**

**cin>>m>>n;**

**cout<<"Enter the elements of first matrix";**

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

**{**

**for(j=0;j<n;j++)**

**{**

**cin>>a[i][j];**

**}**

**}**

**cout<<"Enter the elements of second matrix";**

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

**{**

**for(j=0;j<n;j++)**

**{**

**cin>>b[i][j];**

**}**

**}**

**void print(int t[10][10])**

**{**

**cout<<"matrix"<<endl;**

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

**{**

**for(j=0;j<n;j++)**

**{**

**cout<<t[i][j]<<" ";**

**}**

**cout<<endl;**

**}**

**}**

**void add()**

**{**

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

**{**

**for(j=0;j<n;j++)**

**{**

**c[i][j]=a[i][j]+b[i][j];**

**}**

**}**

**}**

**void sub()**

**{**

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

**{**

**for(j=0;j<n;j++)**

**{**

**d[i][j]=a[i][j]-b[i][j];**

**}**

**}**

**}**

**void mul()**

**{**

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

**{**

**for(j=0;j<n;j++)**

**{**

**e[i][j]=0;**

**for(k=0;k<n;k++)**

**{**

**e[i][j]+=a[i][k]\*b[k][j];**

**}**

**}**

**}**

**}**

**};**

**int main()**

**{**

**clrscr();**

**Mato m1;**

**m1.read();**

**m1.print(m1.a);**

**m1.print(m1.b);**

**m1.add();**

**cout<<"sum ";**

**m1.print(m1.c);**

**m1.sub();**

**cout<<"difference ";**

**m1.print(m1.d);**

**m1.mul();**

**cout<<"product ";**

**m1.print(m1.e);**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter number of rows,columns**

**2**

**2**

**Enter the elements of first matrix**

**5**

**3**

**2**

**6**

**Enter the elements of second matrix**

**4**

**2**

**1**

**3**

**matrix**

**5 3**

**2 6**

**matrix**

**4 2**

**1 3**

**sum matrix**

**9 5**

**3 9**

**difference matrix**

**1 1**

**1 3**

**product matrix**

**12 19**

**14 22**

**/\* 12.C++ Program to implement the matrix ADT using a classs.using overloading operators.The operations supported by this ADT are:**

**a)Reading a matrix**

**b)Printing a matrix**

**c)Addition of matrices**

**d)Subtraction of matrices**

**e)Multiplication of matrices\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<stdlib.h>**

**int m1,n1,m2,n2;**

**class matrix**

**{**

**int a[10][10],m,n;**

**public:**

**matrix(int a,int b)**

**{**

**m=a;**

**n=b;**

**}**

**friend istream & operator >>(istream &get,matrix &m);**

**friend ostream & operator <<(ostream &put,matrix &m);**

**matrix operator +(matrix);**

**matrix operator -(matrix);**

**matrix operator \*(matrix);**

**};**

**istream & operator >>(istream &get,matrix &x)**

**{**

**for(int i=0;i<x.m;i++)**

**for(int j=0;j<x.n;j++)**

**get>>x.a[i][j];**

**return get;**

**}**

**ostream & operator <<(ostream &put,matrix &x)**

**{**

**for(int i=0;i<x.m;i++)**

**{**

**for(int j=0;j<x.n;j++)**

**put<<x.a[i][j]<<"\t";**

**put<<endl;**

**}**

**return put;**

**}**

**matrix matrix:: operator +(matrix x)**

**{**

**matrix c(m1,n1);**

**for(int i=0;i<m;i++)**

**for(int j=0;j<n;j++)**

**c.a[i][j]=a[i][j]+x.a[i][j];**

**return c;**

**}**

**matrix matrix:: operator -(matrix x)**

**{**

**matrix c(m1,n1);**

**c.m=m;**

**c.n=n;**

**for(int i=0;i<m;i++)**

**for(int j=0;j<n;j++)**

**c.a[i][j]=a[i][j]-x.a[i][j];**

**return c;**

**}**

**matrix matrix:: operator \*(matrix x)**

**{**

**matrix c(m1,n2);**

**for(int i=0;i<m1;i++)**

**for(int j=0;j<n2;j++)**

**{**

**c.a[i][j]=0;**

**for(int k=0;k<n1;k++)**

**c.a[i][j]+=(a[i][k]\*x.a[k][j]);**

**}**

**return c;**

**}**

**void main()**

**{**

**clrscr();**

**cout<<"\nEnter the order of first matrix :";**

**cin>>m1>>n1;**

**cout<<"\nEnter the order of second matrix :";**

**cin>>m2>>n2;**

**matrix a(m1,n1),b(m2,n2),c(m1,n1),m(m1,n1),n(m1,n2);**

**if(m1==n2)**

**{**

**cout<<"\nEnter the elements of first matrix :";**

**cin>>a;**

**cout<<"\nEnter the elements of second matix :";**

**cin>>b;**

**c=a+b;**

**cout<<"\nThe first matrix A :\n";**

**cout<<a;**

**cout<<"\nThe second matrix B :\n";**

**cout<<b;**

**cout<<"\nThe resultant matrix after addition is :\n";**

**cout<<c;**

**m=a-b;**

**cout<<"\nThe resultant matrix after subtraction is :\n";**

**cout<<m;**

**n=a\*b;**

**cout<<"\nThe resultant matrix after multiplication is :\n";**

**cout<<n;**

**}**

**else**

**cout<<"\nMatrix operation is not possible";**

**getch();**

**}**

**//OUTPUT**

**Enter the order of first matrix :2**

**2**

**Enter the order of second matrix :2**

**2**

**Enter the elements of first matrix :1**

**2**

**3**

**4**

**Enter the elements of second matix :1**

**1**

**1**

**1**

**The first matrix A :**

**1 2**

**3 4**

**The second matrix B :**

**1 1**

**1 1**

**The resultant matrix after addition is :**

**2 3**

**4 5**

**The resultant matrix after subtraction is :**

**0 1**

**2 3**

**The resultant matrix after multiplication is :**

**3 7**

**3 7**

**/\* 13.C++ Program to implement the complex number ADT using a class.The operations supported by this ADT are:**

**a.Reading a complex number.**

**b.Writing a complex number.**

**c.Addition of complex numbers.**

**d.Subtraction of complex numbers.**

**e.Multiplication of complex numbers.**

**f.Division of complex numbers.\*/**

**//PROGRAM:**

**#include<iostream.h>**

**#include<conio.h>**

**class complex;**

**class complex1**

**{**

**float real,imaginary;**

**public:**

**void read();**

**void print();**

**void add(complex,complex);**

**void sub(complex,complex);**

**void mul(complex,complex);**

**void div(complex,complex);**

**};**

**class complex:public complex1**

**{**

**public:**

**void read()**

**{**

**cout<<"Enter the real part"<<endl;**

**cin>>real;**

**cout<<"Enter imaginary part"<<endl;**

**cin>>imaginary;**

**}**

**void print()**

**{**

**if(imaginary>0)**

**cout<<real<<"+"<<imaginary<<"i"<<endl;**

**else if(imaginary==0)**

**cout<<real<<endl;**

**else**

**cout<<real<<imaginary<<"i"<<endl;**

**}**

**void add(complex c1, complex c2)**

**{**

**real=c1.real+c2.real;**

**imaginary=c1.imaginary+c2.imaginary;**

**print();**

**}**

**void sub(complex c1, complex c2)**

**{**

**real=c1.real-c2.real;**

**imaginary=c1.imaginary-c2.imaginary;**

**print();**

**}**

**void mul(complex c1,complex c2)**

**{**

**real=(c1.real\*c2.real)-(c1.imaginary\*c2.imaginary);**

**imaginary=(c1.real\*c2.imaginary)+(c1.imaginary\*c2.real);**

**print();**

**}**

**void div(complex c1,complex c2)**

**{**

**real=((c1.real\*c2.real)+(c1.imaginary\*c2.imaginary))/((c2.real\*c2.real)+(c2.imaginary\*c2.imaginary));**

**imaginary=((c1.imaginary\*c2.real)-(c1.real\*c2.imaginary))/((c2.real\*c2.real)+(c2.imaginary\*c2.imaginary));**

**print();**

**}**

**int main()**

**{**

**complex c1,c2,c3;**

**clrscr();**

**c1.read();**

**cout<<"First complex number";**

**c1.print();**

**c2.read();**

**cout<<endl<<"Second complex number";**

**c2.print();**

**cout<<endl<<"Sum is";**

**c3.add(c1,c2);**

**cout<<endl<<"Difference is";**

**c3.sub(c1,c2);**

**cout<<endl<<"Product is";**

**c3.mul(c1,c2);**

**cout<<endl<<"Quotient is";**

**c3.div(c1,c2);**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter the real part**

**2**

**Enter imaginary part**

**4**

**First complex number**

**2+4i**

**Enter the real part**

**1**

**Enter imaginary part**

**2**

**Second complex number**

**1+2i**

**Sum is**

**3+6i**

**Difference is**

**1+2i**

**Product is**

**-6+8i**

**Quotient is**

**2+0i**

**/\*14.C++ Program that overloads the + operator and relational operators to perform:**

**a.concatenation of two strings.**

**b.comparision of two strings.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<string.h>**

**#include<stdio.h>**

**class string**

**{**

**char str[100];**

**public:**

**void input();**

**void output();**

**string operator+(string s);**

**void operator==(string s);**

**};**

**void string::input()**

**{**

**cout<<"Enter the string\n";**

**gets(str);**

**}**

**string string::operator+(string s)**

**{**

**string temp;**

**strcpy(temp.str,str);**

**strcat(temp.str,s.str);**

**return(temp);**

**}**

**void string::operator==(string st)**

**{**

**if(strcmp(str,st.str)==0)**

**cout<<"\nStrings are Equal";**

**else**

**cout<<"\nStrings are not Equal";**

**}**

**void string::output()**

**{**

**cout<<"The string is\n";**

**cout<<str;**

**}**

**void main()**

**{**

**string s1,s2,s3;**

**clrscr();**

**s1.input();**

**s2.input();**

**s3=s1+s2;**

**s3.output();**

**s1==s2;**

**getch();**

**}**

**//OUTPUT**

**Enter the string:**

**binary**

**Enter the string:**

**tree**

**The string is:**

**binarytree**

**Strings are not Equal**

**/\* 15.Implement the complex number ADT in c++ using a class. Using overloaded operators.The operations supported by this ADT are:**

**a.Reading a complex number.**

**b.Writing a complex number.**

**c.Addition of complex numbers.**

**d.Subtraction of complex numbers.**

**e.Multiplication of complex numbers.**

**f.Division of complex numbers.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<math.h>**

**#include<stdlib.h>**

**class complex**

**{**

**double r,i;**

**static int count;**

**public:**

**complex operator+(complex n2);**

**complex operator-(complex n2);**

**complex operator\*(complex n2);**

**complex operator/(complex n2);**

**friend istream& operator>>(istream& op, complex &x);**

**friend ostream& operator<<(ostream& op, complex &x);**

**};**

**int complex::count;**

**istream& operator>>(istream& ip, complex &x)**

**{**

**cout << "Type the real and imaginary parts of complex number"<<x.count+1<<" : ";**

**ip>>x.r;**

**ip>>x.i;**

**x.count++;**

**return ip;**

**}**

**ostream& operator<<(ostream& op, complex &x)**

**{**

**if(x.i>=0)**

**op << "The complex number is : "<<x.r<<"+"<<x.i<<"i"<<endl;**

**else**

**op << "The complex number is : "<<x.r<<x.i<<"i"<<endl;**

**return op;**

**}**

**complex complex::operator+(complex n2)**

**{**

**complex n3;**

**n3.r= r + n2.r;**

**n3.i= i + n2.i;**

**return n3;**

**}**

**complex complex::operator-(complex n2)**

**{**

**complex n3;**

**n3.r= r - n2.r;**

**n3.i= i - n2.i;**

**return n3;**

**}**

**complex complex::operator\*(complex n2)**

**{**

**complex n3;**

**n3.r= (r \* n2.r) - (i \* n2.i);**

**n3.i= (r \* n2.i) + (i \* n2.r);**

**return n3;**

**}**

**complex complex::operator/(complex n2)**

**{**

**complex n3;**

**n3.r= ((r \* n2.r) + (i \* n2.i))/(n2.r\*n2.r + n2.i\*n2.i);**

**n3.i= ((i \* n2.r) - (r \* n2.i))/(n2.r\*n2.r + n2.i\*n2.i);**

**return n3;**

**}**

**void main()**

**{**

**complex a,b,c;**

**clrscr();**

**cin>>a;**

**cin>>b;**

**cout<<a;**

**cout<<b;**

**cout<<"Sum is ";**

**c=a+b;**

**cout<<c;**

**cout<<"Difference is ";**

**c=a-b;**

**cout<<c;**

**cout<<"Product is ";**

**c=a\*b;**

**cout<<c;**

**cout<<"Quotient is ";**

**c=a/b;**

**cout<<c;**

**getch();**

**}**

**//OUTPUT**

**Enter the real and imaginary parts of a complex number 2**

**4**

**Enter the real and imaginary parts of a complex number 1**

**2**

**The complex number is : 2+4i**

**The complex number is : 1+2i**

**Sum is The complex number is : 3+6i**

**Difference is The complex number is : 1+2i**

**Product is The complex number is :-6+8i**

**Quotient is The complex number is : 2+0i**

**/\*16.Template based C++ program that determines if a**

**particular value occurs in an array of values.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**template <class T>**

**void search(T arr[])**

**{**

**T a;**

**int flag=0;**

**cout<<"Enter the element which you want to search:";**

**cin>>a;**

**for(T i=0;i<10;i++)**

**if(a==arr[i])**

**{**

**flag=1;**

**break;**

**}**

**if(flag==1)**

**cout<<"\nElement is present in the array.";**

**else**

**cout<<"\nElement is not present in the array.";**

**}**

**void main()**

**{**

**clrscr();**

**int arri[50],ch,i,n;**

**char arrc[51];**

**float arrf[50];**

**cout<<"Program to search element in an array.\n Which array you want to make?\n";**

**cout<<"Enter your choice:\n1.integer\n2.characters\n3.Real number\n";**

**cin>>ch;**

**cout<<"Enter number of elements in the array \n";**

**cin>>n;**

**switch(ch)**

**{**

**case 1: cout<<"Enter array of integers:";**

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

**cin>>arri[i];**

**search(arri);**

**break;**

**case 2: cout<<"Enter array of characters:";**

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

**cin>>arrc[i];**

**search(arrc);**

**break;**

**case 3: cout<<"Enter array of real numbers:";**

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

**cin>>arrf[i];**

**search(arrf);**

**break;**

**default:cout<<"Wrong choice";**

**}**

**getch();**

**}**

**//OUTPUT**

**Program to search element in an array.**

**Which array you want to make?**

**Enter your choice:**

**1.integer**

**2.characters**

**3.Real number**

**3**

**Enter number of elements in the array**

**3**

**Enter array of real numbers:**

**3.4**

**22**

**79**

**Enter the element you want to search:22**

**Element is present in array**

**/\*17.C++ Program that uses functions to**

**(a)insert sub-string in main string**

**(b)delete n characters from a given position in string\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<stdio.h>**

**#include<conio.h>**

**#include<string.h>**

**#include<stdlib.h>**

**void insert();**

**void deletes();**

**int main()**

**{**

**int ch;**

**clrscr();**

**cout<<"Enter choice \n 1.Insert substring in main string \n 2.Delete substring from main string \n";**

**cin>>ch;**

**if(ch==1)**

**insert();**

**if(ch==2)**

**deletes();**

**else**

**{**

**cout<<"invalid choice";**

**exit(0);**

**}**

**return 0;**

**}**

**void insert()**

**{**

**char st[15],sub[15],temp[15];**

**int i,j,k,pos;**

**cout<<"Enter Main String :";**

**cin>>st;**

**cout<<"Enter Sub String : ";**

**cin>>sub;**

**cout<<"Enter the position to insert the substring :";**

**cin>>pos;**

**pos=pos-1;**

**strcpy(temp,st);**

**for(i=pos,j=0;j<=strlen(sub);i++,j++)**

**st[i]=sub[j];**

**for(j=pos,i=strlen(st);j<=strlen(temp);i++,j++)**

**st[i]=temp[j];**

**st[i]='\0';**

**cout<<"string is "<<st;**

**getch();**

**}**

**void deletes()**

**{**

**char st[15];**

**int i,j,l,pos,n;**

**cout<<"Enter Any String : ";**

**cin>>st;**

**cout<<"Enter the position to delete the characters :";**

**cin>>pos;**

**pos=pos-1;**

**cout<<"Enter the no of characters to be deleted:";**

**cin>>n;**

**l=strlen(st);**

**if(pos>l)**

**cout<<" Deletion is not possible";**

**else**

**{**

**for(i=pos,j=pos+n;i<=l;i++,j++)**

**st[i]=st[j];**

**st[i]='\0';**

**}**

**cout<<"string is "<<st;**

**getch();**

**}**

**//OUTPUT**

**Enter choice**

**1.Insert substring in main string**

**2.Delete substring from main string**

**1**

**Enter Main String :**

**arctic**

**Enter Sub String :**

**nta**

**Enter the position to insert the substring**

**2**

**string is :Antarctic**

**/\*18.C++ Program that uses a function to reverse the character string in place,without any duplication of characters.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<stdio.h>**

**#include<string.h>**

**void duplicate()**

**{**

**char str[30];**

**int i,j,k,l;**

**clrscr();**

**cout<<"Enter a string ";**

**gets(str);**

**l=strlen(str);**

**str[l]='\0';**

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

**{**

**for(j=i+1;j<l;j++)**

**{**

**if(str[i]==str[j])**

**{**

**k=j;**

**for(k;k<l;k++)**

**{**

**str[k]=str[k+1];**

**}**

**str[k]='\0';**

**}**

**}**

**}**

**cout<<"String is ";**

**strrev(str);**

**puts(str);**

**}**

**int main()**

**{**

**clrscr();**

**duplicate();**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter a string :element**

**String is :tneml**

**/\*19.C++ Program to make the frequency count of letters in a given text.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<string.h>**

**#include<conio.h>**

**#include<stdio.h>**

**int main()**

**{**

**char string[100], ch;**

**int c = 0, count[26] = {0};**

**clrscr();**

**cout<<" Enter a string: \n";**

**gets(string);**

**while ( string[c] != '\0' )**

**{**

**if ( string[c] >= 'a' && string[c] <= 'z' )**

**count[string[c]-'a']++;**

**c++;**

**}**

**for ( c = 0 ; c < 26 ; c++ )**

**{**

**if( count[c] != 0 )**

**cout<<char(c+'a')<<" occurs "<<count[c]<<" times in the entered string.\n";**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter a string: element**

**e occurs 3 times in the entered string.**

**L occurs 1 times in the entered string.**

**m occurs 1 times in the entered string.**

**n occurs 1 times in the entered string.**

**t occurs 1 times in the entered string.**

**/\*20.C++ Program to count the lines , words and characters in a given text\*/**

**//PROGRAM**

**#include <iostream.h>**

**#include <ctype.h>**

**#include <conio.h>**

**void main()**

**{**

**char name[10][10],c;**

**int lines=1;**

**int words=1;**

**int chars=1;**

**clrscr();**

**cout<<"Enter string termanate by # : ";**

**cin.get(c);**

**while(c != '#')**

**{**

**cin.get(c);**

**if(isalpha(c))**

**chars++;**

**if(c==' ' || c=='\n')**

**words++;**

**if(c=='\n')**

**lines++;**

**}**

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

**cout<<"Lines: "<< lines<<endl;**

**cout<<"Words: "<< words<<endl;**

**cout<<"Characters: "<< chars;**

**getch();**

**getch();**

**}**

**//OUTPUT**

**Enter string termanate by # :**

**welcome to hyderabad**

**you are in secunderabad railway station**

**Lines:2**

**Words:9**

**Characters:52**

**/\*21.C++ Program to determine if given string is a palindrome or not.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<string.h>**

**int main()**

**{**

**char str1[20],str2[20];**

**int i;**

**clrscr();**

**cout<<"Enter string ";**

**cin>>str1;**

**strcpy(str2,str1);**

**strrev(str1);**

**i=strcmp(str1,str2);**

**if(i==0)**

**cout<<"Palindrome sting";**

**else**

**cout<<"Not palindrome";**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter string: malayalam**

**Palindrome string.**

**/\*22.C++ Program to make the frequency count of words in a given text.\*/**

**//PROGRAM**

**#include<string.h>**

**#include<iostream.h>**

**#include<conio.h>**

**#include<stdio.h>**

**#define SIZE 50**

**struct freq**

**{**

**int n;**

**char \*s;**

**};**

**void display(char text[][SIZE],int num)**

**{**

**for(int i=0;i<=num;++i)**

**{**

**puts(text[i]);**

**}**

**}**

**int main()**

**{**

**char text[SIZE][SIZE],ch;**

**int i=0,j=0,k=-1,a;**

**int flg=1;**

**clrscr();**

**freq words[120];**

**cout<<"Please enter text trucated by $"<<endl;**

**while((ch=getchar())!='$')**

**{**

**if(ch==' '&&flg==1)**

**continue;**

**else if(ch==' '||ch=='\n')**

**{**

**flg=1;**

**++i;**

**j=0;**

**}**

**else**

**{**

**flg=0;**

**text[i][j++]=ch;**

**text[i][j]='\0';**

**}**

**}**

**cout<<"Given text is :"<<endl;**

**display(text,i);**

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

**{**

**if(strcmp(text[j],"#")!=0)**

**{**

**words[++k].s=text[j];**

**words[k].n=1;**

**for(a=j+1;a<=i;++a)**

**{**

**if(strcmp(text[j],text[a])==0)**

**{**

**++words[k].n;**

**strcpy(text[a],"#");**

**}**

**}**

**}**

**else**

**continue;**

**}**

**for(j=0;j<=k;++j)**

**{**

**cout<<words[j].s<<"\t is repeated "<<words[j].n<< "times"<<endl;**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Please enter text truncated by $**

**lion is a wild animal lion eats other wild animal$**

**Given text is:**

**lion**

**is**

**a**

**wild**

**animal**

**lion**

**eats**

**other**

**wild**

**animal**

**lion is repeated 2 times**

**is is repeated 1 times**

**a is repeated 1 times**

**wild is repeated 2 times**

**animal is repeated 2 times**

**eats is repeated 1 times**

**other is repeated 1 times**

**/\*23.C++ Program that displays the position or index in the string S where the string t begins, or -1 if S doesn't contain t.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<string.h>**

**int main()**

**{**

**char s[30],t[20];**

**char \*pos;**

**clrscr();**

**cout<<"Enter the main string";**

**cin>>s;**

**cout<<"Enter the string to be searched";**

**cin>>t;**

**pos=strstr(s,t);**

**if(pos)**

**cout<<"Second string found in the main String at"<<pos-s+1<<" index";**

**else**

**cout<<"-1";**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter the main string:**

**elephant**

**Enter the string to be searched:**

**ant**

**Second string found in the main String at 6 index**

**/\*24.C++ Program to find 2's complement of a given binary number\*/**

**//PROGRAM**

**#include<stdio.h>**

**#include<conio.h>**

**#include<iostream.h>**

**#include<string.h>**

**void main()**

**{**

**char str[10],i,j;**

**clrscr();**

**cout<<"Enter Binary Number:";**

**gets(str);**

**for(i=strlen(str)-1;i>=0;i--)**

**if(str[i]=='1')**

**break;**

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

**if(str[j]=='1')**

**str[j]='0';**

**else**

**str[j]='1';**

**cout<<"2's Complement is:"<<str;**

**getch();**

**}**

**//OUTPUT**

**Enter Binary Number: 1100**

**2's Complement is : 0100**

**/\*25.C++ Program to count number of 1 bits in a given integer\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**void main()**

**{**

**int num,count=0;**

**clrscr();**

**cout<<"Enter a number";**

**cin>>num;**

**while(num>0)**

**{**

**num&=(num-1);**

**count++;**

**}**

**cout<<"The number of 1 bits is : "<<count;**

**getch();**

**}**

**//OUTPUT**

**Enter a number : 10**

**The number of 1 bits is : 2**

**/\*26.C++ Program to generate pascal's triangle.\*/**

**//PROGRAM**

**#include <iostream.h>**

**#include <conio.h>**

**int main()**

**{**

**int a[20][20],i,j,n,k;**

**clrscr();**

**cout<<" Enter number of rows ";**

**cin>>n;**

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

**{**

**j=1;**

**a[i][0] = 1;**

**a[i][i] = 1;**

**while(j<i)**

**{**

**a[i][j] = a[i-1][j-1] + a[i-1][j];**

**j++;**

**}**

**}**

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

**{**

**j=n;**

**while(j>i)**

**{**

**cout<<" ";**

**j--;**

**}**

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

**{**

**cout<<a[i][k]<<" ";**

**}**

**cout<<endl;**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter number of rows:5**

**1**

**1 1**

**1 2 1**

**1 3 3 1**

**1 4 6 4 1**

**/\*27.C++ Program to construct pyramid of numbers.\*/**

**//PROGRAM**

**#include <iostream.h>**

**#include <conio.h>**

**int main()**

**{**

**int i,j,n,k;**

**clrscr();**

**cout<<"Enter number of rows ";**

**cin>>n;**

**for( i = 1; i <= n ; i++)**

**{**

**for( k = i ; k <=n ; k++)**

**{**

**cout<<" ";**

**}**

**for( j = 1 ; j <=i ; j++)**

**{**

**cout<<i<<" ";**

**}**

**cout<<endl;**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter number of rows 5**

**1**

**2 2**

**3 3 3**

**4 4 4 4**

**5 5 5 5 5**

**/\*28.C++ Program to compute the sine series.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#define PI 3.1415926**

**void sin(float f,int n);**

**long fact(int n)**

**{**

**long f=1;**

**for(int i=1;i<=n;++i)**

**f =f\*i;**

**return f;**

**}**

**double power(float b,int p)**

**{**

**double k=1;**

**for(int i=1;i<=p;++i)**

**k=k\*b;**

**return k;**

**}**

**long power(int b,int p)**

**{**

**long k=1;**

**for(int i=1;i<=p;++i)**

**k=k\*b;**

**return k;**

**}**

**int main()**

**{**

**float x;**

**int m ;**

**clrscr();**

**cout<<"\n Enter x value ";**

**cin>>x;**

**cout<<"\n Enter n value ";**

**cin>>m;**

**while(m<0)**

**{**

**cout<<"\n Enter n value ";**

**cin>>m;**

**}**

**x=x\*(PI/180);**

**sin(x,m);**

**getch();**

**return 0;**

**}**

**void sin(float x,int num)**

**{**

**int n,m;**

**long a,c;**

**double b,Result=0;**

**for(n=0;n<=num;++n)**

**{**

**a= power(-1,n) ; //a =(-1)^n**

**m=(2\*n)+1;**

**b=power(x,m); //b=x^((2\*n)+1)**

**c=fact(m); //c=((2\*n)+1)!**

**Result=Result+((a\*b)/c);**

**}**

**cout<<"sine value for "<<x<<" is : "<<Result;**

**}**

**//OUTPUT**

**Enter x value 90**

**Enter n value 2**

**sine value for 1.570796 is : 1.004525**

**/\*29.C++ Program roman numeral into arabic integer\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<stdlib.h>**

**#include<conio.h>**

**void main()**

**{**

**char s[10];**

**int sum=0,k=0;**

**clrscr();**

**cout<<"Enter roman numeral :";**

**cin>>s;**

**while(s[k]!='\0')**

**{**

**switch(s[k])**

**{**

**case 'm':sum+=1000;**

**k++;**

**break;**

**case 'd':**

**sum+=500;**

**k++;**

**break;**

**case 'c':if(s[k+1]=='d')**

**{**

**sum+=400;**

**k+=2;**

**break;**

**}**

**if(s[k+1]=='m')**

**{**

**sum+=900;**

**k+=2;**

**break;**

**}**

**else**

**{**

**sum+=100;**

**k++;**

**break;**

**}**

**case 'l':**

**sum+=50;**

**k++;**

**break;**

**case 'x': if(s[k+1]=='l')**

**{**

**sum+=40;**

**k+=2;**

**break;**

**}**

**if(s[k+1]=='c')**

**{**

**sum+=90;**

**k+=2;**

**break;**

**}**

**else**

**{**

**sum+=10;**

**k++;**

**break;**

**}**

**case 'v':**

**sum+=5;**

**k++;**

**break;**

**case 'i':if(s[k+1]=='x' )**

**{**

**sum+=9;**

**k+=2;**

**break;**

**}**

**if(s[k+1]=='v')**

**{**

**sum+=4;**

**k+=2;**

**break;**

**}**

**else**

**{**

**sum+=1;**

**k++;**

**break;**

**}**

**default :cout<<"Invalid Roman numeral:";**

**exit(0);**

**}**

**}**

**cout<<"The equivalent Arabic integer is :"<<sum;**

**getch();**

**}**

**//OUTPUT**

**Enter roman numeral :mcmxcvi**

**The equivalent Arabic integer is :1996**

**/\*30.C++ Program which converts a positive arabic number into corresponding roman numeral.\*/**

**//PROGRAM**

**#include <iostream.h>**

**#include <string.h>**

**#include <math.h>**

**#include<conio.h>**

**int main()**

**{**

**char ans, ch;**

**clrscr();**

**do**

**{**

**int yr, m, cm, d, c, xc, l, x, ix, v, i;**

**cout<< " Enter any ARABIC NUMBER you want: ";**

**cin>> yr;**

**cout<<"\n In Roman Numeral Format: ";**

**for(m = 1000;yr >= m; yr -= 1000)**

**{**

**cout<<"M";**

**}**

**for(cm = 900; yr >= cm; yr -= 900)**

**{**

**cout<<"CM";**

**}**

**for(d = 500;yr >= d; yr -= 500)**

**{**

**cout<<"D";**

**}**

**if(yr >= 400)**

**{**

**cout<<"CD"; yr -= 400;**

**}**

**else**

**{**

**for(c = 100;yr >= c; yr -= 100)**

**{**

**cout<<"C";**

**}**

**}**

**for(xc = 90; yr >= xc; yr -= 90)**

**{**

**cout<< "XC";**

**}**

**for(l = 50;yr >= l; yr -= 50)**

**{**

**cout<<"L";**

**}**

**if(yr >= 40)**

**{**

**cout<<"XL";**

**yr -= 40;**

**}**

**else**

**{**

**for(x = 10;yr >= x; yr -= 10)**

**{**

**cout<<"X";**

**}**

**}**

**for(ix = 9; yr >= ix; yr -= 9)**

**{**

**cout<< "IX";**

**}**

**for(v = 5;yr >= v; yr -= 5)**

**{**

**cout<<"V";**

**}**

**if(yr >= 4)**

**{**

**cout<<"IV"; yr -= 4;**

**}**

**else**

**{**

**for(i = 1;yr >= i; yr -= 1)**

**{**

**cout<<"I";**

**}**

**}**

**cout<< "\n\n Try Again?[y/n]: ";**

**cin>>ans;**

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

**return 0;**

**}**

**//OUTPUT**

**Enter any ARABIC NUMBER you want: 45**

**In Roman Numeral Format: XLV**

**Try Again?[y/n]: n**

**/\*31.C++ Program to display contents of a text file\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<stdio.h>**

**#include<conio.h>**

**#include<fstream.h>**

**void main()**

**{**

**char c,fname[10];**

**clrscr();**

**ofstream out;**

**cout<<"Enter File name:";**

**cin>>fname;**

**out.open(fname);**

**cout<<"Enter contents to store in file (Enter # at end):\n";**

**while((c=getchar())!='#')**

**{**

**out<<c;**

**}**

**out.close();**

**ifstream in(fname);**

**cout<<"\n\n Printing contents of the file "<<fname<<endl;**

**while(in.eof()==0)**

**{**

**in.get(c);**

**cout<<c;**

**}**

**getch();**

**}**

**//OUTPUT**

**Enter File name : library**

**Enter contents to store in file (Enter # at end):**

**Library has many books. books are kept in order.#**

**Printing contents of the file : library**

**Library has many books. books are kept in order.**

**/\*32.C++ Program which copies one file to another.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<fstream.h>**

**#include<conio.h>**

**#include<stdio.h>**

**int main()**

**{**

**char c,fname[10],name[10];**

**clrscr();**

**ofstream out;**

**cout<<"Enter File name: ";**

**cin>>fname;**

**out.open(fname);**

**cout<<"Enter contents to store in file (Enter # at end):\n";**

**while((c=getchar())!='#')**

**{**

**out<<c;**

**}**

**out.close();**

**ifstream in(fname);**

**ofstream copy;**

**cout<<"\n\n Enter the new filename \n";**

**cin>>name;**

**cout<<" \n Copying contents of the file "<<fname<<"to "<<name<<endl;**

**copy.open(name)**

**while(in.eof()==0)**

**{**

**in.get(c);**

**copy<<c;**

**}**

**in.close();**

**copy.close();**

**ifstream new(fame);**

**cout<<"\n\n Printing contents of the new file "<<name<<endl;**

**while(new.eof()==0)**

**{**

**new.get(c);**

**cout<<c;**

**}**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter File name: books**

**Enter contents to store in file (Enter # at end):**

**There are many types of books like encyclopedias, journals ,text books etc. #**

**Enter the new filename :shelf**

**Copying contents of the file books to shelf**

**Printing contents of the new file :shelf**

**There are many types of books like encyclopedias, journals ,text books etc.**

**/\*33.C++ Program to count the lines , words and characters in a given text file\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<stdio.h>**

**#include<ctype.h>**

**#include<conio.h>**

**#include<fstream.h>**

**void main()**

**{**

**char c,fname[10];**

**int lines=1,words=1,chars=1;**

**clrscr();**

**ofstream out;**

**cout<<"Enter File name:";**

**cin>>fname;**

**out.open(fname);**

**cout<<"Enter contents to store in file (Enter # at end):\n";**

**while((c=getchar())!='#')**

**{**

**out<<c;**

**}**

**out.close();**

**ifstream in(fname);**

**cout<<"\n\n Printing contents of the file "<<fname<<endl;**

**while(in.eof()==0)**

**{**

**in.get(c);**

**if(isalpha(c))**

**chars++;**

**if(c==' ' || c=='\n')**

**words++;**

**if(c=='\n')**

**lines++;**

**}**

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

**cout<<"Lines: "<< lines<<endl;**

**cout<<"Words: "<< words<<endl;**

**cout<<"Characters: "<< characters;**

**in.close();**

**getch();**

**}**

**//OUTPUT**

**Enter File name :station**

**Enter contents to store in file (Enter # at end):**

**Welcome to hyderabad**

**you are in secunderabad railway station#**

**Lines:2**

**Words:9**

**Characters:52**

**/\*34.C++ Program to change a specific character in a file\*/**

**//PROGRAM**

**#include<iostream.h> //file name, character number given by user.**

**#include<conio.h>**

**#include<fstream.h>**

**void main()**

**{**

**char c,fname[10];**

**clrscr();**

**ofstream out;**

**cout<<"Enter File name: ";**

**cin>>fname;**

**out.open(fname);**

**cout<<"Enter contents to store in file (Enter # at end):\n";**

**while((c=getchar())!='#')**

**{**

**out<<c;**

**}**

**out.close();**

**ifstream in(fname);**

**cout<<"\n\nPrinting contents of the file "<<fname<<endl;**

**while(in.eof()==0)**

**{**

**in.get(c);**

**cout<<c;**

**}**

**in.close();**

**cout<<"Enter the character number to be changed (with spaces)\n";**

**int n;**

**cin>>n;n--;**

**fstream file;**

**file.open(fname,ios::in|ios::out);**

**file.seekp(n,ios::beg);**

**cout<<"Enter new value \n";**

**cin>>c;**

**file<<c;**

**file.close();**

**ifstream init(fname);**

**cout<<"\n\nPrinting contents of modifed file "<<fname<<endl;**

**while(init.eof()==0)**

**{**

**init.get(c);**

**cout<<c;**

**}**

**init.close();**

**getch();**

**}**

**//OUTPUT**

**Enter File name: college**

**Enter contents to store in file (Enter # at end):**

**I study im jntuhceh#**

**Printing contents of the file :college**

**I study im jntuhceh**

**Enter the character number to be changed count even spaces as chracters:10**

**Enter new value:n**

**Printing contents of modifed file :college**

**I study in jntuhceh**

**/\*35.C++ Program to reverse first n characters in a fli.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<stdio.h>**

**#include<conio.h>**

**#include<string.h>**

**#include<fstream.h>**

**void main()**

**{**

**char c[100],fname[10],g[100];**

**clrscr();**

**ofstream out;**

**cout<<"Enter File name:";**

**cin>>fname;**

**out.open(fname);**

**cout<<"Enter contents to store in file (press enter to end):\n";**

**gets(c);**

**out<<c;**

**out.close();**

**int n;**

**ifstream in(fname);**

**cout<<"Enter n";**

**cin>>n;**

**cout<<"\n\nPrinting contents of the file "<<fname<<" in reverse "<<endl;**

**in.seekg(0,ios::beg);**

**for(int i=n,j=0;i>0;i--,j++)**

**{**

**in.get(c[i]);**

**g[j]=c[i];**

**}**

**g[j++]='\0';**

**strrev(g);**

**cout<<g;**

**getch();**

**}**

**//OUTPUT**

**Enter File name: google**

**Enter contents to store in file (press enter to end):**

**Google is a search engine**

**Enter n: 9**

**Printing contents of the file google in reverse**

**si elgoog**

**/\*36.C++Program that uses a function to delete all duplicate characters in a given string.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<stdio.h>**

**#include<string.h>**

**void duplicate()**

**{**

**char str[30];**

**int i,j,k,l;**

**clrscr();**

**cout<<"Enter a string ";**

**gets(str);**

**l=strlen(str);**

**str[l]='\0';**

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

**{**

**for(j=i+1;j<l;j++)**

**{**

**if(str[i]==str[j])**

**{**

**k=j;**

**for(k;k<l;k++)**

**{**

**str[k]=str[k+1];**

**}**

**str[k]='\0';**

**}**

**}**

**}**

**cout<<"String is ";**

**puts(str);**

**}**

**int main()**

**{**

**clrscr();**

**duplicate();**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter a string :element**

**String is :elmnt**

**/\*37.C++ Program that uses a function i to a which converts a number to a character string.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<stdlib.h>**

**#include<conio.h>**

**int main()**

**{**

**int n;**

**char a[10];**

**clrscr();**

**cout<<"Enter a number \n";**

**cin>>n;**

**itoa(n,a,10);**

**cout<<"Num "<<n<<" Char "<<a;**

**getch();**

**return 0;**

**}**

**//OUTPUT**

**Enter a number 45**

**Num 45 Char 45**

**/\*38.C++ Program that uses recursive function to find the binary equivalent of a given non-negative integer.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**int\* recursive(int);**

**int a[50],\*c;**

**static int i=0;**

**void main()**

**{**

**int n;**

**clrscr();**

**cout<<"Enter a non negative number \n";**

**cin>>n;**

**c=recursive(n);**

**for(i;i!=0;i--)**

**{**

**cout<<c[i];**

**}**

**getch();**

**}**

**int\* recursive(int n)**

**{**

**i++;**

**a[i]=n%2;**

**n=n/2;**

**if(n==0)**

**return a;**

**else**

**recursive(n);**

**return a;**

**}**

**//OUTPUT**

**Enter a non negative number 20**

**10100**

**/\*39.C++ Program to generate prime numbers upto n using Sieve of Eratosthenes method.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#define SIZE 100**

**void main()**

**{**

**int i,j,n,a[SIZE];**

**cout<<"Please enter a number :"<<endl;**

**cin>>n;**

**for(i=2;i<=n;++i)**

**a[i-2]=i;**

**for(i=2;i<=n;++i)**

**{**

**if(a[i-2]==0)**

**continue;**

**cout<<a[i-2]<<" is prime"<<endl;**

**for(j=i+1;j<=n;++j)**

**{**

**if(j%i==0)**

**a[j-2]=0;**

**}**

**}**

**getch();**

**}**

**//OUTPUT**

**Please enter a number :19**

**2 is prime**

**3 is prime**

**5 is prime**

**7 is prime**

**11 is prime**

**13 is prime**

**17 is prime**

**/\*40.C++ Program to (a)write an object to a file (b)read an object from a file.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**#include<fstream.h>**

**class Inventory**

**{**

**char name[10];**

**int code,cost;**

**public: void readdata();**

**void writedata();**

**};**

**void Inventory::readdata()**

**{**

**cout<<"\nEnter name ";**

**cin>>name;**

**cout<<"\nEnter code ";**

**cin>>code;**

**cout<<"\nEnter cost ";**

**cin>>cost;**

**}**

**void Inventory::writedata()**

**{**

**cout<<"\n Name :"<<name<<"\n Code :"<<code<<"\n Cost :"<<cost;**

**}**

**void main()**

**{**

**Inventory item[2];**

**fstream object;**

**object.open("object.dat",ios::in|ios::out);**

**cout<<"Enter the details for two items \n";**

**for(int i=0;i<2;i++)**

**{**

**item[i].readdata();**

**object.write((char\*)&item[i],sizeof(item[i]));**

**}**

**object.seekg(0);**

**cout<<"\n output\n";**

**for(i=0;i<2;i++)**

**{**

**object.read((char\*)&item[i],sizeof(item[i]));**

**item[i].writedata();**

**}**

**object.close();**

**getch();**

**}**

**//OUTPUT**

**Enter the details for two items**

**Enter name :oopc**

**Enter code :4**

**Enter cost :450**

**Enter name: ads**

**Enter code :5**

**Enter cost :380**

**Name :oopc**

**Code :4**

**Cost :450**

**Name :ads**

**Code :5**

**Cost :380**

**/\*41.a.C++ Program that illustrate Single inheritance.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<iomanip.h>**

**#include<conio.h>**

**class Emp**

**{**

**public:**

**int eno;**

**char name[20],des[20];**

**void get()**

**{**

**cout<<"Enter the employee number:";**

**cin>>eno;**

**cout<<"Enter the employee name:";**

**cin>>name;**

**}**

**};**

**class Salary:public Emp**

**{**

**float bp,hra,np;**

**public:**

**void get1()**

**{**

**cout<<"Enter the basic pay:";**

**cin>>bp;**

**cout<<"Enter the Human Resource Allowance:";**

**cin>>hra;**

**}**

**};**

**void main()**

**{**

**int i,n;**

**char ch;**

**Salary s[10];**

**clrscr();**

**cout<<"Enter the number of employees: ";**

**cin>>n;**

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

**{**

**s[i].get();**

**s[i].get1();**

**}**

**getch();**

**}**

**//OUTPUT**

**Enter the number of employees : 2**

**Enter the employee number :1**

**Enter the employee name :sri**

**Enter the basic pay :40000**

**Enter the Human Resource Allowance :7000**

**Enter the employee number :2**

**Enter the employee name :sai**

**Enter the basic pay :42000**

**Enter the Human Resource Allowance :8000**

**/\*41.b.C++ Program that illustrate Multiple inheritance.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**class student**

**{**

**protected:**

**int rno,m1,m2;**

**public:**

**void get()**

**{**

**cout<<"Enter the Roll number :";**

**cin>>rno;**

**cout<<"Enter the two subject marks :";**

**cin>>m1>>m2;**

**}**

**};**

**class sports**

**{**

**protected:**

**int sm;**

**public:**

**void getsm()**

**{**

**cout<<"\nEnter the sports mark :";**

**cin>>sm;**

**}**

**};**

**class statement:public student,public sports**

**{**

**int tot,avg;**

**public:**

**void display()**

**{**

**tot=(m1+m2+sm);**

**avg=tot/3;**

**cout<<"\n\n\tRoll No : "<<rno<<"\n\tTotal : "<<tot;**

**cout<<"\n\tAverage : "<<avg;**

**}**

**};**

**void main()**

**{**

**clrscr();**

**statement obj;**

**obj.get();**

**obj.getsm();**

**obj.display();**

**getch();**

**}**

**//OUTPUT**

**Enter the Roll number :25**

**Enter the two subject marks :79 81**

**Enter the sports mark :95**

**Roll Number : 25**

**Total : 255**

**Average : 85**

**/\*41.c.C++ Program that illustrate Multi level inheritance.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**class top**

**{**

**public :**

**int a;**

**void getdata()**

**{**

**cout<<"\nEnter a Number :";**

**cin>>a;**

**}**

**};**

**class middle :public top**

**{**

**public:**

**int b;**

**void square()**

**{**

**getdata();**

**b=a\*a;**

**cout<<"\nSquare Is :"<<b;**

**}**

**};**

**class bottom :public middle**

**{**

**public:**

**int c;**

**void cube()**

**{**

**square();**

**c=b\*a;**

**cout<<"\nCube Is :"<<c;**

**}**

**};**

**void main()**

**{**

**clrscr();**

**bottom b1;**

**b1.cube();**

**getch();**

**}**

**//OUTPUT**

**Enter a Number**

**5**

**Square is :25**

**Cube is :125**

**/\*41.d.C++ Program that illustrate Hierarchical inheritance.\*/**

**//PROGRAM**

**#include <iostream.h>**

**#include<conio.h>**

**class Side**

**{**

**protected:**

**int l;**

**public:**

**void set\_values (int x)**

**{**

**l=x;**

**}**

**};**

**class Square: public Side**

**{**

**public:**

**int sq()**

**{**

**return (l \*l);**

**}**

**};**

**class Cube:public Side**

**{**

**public:**

**int cub()**

**{**

**return (l\*l\*l);**

**}**

**};**

**void main ()**

**{**

**int n;**

**clrscr();**

**cout<<"Enter the length \n";**

**cin>>n;**

**Square s;**

**s.set\_values (n);**

**cout << "The area of square is :" << s.sq() << endl;**

**Cube c;**

**c.set\_values (n);**

**cout << "The volume of cube is :" << c.cub() << endl;**

**getch();**

**}**

**OUTPUT**

**Enter the length 9**

**The area of square is :81**

**The volume of cube is :729**

**/\*42.C++ Program that illustrates the order of excecution of constructors and destructors when new class is derived from more than one base class.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**class base1**

**{**

**public:**

**base1()**

**{**

**cout<<"constructing base1"<<endl;**

**}**

**~base1()**

**{**

**cout<<"destructing base1"<<endl;**

**}**

**};**

**class base2**

**{**

**public:**

**base2()**

**{**

**cout<<"constructing base2 "<<endl;**

**}**

**~base2()**

**{**

**cout<<"destructing base2"<<endl;**

**}**

**};**

**class derived1:public base1,public base2**

**{**

**public:**

**derived1()**

**{**

**cout<<"constructing derived1"<<endl;**

**}**

**~derived1()**

**{**

**cout<<"destructing derived1"<<endl;**

**}**

**};**

**class derived2:public base2,public base1**

**{**

**public:**

**derived2()**

**{**

**cout<<"constructing derived2"<<endl;**

**}**

**~derived2()**

**{**

**cout<<"destructing derived2"<<endl;**

**}**

**};**

**void main()**

**{**

**derived1 obj1;**

**derived2 obj2;**

**getch();**

**}**

**//OUTPUT**

**constructing base1**

**constructing base2**

**constructing derived1**

**constructing base2**

**constructing base1**

**constructing derived2**

**destructing derived2**

**destructing base1**

**destructing base2**

**destructing derived1**

**destructing base2**

**destructing base1**

**/\*43.C++ Program that illustrates how run time polymorphism is achieved using virtual functions.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**class base**

**{**

**public:**

**int i;**

**base(int n)**

**{**

**i=n;**

**}**

**virtual void function()**

**{**

**cout<<"\n In base version of function \n";**

**cout<<i;**

**}**

**};**

**class derived1:public base**

**{**

**public:**

**derived1(int x):base(x){}**

**void function()**

**{**

**cout<<"\n In derived version of function double the value \n";**

**cout<<i+i;**

**}**

**};**

**void main()**

**{**

**base \*p;**

**int num;**

**clrscr();**

**cout<<"Enter a number \n";**

**cin>>num;**

**base obj(num);**

**derived1 d1\_obj(num);**

**p=&obj;**

**p->function();**

**p=&d1\_obj;**

**p->function();**

**getch();**

**}**

**//OUTPUT**

**Enter a number**

**5**

**In base version of function**

**5**

**In derived version of function double the value**

**10**

**/\*44.C++ Program that illustrates role of virtual base class in building class hierarchy.\*/**

**//PROGRAM**

**#include<iostream.h>**

**#include<conio.h>**

**class Base**

**{**

**public:**

**int i;**

**};**

**class Derived1:virtual public Base**

**{**

**public:**

**int j;**

**};**

**class Derived2:virtual public Base**

**{**

**public:**

**int k;**

**};**

**class Derived3:public Derived1,public Derived2**

**{**

**public:**

**int sum()**

**{**

**return(i+j+k);**

**}**

**};**

**void main()**

**{**

**Derived3 obj;**

**clrscr();**

**obj.i=150;**

**obj.j=170;**

**obj.k=190;**

**cout<<"sum is="<<obj.sum();**

**getch();**

**}**

**//OUTPUT**

**sum is 510**

**/\*45.C++ Program that illustrates the role of abstract class in building class hierarchy.\*/**

**//PROGRAM**

**#include <iostream.h>**

**#include <conio.h>**

**const int max = 80;**

**class first**

**{**

**protected:**

**char name[max];**

**char cls [max] ;**

**public:**

**virtual void insert()=0;**

**virtual void show()=0;**

**};**

**class second: public first**

**{**

**protected:**

**float fees;**

**public:**

**void insert()**

**{**

**cout<<"\n INPUT\n";**

**cout<<"Name : ";**

**cin>>name;**

**cout<<"Class: ";**

**cin>>cls;**

**cout<<"Fees :";**

**cin>>fees;**

**}**

**void show()**

**{**

**cout<<"\n OUTPUT\n";**

**cout<<"\nName : "<<name<<"\n";**

**cout<<"Class: "<<cls<<"\n";**

**cout<<"Fees : "<<fees<<"\n";**

**}**

**};**

**void main()**

**{**

**clrscr();**

**second s1;**

**s1.insert();**

**s1.show();**

**getch();**

**}**

**//OUTPUT**

**INPUT**

**Name : swapnika**

**Class : cse**

**Fees : 35000**

**OUTPUT**

**Name :swapnika**

**Class :cse**

**Fees :35000**