

1. Write a java program to find the distance between two pointers

Program code

```
import java.io.*;
import java.lang.*;

class Distance

{
public static void main(String args[])
{
int x1=0,x2=0,y1=0,y2=0;
double dis;
DataInputStream d=new DataInputStream(System.in);
try
{
System.out.println("enter first point(x1,y1)");
x1=Integer.parseInt(d.readLine());
y1=Integer.parseInt(d.readLine());
System.out.println("enter second point(x2,y2)");
x2=Integer.parseInt(d.readLine());
y2=Integer.parseInt(d.readLine());
}
catch(Exception e){}
dis=Math.sqrt(Math.pow((x2-x1),2)+Math.pow((y2-y1),2));
System.out.println("distance="+dis);
}
}
```

Output:

```
C:\java\manju>javac Distance.java
C:\java\manju>java Distance
enter first point(x1,y1)
2
3
enter second point(x2,y2)
5
7
distance=5.0
```

2. Write a program to find the sum, difference, product, quotient and remainder of two numbers passed as command line argument.

Program code

```
class CommandlineExample
{
public static void main(String args[])
{
int a,b;
a=Integer.parseInt(args[0]);
b=Integer.parseInt(args[1]);

System.out.println("Sum="+(a+b));
System.out.println("Difference="+(a-b));
System.out.println("Product="+a*b);
System.out.println("Quotient="+((float)a/b));
System.out.println("Reminder="+a%b) ;
}
}
```

Output:

```
C:\java\manju>javac CommandlineExample.java
C:\java\manju>java CommandlineExample 10 5
Sum=15
Difference=5
Product=50
Quotient=2.0
Reminder=0
```

3. Write a java program to display Fibonacci series up to a limit.

Program code

```
import java.io.*;
class fib
{
public static void main(String args[])
{
DataInputStream s=new DataInputStream
(System.in);
try
```

```

{
int n,i,a=0,b=1,c=0;
System.out.println("Enter the number");
n=Integer.parseInt(s.readLine());
for(i=0;c<=n;i++)
{
System.out.println(c);
a=b;
b=c;
c=a+b;
}
}
catch(Exception e)
{}
}
}

```

Output:

```

C:\java\manju>javac fib.java
C:\java\manju>java fib
Enter the number
10
0
1
1
2
3
5
8

```

4. Write a java program to display Armstrong numbers within a range.

Program code

```

import java.io.*;
class armstrong
{
public static void main(String args[])
{
DataInputStream c=new DataInputStream(System.in);
try
{
int i,u,s,t,d,l;

```

```

System.out.println("Enter the lower and upper limit");
l=Integer.parseInt(c.readLine());
u=Integer.parseInt(c.readLine());
for(i=l;i<=u;i++)
{
t=i;
s=0;
while(t>0)
{
d=t%10;
s=s+d*d*d;
t=t/10;
}
if(i==s)
System.out.println(i);
}
}
catch(Exception e)
{}
}
}

```

Output:

```

C:\java\manju>javac armstrong.java
C:\java\manju>java armstrong
Enter the lower and upper limit
100
500
153
370
371
407

```

5. Given the side of triangle, write a program to check whether the triangle is equilateral, isosceles or scalene and find its area.

Program code

```

import java.io.*;
class trianglearea
{
public static void main(String args[])
{

```

```
DataInputStream d=new DataInputStream
```

```
(System.in);  
try  
{  
int a,b,c;  
double s,area;  
System.out.println("Enter the three  
  
sides");  
a=Integer.parseInt(d.readLine());  
b=Integer.parseInt(d.readLine());  
c=Integer.parseInt(d.readLine());  
if((a==b)&&(b==c))  
System.out.println("Equilateral");  
else if((a==b) || (b==c) || (c==a))  
System.out.println("Isosceles");  
else  
System.out.println("Scalene");  
s=(a+b+c)/2;  
area=Math.sqrt((s-a)*(s-a)+(s-b)*(s-  
  
b)+(s-c)*(s-c));  
System.out.println("Area="+area);  
}  
catch(Exception e)  
{  
}  
}
```

Output:

```
C:\java\manju>javac trianglearea.java
```

```
C:\java\manju>java trianglearea
```

```
Enter the three sides
```

```
3
```

```
3
```

```
6
```

```
Isosceles
```

```
Area=4.242640687119285
```

6. Read an array of 10 or more numbers and write a program to find the
 - a. Smallest element in the array.
 - b. Largest element in the array.
 - c. Second largest element in the array.

Program code

```
import java.io.*;

class arrayoperation

{

public static void main(String args[])

{

DataInputStream d=new DataInputStream(System.in);

int a[]=new int[20];

int temp=0,n=0,i=0,j=0;

try

{

System.out.println("enter limit");

n=Integer.parseInt(d.readLine());

System.out.println("enter number");

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

a[i]=Integer.parseInt(d.readLine());

}

catch(Exception e){}

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

{

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

```
{
if(a[j]>a[j+1])
{
temp=a[j];
a[j]=a[j+1];
a[j+1]=temp;
}}}
System.out.println("small:"+a[1]);
System.out.println("large:"+a[n]);
System.out.println("seclarge:"+a[n-1]);
} }
```

Output:

C:\java\manju>javac arrayoperation.java

C:\java\manju>java arrayoperation

enter limit

10

enter number

22

23

14

35

2

56

89

45

40

10

small:2

large:89

seclarge:56

7. Write a program to perform base conversion
 - a. Integer to Binary
 - b. Integer to Octal
 - c. Integer to Hexadecimal

Program code

```
import java.io.*;

class convert

{

int num;

DataInputStream d=new DataInputStream(System.in);

void getval()

{

try

{

System.out.println(" Enter the number:");

num=Integer.parseInt(d.readLine());

}

catch(Exception e)

{}

}

void convert()

{

String hexa=Integer.toHexString(num);

System.out.println("Hexa decimal Value="+hexa);

String octal=Integer.toOctalString(num);

System.out.println("octal value="+octal);

String binary=Integer.toBinaryString(num);
```

```

System.out.println("Binary value="+binary);
}
}
class baseconversion
{
public static void main(String args[])
{
convert obj=new convert();
obj.getval();
obj.convert();
}
}

```

Output:

C:\java\manju>javac baseconversion.java

C:\java\manju>java baseconversion

Enter the number:

11

Hexa decimal Value=b

octal value=13

Binary value=1011

- Write a program to merge two array.

Program code

```

import java.io.*;
public class merge
{
public static void main(String args[])
{

```

```

int array1[]=new int[10];
int array2[]=new int[10];
int m=0,n=0,i;
DataInputStream d=new DataInputStream(System.in);
try
{
System.out.println("Enter the limit of the first array:");
m=Integer.parseInt(d.readLine());
System.out.println("enter the elements of the first array:");
for(i=0;i<m;i++)
array1[i]=Integer.parseInt(d.readLine());
System.out.println("Enter the limit of the second array:");
n=Integer.parseInt(d.readLine());
System.out.println("enter elements of second array:");
for(i=0;i<n;i++)
array2[i]=Integer.parseInt(d.readLine());
}
catch(Exception e)
{
}
int length;
length=m+n;
int result[]=new int[length];
int pos=0;
for(i=0;i<m;i++)
{
result[pos]=array1[i];
pos ++;
}
for(i=0;i<n;i++)
{
result[pos]=array2[i];
pos ++;
}
System.out.println("The merged array:");
for(i=0;i<length;i++)
System.out.println(result[i]);
}
}

```

Output:

C:\java\manju>javac merge.java

```
C:\java\manju>java merge
Enter the limit of the first array:
3
enter the elements of the first array:
1
2
3
Enter the limit of the second array:
2
enter elements of second array:
4
5
The merged array:
1
2
3
4
5
```

9. Java programming code to find HCF, LCM of two numbers.

Program code

```
import java.io.*;
class LcmHcf
{
public static void main(String args[])
{
int a,b,x=0,y=0,t,lcm,hcf;
DataInputStream d=new DataInputStream(System.in);
try
{
System.out.println("enter two numbers");
x=Integer.parseInt(d.readLine());
y=Integer.parseInt(d.readLine());
}
catch(Exception e){}
a=x;
b=y;
while(b!=0)
{
t=b;
b=a%b;
```

```
a=t;
}
hcf=a;
lcm=(x*y)/hcf;
System.out.println("LCM="+lcm);
System.out.println("HCF="+hcf);
}
}
```

Output:

```
C:\java\manju>javac LcmHcf.java
C:\java\manju>java LcmHcf
enter two numbers
2
8
LCM=8
HCF=2
```

10. Write a program to find the trace and transpose of a matrix.

Program code

```
import java.io.*;
class Trace
{
public static void main(String args[])
{
int r=0,c=0,i,j,trace=0;
int m[][]=new int[5][5];
DataInputStream d= new DataInputStream(System.in);
try
{
System.out.println("Enter order of

matrix");
r=Integer.parseInt(d.readLine());
c=Integer.parseInt(d.readLine());

System.out.println("Enter elements of

matrix");
for(i=0;i<r;i++)
```

```

for(j=0;j<c;j++)
m[i][j]=Integer.parseInt(d.readLine());
}
catch(Exception e){}
System.out.println("matrix");
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
System.out.print(m[i][j]+"\\t");
}
System.out.print("\\n");
}
System.out.println("transpose");
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
System.out.print(m[j][i]+"\\t");
}
System.out.print("\\n");
}
if(r==c)
{
for(i=0;i<r;i++)
for(j=0;j<c;j++)
if(i==j)
trace=trace+m[i][j];
System.out.println("trace="+trace);
}
else
System.out.println("cannot find trace

```

because entered matrix is not a square

```

matrix");
}
}

```

Output:

C:\java\manju>javac Trace.java

```
C:\java\manju>java Trace
```

```
Enter order of matrix
```

```
2
```

```
2
```

```
Enter elements of matrix
```

```
1
```

```
0
```

```
1
```

```
1
```

```
matrix
```

```
1 0
```

```
1 1
```

```
transpose
```

```
1 1
```

```
0 1
```

```
trace=2
```