1. Write a program to count all the prime and composite numbers entered by the user.

   Sample Input:

   Enter the numbers

   4

   54

   29

   71

   7

   59

   98

   23

   Sample Output:

   Composite number:3

   Prime number:5

```
int arr[]={4,54,29,71,7,59,98,23};
int com=0,pri=0;
for(int i=0;i<arr.length;i++)
{
    int c=0;
    for(int j=1;j<arr[i];j++)
    {
        if(arr[i]%j==0)
            c++;
    }
    if(c>1)
        com++;
    else
        pri++;

}
System.out.print("Composite Number: "+com);
System.out.println("\nPrime number: "+pri);
```

2. Find the Mth maximum number and Nth minimum number in an array and then find the sum of it and difference of it.

Sample Input:
Array of elements = {14, 16, 87, 36, 25, 89, 34}
M = 1
N = 3
Sample Output:
1st Maximum Number = 89
3rd Minimum Number = 25
Sum = 114

Difference = 64

```
int arr[]={14, 16, 87, 36, 25, 89, 34};
int len=arr.length;
 for(int i=0;i<len;i++) {
     for (int j = i + 1; j < len; j++) {
         if (arr[i] > arr[j]) {
             int temp = arr[i];
             arr[i] = arr[j];
             arr[j] = temp;
         }
     }
 }
int m=1,n=3;
int max=arr[len-m];
int min=arr[n-1];
System.out.print(m+" maximum number = "+max);
System.out.print("\n"+n+" minimum number = "+min);
int sum=max+min;
int Diff=max-min;
System.out.print("\nSum = "+sum);
System.out.print("\nDifference = "+Diff);
```

3. Write a program to print the total amount available in the ATM machine with the conditions applied.

Total denominations are 2000, 500, 200, 100, get the denomination priority from the user and the total number of notes from the user to display the total available balance to the user

Sample Input:

Enter the 1st Denomination: 500
Enter the 1st Denomination number of notes: 4
Enter the 2nd Denomination: 100
Enter the 2nd Denomination number of notes: 20
Enter the 3rd Denomination: 200
Enter the 3rd Denomination number of notes: 32
Enter the 4th Denomination: 2000
Enter the 4th Denomination number of notes: 1

Sample Output:

Total Available Balance in ATM: 12400

```
int n1=500,d1=4,n2=100,d2=20,n3=200,d3=32,n4=2000,d4=1;
int Total=(n1*d1)+(n2*d2)+(n3*d3)+(n4*d4);
System.out.print("Total Available Balance in ATM: "+Total);
```

4. Write a program using choice to check

   Case 1: Given string is palindrome or not

   Case 2: Given number is palindrome or not

   Sample Input:

   Case = 1

String = MADAM

Sample Output:

Palindrome

```
String s1="MADAM";
String s2="";
int len=s1.length();
for(int i=len-1;i>=0;i--)
{
    s2=s2+s1.charAt(i);
}
if(s1.equals(s2))
    System.out.print("Palindrome");
else
    System.out.print("Not palindrome");
```

5. Write a program to convert Decimal number equivalent to Binary number and octal numbers?

Sample Input:
Decimal Number: 15
Sample Output:
Binary Number = 1111
Octal = 17

```
int dec=15;
String bin=Integer.toBinaryString(dec);
String oct=Integer.toOctalString(dec);
System.out.println("Binary number = "+bin);
System.out.print("octal number = "+oct);
```

6. In an organization they decide to give bonus to all the employees on New Year. A 5% bonus on salary is given to the grade A workers and 10% bonus on salary to the grade B workers. Write a program to enter the salary and grade of the employee. If the salary of the employee is less than $10,000 then the employee gets an extra 2% bonus on salary Calculate the bonus that has to be given to the employee and print the salary that the employee will get.

Sample Input & Output:

Enter the grade of the employee: B

Enter the employee salary: 50000

Salary=50000

Bonus=5000.0

Total to be paid:55000.0

```
Scanner input=new Scanner(System.in);
int a,b;
```

```
double bonus=0;
System.out.print("Enter the grade of the employee :");
char a1=input.next().charAt(0);
System.out.print("Enter the salary of employee :");
int b1=input.nextInt();
if(a1=='A')
{
    bonus=b1*(0.05);
    if(b1<10000)
    {
        bonus=bonus+b1*(0.02);
    }
    System.out.println("salary = "+b1);
    System.out.println("bonus = "+bonus);
    System.out.println("total to be paid ="+(b1+bonus));
}
else if(a1=='B')
{
    bonus=b1*(0.1);
    if(b1<10000)
    {
        bonus=bonus+b1*(0.02);
    }
    System.out.println("salary = "+b1);
    System.out.println("bonus = "+bonus);
    System.out.println("total to be paid ="+(b1+bonus));
}
else {
    System.out.print("Enter valid grade");
}
```

7. Write a program to print the first n perfect numbers. (Hint Perfect number means **a positive integer that is equal to the sum of its proper divisors)**

Sample Input:

N = 3

Sample Output:

First 3 perfect numbers are: 6 , 28 , 496

```
Scanner input=new Scanner(System.in);
int n=input.nextInt();
 int sum=0,temp=0;
 for(int j=2;j<=1000;j++)
 {
     if(n>temp)
         sum=1;
     for(int i=2;i<j;i++)
     {
         if(j%i==0)
             sum=sum+i;
     }
     if(sum==j)
     {
         System.out.print(j+" ");
         temp=temp+1;
```

```
    }

}
```

8. Write a program to print the first n perfect numbers. (Hint Perfect number means **a positive integer that is equal to the sum of its proper divisors)**

Sample Input:

N = 3

Sample Output:

First 3 perfect numbers are: 6 , 28 , 496

```
Scanner input=new Scanner(System.in);
int n=input.nextInt();
 int sum=0,temp=0;
 for(int j=2;j<=1000;j++)
 {
     if(n>temp)
         sum=1;
     for(int i=2;i<j;i++)
     {
         if(j%i==0)
             sum=sum+i;
     }
     if(sum==j)
     {
         System.out.print(j+" ");
         temp=temp+1;
     }

 }
```

9. Write a program to enter the marks of a student in four subjects. Then calculate the total and aggregate, display the grade obtained by the student. If the student scores an aggregate greater than 75%, then the grade is Distinction. If aggregate is 60>= and <75, then the grade is First Division. If aggregate is 50 >= and <60, then the grade is Second Division. If aggregate is 40>= and <50, then the grade is Third Division. Else the grade is Fail.

Sample Input & Output:

Enter the marks in python: 90

Enter the marks in c programming: 91

Enter the marks in Mathematics: 92

Enter the marks in Physics: 93

Total= 366

Aggregate = 91.5

DISTINCTION

```
int a1=90;
int a2=91;
int a3=92;
int a4=93;
int  total=(a1+a2+a3+a4) ;
float agg=total/4f;
System.out.println(total);
System.out.println(agg);
if(agg>75)
    System.out.println("DISTINCTION");
else if(agg>=60 && agg<75)
    System.out.println("First Division");
else if(agg>=50 && agg<60)
    System.out.println("Second Division");
else if(agg>=40 && agg<50)
    System.out.println("Third Division");
else System.out.println("Fail");
```

10. Write a program to calculate tax given the following conditions:
    a. If income is less than or equal to 1,50,000 then no tax
    b. If taxable income is 1,50,001 – 3,00,000 the charge 10% tax
    c. If taxable income is 3,00,001 – 5,00,000 the charge 20% tax
    d. If taxable income is above 5,00,001 then charge 30% tax

Sample Input:

Enter the income:200000

Sample Output:

Tax= 20000

```
Scanner input=new Scanner(System.in);
int income=input.nextInt();
float tax;
if(income<=150000)
    System.out.println("No tax");
else if(income>=150001 && income<=300000)
    System.out.println("Tax= "+income/10);
```

```
else if(income>=300001 && income<=500000)
    System.out.println("Tax= "+income/20);
else
    System.out.println("Tax= "+income/30);
```

11. Write a program to enter the marks of a student in four subjects. Then calculate the total and aggregate, display the grade obtained by the student. If the student scores an aggregate greater than 75%, then the grade is Distinction. If aggregate is 60>= and <75, then the grade is First Division. If aggregate is 50 >= and <60, then the grade is Second Division. If aggregate is 40>= and <50, then the grade is Third Division. Else the grade is Fail.

Sample Input & Output:

Enter the marks in python: 90
Enter the marks in c programming: 91
Enter the marks in Mathematics: 92
Enter the marks in Physics: 93
Total= 366
Aggregate = 91.5
DISTINCTION

```
int a1=90;
int a2=91;
int a3=92;
int a4=93;
int  total=(a1+a2+a3+a4) ;
float agg=total/4f;
System.out.println(total);
System.out.println(agg);
if(agg>75)
    System.out.println("DISTINCTION");
else if(agg>=60 && agg<75)
    System.out.println("First Division");
else if(agg>=50 && agg<60)
    System.out.println("Second Division");
else if(agg>=40 && agg<50)
    System.out.println("Third Division");
else System.out.println("Fail");
```

12. Write a program to print the multiplication table of number m up to n.
    Sample Input:
    M = 4
    N = 5
    Sample Output:
    1x4=4
    2x4=8
    3x4=12
    4x4=16
    5x4=20

```
int M=4;
int N=5;
for(int i=1;i<=N;i++)
{
    System.out.println(i+"x"+M+"="+(i*M));
}
```

13. Write a program to read the numbers until -1 is encountered. Find the average of positive numbers and negative numbers entered by user.

    Sample Input:

    Enter -1 to exit…
    Enter the number: 7
    Enter the number: -2
    Enter the number: 9
    Enter the number: -8
    Enter the number: -6
    Enter the number: -4
    Enter the number: 10
    Enter the number: -1

    Sample Output:

    The average of negative numbers is: -5.0
    The average of positive numbers is : 8.66666667

```
int i=0,j=0;
int n=0;
int s1=0,s2=0;
int possum=0,negsum=0;
while(n!=-1)
{
    n=input.nextInt();
    if(n==-1)
        break;
    if(n>0)
    {
        i++;
        s1=s1+n;
    }
    else
    {
        j++;
        s2=s2+n;
    }
}
System.out.println(i);
System.out.println(j);
double  pos=(s1/i);
double neg=s2/j;
System.out.println("The average of positive: "+pos);
System.out.println("The average of negative: "+neg);
```

Test cases:

1. -1,43, -87, -29, 1, -9
2. 73, 7-6,2,10,28,-1
3. -5, -9, -46,2,5,0
4. 9, 11, -5, 6, 0,-1
5. -1,-1,-1,-1,-1

14. Write a program to read a character until a ***** is encountered. Also count the number of uppercase, lowercase, and numbers entered by the users.
Sample Input:
Enter * to exit…
Enter any character: W
Enter any character: d
Enter any character: A
Enter any character: G
Enter any character: g
Enter any character: H
Enter any character: *
Sample Output:
Total count of lower case:2
Total count of upper case:4
Total count of numbers =0

```
Scanner input=new Scanner(System.in);
System.out.println("Enter * to exit....");
char c='0';
int lower=0,upper=0,digit=0;
while(c!='*')
{
    c=input.next().charAt(0);
    if(c>=65 && c<=90)
        upper=upper+1;
    else if(c>=97 && c<=122)
        lower=lower+1;
    else if(c>=48 && c<=57)
        digit=digit+1;
}
System.out.println("Lower: "+lower);
System.out.println("Upper: "+upper);
System.out.println("Digit: "+digit);
```

15.Write a program to calculate the factorial of number using recursive function.

Sample Input & Output:

Enter the value of n: 6

Sample Input & Output:

The factorial of 6 is: 720

```
Scanner input=new Scanner(System.in);
    int n=input.nextInt();
    int fact=1;
    for(int i=1;i<=n;i++)
    {
        fact=fact*i;
    }
    System.out.println("The factorial of "+n+" is: "+fact);
```

16.Write a Program to Find the Nth Largest Number in a array.

Sample Input:

List : {14, 67, 48, 23, 5, 62}

N = 4

Sample Output:

4th Largest number: 23

```
Scanner input = new Scanner(System.in);
int a[] = {14, 67, 48, 23, 5, 62};
int len = a.length;
Arrays.sort(a);
int N = 4;
System.out.println(N + " Largest number: " + a[len-N]);
```

17.Write a program to convert the Binary to Decimal, Octal

Sample Input:

Given Number: 1101

Sample Output:

Decimal Number: 13

Octal:15

```
Scanner input=new Scanner(System.in);
 String bin=input.nextLine();
 int dec=Integer.parseInt(bin,2);
 System.out.println("Decimal: "+dec);
 String oct=Integer.toOctalString(dec);
 System.out.println("Octal: "+oct);
```

18. Write a program to find the number of special characters in the given statement

Sample Input:

Given statement: Modi Birthday @ September 17, #&$% is the wishes code for him.

Sample Output:

Number of special Characters: 5

19. Write a Program to Remove the Duplicate Items from a array.

Sample Input:

Enter the number of elements in array:7
Enter element1:10
Enter element2:20
Enter element3:20
Enter element4:30
Enter element5:40
Enter element6:40
Enter element7:50

Sample Output:

Non-duplicate items:
[10, 20, 30, 40, 50]

```
Scanner input=new Scanner(System.in);
System.out.print("Enter the number of elements: ");
int n=input.nextInt();
int a[]=new int[n];
for(int i=0;i<n;i++)
{
    System.out.print("Enter element: ");
    a[i]=input.nextInt();
}
for(int i=0;i<n;i++)
{
    for(int j=i+1;j<n;j++)
    {
        if(a[i]==a[j])
        {
            for(int k=j;k<n-1;k++)
            {
                a[k]=a[k+1];
            }
            j--;
            n--;
        }
    }
}
for(int i=0;i<n;i++)
{
```

```
    System.out.print(a[i]+" ");
}
```

20. Bank is a class that provides method to get the rate of interest. But, rate of interest may differ according to banks. For example, SBI, ICICI and AXIS banks are providing 8.4%, 7.3% and 9.7% rate of interest. Write a Java program for above scenario.
    Sample Input SBI, 8.4
    Sample Output

    Test case

    1. SBI, 8.3
    2. ICICI, 7.3
    3. AXIS, 9.7
    4. SBI, 8.6
    5. AXIX, 7.6

```
class Bank
{
    float getROI()
    {
        return 0;
    }
}
class SBI extends Bank
{
    float getROI()
    {
        return 8.4f;
    }
}
class ICICI extends Bank
{
    float getROI()
    {
        return 8.4f;
    }
}
class AXIS extends Bank
{
    float getROI()
    {
        return 8.4f;
    }
}

public class ak {
    public static void main(String[] args) {
        Bank b;
        b = new SBI();
        System.out.println("SBI, " + b.getROI());
    }
}
```

21. Bring out the situation in which member names of a subclass hide members by the same name in the super class. How it can be resolved? Write Suitable code in Java and Implement above scenario with the Parametrized Constructor (accept int type parameter) of the Super Class can be called from Sub Class Using super () and display the input values provided.

    Sample Input : 100, 200
    Sample Output : 100, 200

```
import java.util.Scanner;
class abc
    {
        abc(int x,int y)
    {
        System.out.print(x+","+y);
    }
    }
    public class ak extends abc
    {
        ak(int x,int y)
        {
            super(x,y);
        }
        public static void main(String[] args)
        {
            Scanner input=new Scanner(System.in);
            int a1=input.nextInt();
            int b1=input.nextInt();
            ak obj=new ak(a1,b1);
        }
    }
```

22. Display Multiplication table for 5 and 10 using various stages of life cycle of the thread by generating a suitable code in Java.
    Sample Input 5, 10
    5 X 1 = 5
    5 X 2 =10
    ….
    10 X 1 =10
    10 X 2 = 20
    ….

```
import java.util.Scanner;
class A extends Thread
{
    public void run()
    {
         int n=5;
         for(int i=1;i<=n;i++)
         {
             System.out.println(n+" X "+i+" = "+(n*i));
         }
    }
}
```

```
class B extends Thread
{
    public void run()
    {
        int n=10;
        for(int i=1;i<=n;i++)
        {
            System.out.println(n+" X "+i+" = "+(n*i));
        }
    }
}
public class ak
{
    public static void main(String[] args)
    {
        Scanner input=new Scanner(System.in);
        A threadA=new A();
        B threadB=new B();
        threadA.start();
        threadB.start();
    }
}
```

Test Cases:

1. 10, 20
2. -10, -30
3. 0, 0
4. SIX, SIX
5. 9.8, 9.6

23. Using the concepts of thread with implementing Runnable interface in Java to generate Fibonacci series.
Sample Input : 5
Sample Output : 0 1 1 2 3 …..
Test Cases

    1. 7
    2. -10
    3. 0
    4. EIGHT FIVE
    5. 12.65

24. Generate a Java code to find the sum of N numbers using array and throw ArrayIndexOutOfBoundsException when the loop variable beyond the size N.

Sample Input : 5
1 2 3 4 5

Sample Output : 15
Test Cases

1. 4, 10

2. -10
3. 0
4. EIGHT SEVEN
5. 12.68

25. Using the concepts of thread with implementing Runnable interface in Java to find whether a given number is prime or not.

Sample Input : 5
Sample Output : 5 is Prime

Sample Output : 15
Test Cases

1. 4
2. -10
3. 0
4. EIGHT SEVEN
5. 11.48

- **26. Given a string `s` consisting of words and spaces, return the length of the** last **word *in the string.* A** word **is a maximal substring consisting of non-space characters only. There will be at least one word, consists of only English letters and spaces ' '.**

Example 1:
Input: s = "Hello World"
Output: 5
Explanation: The last word is "World" with length 5.

Test Case

Test Case	Inputs-1
1.	Maximal Substring Consisting
2.	**lea@st one wor2d**
3.	1254 98076
4.	& * ( ) % # $
5.	letters and spaces

```
import java.util.Arrays;
import java.util.Locale;
import java.util.Scanner;
public class ak {
    public static void main(String[] args)
    {
        Scanner input=new Scanner(System.in);
        String name=input.next();
        name.split("NULL");
        name=name.trim();
        int len=0;
        for(int i=name.length()-1;i>=0;i--)
        {
            if(name.charAt(i)==' ')
```

```
                break;
            else
                len++;
        }
        System.out.println(len);
    }
}
```