

CS162

ASSIGNMENT 1

NAME:

ARCHIT AGRAWAL

ROLL NO. :

202052307

SECTION:

A

1. Creating a class named Main and defining the functions listed below in it

1. division(int, int)

2. gcd(int, int)

3. lcm(int, int)

4. power(int ,int)

5. max(int[])

6. min(int[])

7. abs(int)

8. factorial(int)

9. sum(int[]) (e.g. sum[1, 7, 3] output:- $1+7+3 = 11$)

10. sumOfDigits(int) (e.g. sumOfDigits(4785) output:- $4+7+8+5 = 24$)

11. sqrt(int) (Note:- Don't use Math.sqrt())

12. isPrime(int)

13. isLeapYear(int)

14. isPalindrome(int)

15. isArmstrong(int)

16. ArithmeticSequenceSum(int a, int d, int n)

"a" as the first term , "d" the common difference between the terms and "n" is the total number of terms in the sequence.

17. GeometricSequenceSum(int a, int r,int n)

"a" as a start term , "r" as a common ratio and "n" is the total number of terms in the sequence.

18. Linear Search (return an index of element if found otherwise return -1)

CODE:

```
import java.util.*;

public class Main {

    public static double division(int a,int b){
        if(b!=0){
            double c = ((double)a)/b;
            return c;
        }
        else{
            System.out.println("Division by 0 is undefined");
            return -1;
        }
    }

    public static int gcd(int a,int b){

        if(a == 0 && b == 0){
            System.out.println("GCD is not defined");
            return -1;
        }
        else{
            int temp;
            while(b > 0){
                temp = b;
                b = a % b;
                a = temp;
            }
            return a;
        }
    }

    public static int lcm(int a,int b){
        if(a == 0 && b == 0){
            System.out.println("LCM is not defined");
            return -1;
        }
        else{
            int temp;
            int tempA = a;
            int tempB = b;

            while(tempB > 0){
                temp = tempB;
                tempB = tempA % tempB;
                tempA = temp;
            }
            return (a * b)/tempA;
        }
    }

    public static double power(int a,int p) {
        long x = 1;
```

```
        int temp = p;
        if(p < 0) p = -1 * p;
        while(p != 0){
            x = x * a;
            p--;
        }
        return temp > 0 ? x : 1.0/x;
    }

    public static int max(int a[]){
        int max = a[0];
        int x = 0;
        for(x = 1; x < a.length; x++){
            if(a[x] > max)
                max = a[x];
        }
        return max;
    }

    public static int min(int a[])
    {
        int min=a[0];
        int x=0;
        for(x=1;x<a.length;x++){
            {
                if(a[x]<min)
                    min=a[x];
            }
        }
        return min;
    }

    public static int abs(int a){

        return (a > 0) ? a : (0 - a);
    }

    public static long factorial(int num){
        if(num < 0){
            System.out.println("Factorial is undefined");
            return -1;
        }
        int i;
        long f = 1;
        for(i = num ; i >= 1 ; i--){
            f = f*i;
        }
        return f;
    }

    public static int sum(int a[]) {
        if(a.length <= 0){
            System.out.println("Array size cannot be less than or equal to
0");
            return -1;
        }
        int s = 0, x;
        for(x = 0; x < a.length; x++){
            s += a[x];
        }
        return s;
    }
}
```

```
public static int sumOfDigits(int n){
    int s = 0;
    while(n>0){
        s += (n % 10);
        n = n/10;
    }
    return s;
}

public static double sqrt(int n) {
    if(n < 0){
        System.out.println("Square root is undefined");
        return -1;
    }
    double min = 0.0, max = n;
    double sqrt = (min + max)/2.0;
    while(true){
        sqrt = (min + max)/2.0;

        if(sqrt * sqrt < n - 0.00001){
            min = sqrt;
        } else if(sqrt * sqrt > n + 0.00001) {
            max = sqrt;
        } else break;
    }
    return sqrt;
}

public static boolean isPrime(int n) {
    int x, f = 0;
    for (x = 2; x < n; x++) {
        if (n % x == 0) {
            f = 1;
            break;
        }
    }
    return f == 0;
}

public static boolean isLeapYear(int year){

    return (year % 400 == 0) || (year % 4 == 0 && year % 100 != 0);

}

public static boolean isPalindrome(int num){
    int rev = 0;
    int n = num;
    while(n > 0){
        rev = rev * 10 + n % 10;
        n = n/10;
    }
    return rev == num;
}

public static boolean isArmstrong(int num) {
    int s = 0, r, n = num;
    while(n > 0){
        r = n%10;
        s = s + (r*r*r);
    }
}
```

```
        n = n/10;
    }
    return s == num;
}

public static long arithmeticSequenceSum(int a, int d, int n){

    return ((2 * a + (n - 1) * d) * n) / 2;
    // ((2 * a + (n - 1) * d) * n) is always even

}

public static long geometricSequenceSum(int a, int r,int n) {
    long sum = 0;
    int c = a;
    int x = 0;
    while(x < n){
        sum = sum + c;
        c = c * r;
        x++;
    }
    return sum;
}

public static int linearSearch(int a[], int num){

    for(int i = 0; i < a.length; i++){
        if(a[i] == num) return i;
    }

    return -1;
}

public static void main(String args[]) {
    Scanner Sc = new Scanner(System.in);
    int a, b;
    System.out.println("Enter two numbers you want to divide ");
    a = Sc.nextInt();
    b = Sc.nextInt();
    System.out.println("
");
    System.out.println(a + "/" + b + " = " +division(a,b));
    System.out.println("
");
    System.out.println("Enter two numbers whose GCD you want to find");
    a = Sc.nextInt();
    b = Sc.nextInt();
    System.out.println("
");
    System.out.println("GCD of " +a + " and " +b + " is : "+gcd(a,b));
    System.out.println("
");
    System.out.println("Enter two numbers whose LCM you want to find");
    a = Sc.nextInt();
    b = Sc.nextInt();
    System.out.println("
");
    System.out.println("LCM of " +a + " and " +b + " is : "+lcm(a,b));
    System.out.println("
");
}
```

```
        System.out.println("Enter the base a and exponent b to calculate a  
to the power b");  
        a = Sc.nextInt();  
        b = Sc.nextInt();  
        System.out.println("  
");  
        System.out.println(a + " to the power " + b + " = " + power(a, b));  
        System.out.println("  
");  
  
        System.out.println("Enter the size of array ");  
        int s = Sc.nextInt();  
        int arr[] = new int[s];  
        System.out.println("Enter the values of array");  
        for(int x = 0; x < s; x++)    arr[x] = Sc.nextInt();  
  
        System.out.println("  
");  
        System.out.println("Maximum value in array :"+max(arr));  
        System.out.println("Minimum value in array :"+min(arr));  
        System.out.println("Sum of values in array :"+sum(arr));  
        System.out.println("  
");  
  
        System.out.println("Enter a number whose absolute value you want to  
find");  
        a = Sc.nextInt();  
        System.out.println("  
");  
        System.out.println("Absolute value of "+a+" : "+abs(a));  
        System.out.println("  
");  
        System.out.println("Enter a number whose factorial you want to  
find");  
        a = Sc.nextInt();  
        System.out.println("  
");  
        System.out.println(a+"! = "+factorial(a));  
        System.out.println("  
");  
        System.out.println("Enter a number the sum of whose digits you want  
to find");  
        a=Sc.nextInt();  
        System.out.println("  
");  
        System.out.println("Sum of digits  of "+a+" : "+sumOfDigits(a));  
        System.out.println("  
");  
        System.out.println("Enter an integer number whose square root you  
want to find");  
        a = Sc.nextInt();  
        System.out.println("  
");  
        System.out.printf("Square root of "+a+" : %.4f\n", sqrt(a));  
        System.out.println("  
");  
        System.out.println("Enter a number to check if it is prime or  
not");  
        System.out.println("  
");  
        a = Sc.nextInt();
```

```
        if(isPrime(a)) System.out.println(a+" is a prime number");
        else System.out.println(a+" is not a prime number");

        System.out.println("
");
        System.out.println("Enter an year to check if it is leap year or
not");
        a = Sc.nextInt();
        System.out.println("
");
        if(isLeapYear(a)) System.out.println(a+" is a Leap Year");
        else System.out.println(a+" is not a Leap Year");

        System.out.println("
");
        System.out.println("Enter a number to check if it is a palindrome
or not");
        a = Sc.nextInt();
        System.out.println("
");
        if(isPalindrome(a)) System.out.println(a+" is a Palindrome");
        else System.out.println(a+" is not a Palindrome");
        System.out.println("
");
        System.out.println("Enter a number to check if it is an Armstrong
number or not ");
        a = Sc.nextInt();
        System.out.println("
");
        if(isArmstrong(a)) System.out.println(a+" is an Armstrong Number");
        else System.out.println(a+" is not an Armstrong Number");
        System.out.println("
");

        System.out.println("Enter first term,common difference and number
of terms of an arithmetic sequence whose sum you want to find");
        a = Sc.nextInt();
        int d = Sc.nextInt();
        int n = Sc.nextInt();
        System.out.println("
");
        System.out.println("Sum of Arithmetic Sequence =
"+arithmeticSequenceSum(a, d, n));
        System.out.println("
");
        System.out.println("Enter first term,common ratio and number of
terms of an geometric sequence whose sum you want to find");
        a = Sc.nextInt();
        int r = Sc.nextInt();
        n = Sc.nextInt();
        System.out.println("
");
        System.out.println("Sum of Geometric Sequence =
"+geometricSequenceSum(a, r, n));
        System.out.println("
");

        System.out.println("Enter size of array");
        s = Sc.nextInt();
        int brr[] = new int[s];
        System.out.println("Enter array elements");
```



```
        for(int x = 0; x < s; x++){
            brr[x]=Sc.nextInt();
        }
        System.out.println("Enter number to search");
        int num = Sc.nextInt();
        System.out.println("
");
        System.out.println("The number is found in the array at the index
(0 based) "+ linearSearch(brr, num));
        System.out.println("
");
    }
}
```

OUTPUT:

```
Enter two numbers you want to divide
9 5
9/5 = 1.8

Enter two numbers whose GCD you want to find
9 15
GCD of 9 and 15 is : 3

Enter two numbers whose LCM you want to find
8 12
LCM of 8 and 12 is : 24

Enter the base a and exponent b to calculate a to the power b
5 -3
5 to the power -3 = 0.008

Enter the size of array
5
Enter the values of array
-1 2 3 -5 4

Maximum value in array :4
Minimum value in array :-5
Sum of values in array :3
```

Enter a number whose absolute value you want to find

-65

Absolute value of -65 : 65

Enter a number whose factorial you want to find

9

9! = 362880

Enter a number the sum of whose digits you want to find

986543

Sum of digits of 986543 : 35

Enter an integer number whose square root you want to find

91

Square root of 91 : 9.5394

Enter a number to check if it is prime or not

89

89 is a prime number

Enter an year to check if it is leap year or not

1800

```
1800 is not a Leap Year
```

```
Enter a number to check if it is a palindrome or not
```

```
9889
```

```
9889 is a Palindrome
```

```
Enter a number to check if it is an Armstrong number or not
```

```
370
```

```
370 is an Armstrong Number
```

```
Enter first term,common difference and number of terms of an arithmetic sequence whose sum you want to find
```

```
5 7 8
```

```
Sum of Arithmetic Sequence = 236
```

```
Enter first term,common ratio and number of terms of an geometric sequence whose sum you want to find
```

```
3 5 7
```

```
Sum of Geometric Sequence = 58593
```

```
Enter size of array
```

```
6
```

```
Enter array elements
```

```
1 5 9 -5 6 3
```

```
Enter number to search
```

```
-5
```

```
The number is found in the array at the index (0 based) 3
```

```
Process finished with exit code 0
```

2. Creating a class named ReverseArray

CODE:

```
//package com.company;  
import java.util.*;  
public class ReverseArray {  
    public static void main(String args[]){  
        Scanner Sc=new Scanner(System.in);  
        System.out.println("Enter size of array");  
        int s = Sc.nextInt();  
        int arr[] = new int[s];  
        int x,temp;  
        System.out.println("Enter array elements");  
        for(x = 0; x < s; x++) {  
            arr[x] = Sc.nextInt();  
        }  
        for(x = 0; x < s/2; x++) {  
            temp = arr[x];  
            arr[x] = arr[s-1-x];  
            arr[s-1-x] = temp;  
        }  
        System.out.println("The reversed array is");  
        for(x = 0; x < s; x++) {  
            System.out.print(arr[x]+" ");  
        }  
    }  
}
```

OUTPUT:

```
Enter size of array  
8  
Enter array elements  
1 3 5 7 -4 -8 9 11  
The reversed array is  
11 9 -8 -4 7 5 3 1  
Process finished with exit code 0  
|
```

3. Creating a class named MaxAbsDifference

CODE:

```
//package com.company;

import java.util.*;
public class MaxAbsDifference {
    public static void main(String args[]){
        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter size of array");
        int s = Sc.nextInt();
        int arr[] = new int[s];int x;
        System.out.println("Enter array elements");
        for(x = 0;x < s; x++) {
            arr[x] = Sc.nextInt();
        }
        int min = arr[0], max = arr[0];
        for(x = 1; x < s; x++){
            if(arr[x] > max) max = arr[x];
            if(arr[x] < min) min = arr[x];
        }
        System.out.println("The maximum absolute difference among the array
elements is "+(max - min));
    }
}
```

OUTPUT:

```
Enter size of array
9
Enter array elements
-1 -8 5 3 2 4 5 18 9
The maximum absolute difference among the array elements is 26

Process finished with exit code 0
|
```

4. Creating a class named AddMatrix

CODE:

```
//package com.company;

import java.util.*;
public class AddMatrix {
    public static void main(String args[]){
        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter the number of rows and columns of matrix
1");
        int m = Sc.nextInt();
        int n = Sc.nextInt();
        int r, c;
```

```
int a[][] = new int[m][n];
System.out.println("Enter the elements of matrix 1 row-wise");
for(r = 0; r < m; r++) {
    for(c = 0; c < n; c++) {
        a[r][c] = Sc.nextInt();
    }
}
//printing the matrix 1
System.out.println("Matrix 1");
for(r = 0; r < m; r++) {
    for(c = 0; c < n; c++) {
        System.out.print(a[r][c] + "\t");
    }
    System.out.println();
}

System.out.println("Enter the number of rows and columns of matrix
2");

int p = Sc.nextInt();
int q = Sc.nextInt();
int b[][] = new int[p][q];

System.out.println("Enter the elements of matrix 2 row-wise");

for(r = 0; r < p; r++) {
    for(c = 0; c < q; c++) {
        b[r][c] = Sc.nextInt();
    }
}
//printing matrix 2
System.out.println("Matrix 2");
for(r = 0; r < p; r++) {
    for(c = 0; c < q; c++) {
        System.out.print(b[r][c] + "\t");
    }
    System.out.println();
}

if(m != p || n != q) System.out.println("Matrices can not be
added");
else {
    System.out.println("The sum of the matrices is ");
    int sum[][] = new int[m][n];

    for (r = 0; r < m; r++) {
        for (c = 0; c < n; c++) {
            sum[r][c] = a[r][c] + b[r][c];
            System.out.print(sum[r][c] + "\t");
        }
        System.out.println();
    }
}
}
```

OUTPUT:

```
Enter the number of rows and columns of matrix 1
3 3
Enter the elements of matrix 1 row-wise
1 2 3
4 5 6
7 8 9
Matrix 1
1  2  3
4  5  6
7  8  9
Enter the number of rows and columns of matrix 2
3 3
Enter the elements of matrix 2 row-wise
9 8 7
6 5 4
3 2 1
Matrix 2
9  8  7
6  5  4
3  2  1
The sum of the matrices is
10 10 10
10 10 10
10 10 10

Process finished with exit code 0
|
```

5. Creating a class named MultiplyMatrix

CODE:

```
//package com.company;
import java.util.*;

public class MultiplyMatrix {
    public static void main(String args[]){
        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter the number of rows and columns for matrix
1");
        int m1 = Sc.nextInt();
        int n1 = Sc.nextInt();

        int r,c;
        int a[][] = new int[m1][n1];

        System.out.println("Enter the elements of matrix 1 row-wise");
        for(r = 0; r < m1; r++){
            for(c = 0; c < n1; c++){
                a[r][c] = Sc.nextInt();
            }
        }
    }
}
```

```
    }  
}  
  
//printing the matrix 1  
System.out.println("Matrix 1");  
for(r = 0; r < m1; r++) {  
    for(c = 0; c < n1; c++) {  
        System.out.print(a[r][c] + "\t");  
    }  
    System.out.println();  
}  
  
System.out.println("Enter the number of rows and columns for matrix  
2");  
  
int m2 = Sc.nextInt();  
int n2 = Sc.nextInt();  
  
int b[][] = new int[m2][n2];  
  
System.out.println("Enter the elements of matrix 2 row-wise");  
for(r = 0; r < m2; r++){  
    for(c = 0; c < n2; c++){  
        b[r][c] = Sc.nextInt();  
    }  
}  
  
//printing matrix 2  
System.out.println("Matrix 2");  
for(r = 0; r < m2; r++) {  
    for(c = 0; c < n2; c++) {  
        System.out.print(b[r][c] + "\t");  
    }  
    System.out.println();  
}  
  
if(n1 != m2){  
    System.out.println("Matrices can not be multiplied.");  
    System.exit(1);  
}  
  
int p[][] = new int[m1][n2];  
int s;  
System.out.println("The product of the matrices is");  
for(r = 0; r < m1; r++) {  
    for(c = 0; c < n2; c++) {  
        s = 0;  
        for(int x = 0; x < m2; x++) {  
            s += a[r][x] * b[x][c];  
        }  
        p[r][c] = s;  
        System.out.print(p[r][c] + "\t");  
    }  
    System.out.println();  
}  
}
```


OUTPUT:

```
Enter the number of rows and columns for matrix 1
3 3
Enter the elements of matrix 1 row-wise
1 2 3
4 5 6
7 8 9
Matrix 1
1 2 3
4 5 6
7 8 9
Enter the number of rows and columns for matrix 2
3 2
Enter the elements of matrix 2 row-wise
1 2
3 4
5 6
Matrix 2
1 2
3 4
5 6
The product of the matrices is
22 28
49 64
76 100

Process finished with exit code 0
|
```

6. Creating a class named Palindrome

CODE:

```
//package com.company;

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

        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter a string to check if it is a palindrome
or not");
        String str = Sc.nextLine();
        int x, c = 0, l = str.length();
```

```
for(x = 0; x < l/2; x++){
    if(str.charAt(x) != str.charAt(l-1-x)){
        c--;
        break;
    }
}
if(c == 0) System.out.println("PALINDROME");
else System.out.println("NOT A PALINDROME");
}
```

OUTPUT:

```
Enter a string to check if it is a palindrome or not
aabbabbaa
PALINDROME
```

```
Process finished with exit code 0
```

```
Enter a string to check if it is a palindrome or not
aabbbaaab
NOT A PALINDROME
```

```
Process finished with exit code 0
```

7. Creating a class named IsNumeric

CODE:

```
//package com.company;

import java.util.*;
public class IsNumeric {
    public static void main(String args[]){
        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter a string to check if it contains only
numbers or not");
        String str = Sc.nextLine();
        int x, c = 0;
        for(x = 0; x < str.length(); x++) {
            char ch = str.charAt(x);
            if(ch < 48 || ch > 57) {
                c--;
                break;
            }
        }
    }
}
```

```
    }  
    }  
    if(c==0) System.out.println("NUMERIC STRING");  
    else System.out.println("NOT A NUMERIC STRING");  
    }  
}
```

OUTPUT:

```
Enter a string to check if it contains only numbers or not  
11345h19323f  
NOT A NUMERIC STRING  
  
Process finished with exit code 0  
|
```

```
Enter a string to check if it contains only numbers or not  
1197538462891346  
NUMERIC STRING  
  
Process finished with exit code 0  
|
```

8. Creating a class named IsEqual

CODE:

```
//package com.company;  
  
import java.util.*;  
public class IsEqual {  
    public static void main(String args[]){  
        Scanner Sc = new Scanner(System.in);  
        System.out.println("Enter string 1");  
        String s1 = Sc.nextLine();  
        System.out.println("Enter string 2");  
        String s2 = Sc.nextLine();  
        if(s1.length() != s2.length()){  
            System.out.println("NOT EQUAL");  
        }  
    }  
}
```

```
        System.exit(1);
    }
    int x, c = 0;
    for(x = 0; x < s1.length(); x++) {
        if(s1.charAt(x) != s2.charAt(x)){
            c++;
            break;
        }
    }
    if(c == 0) System.out.println("EQUAL STRINGS");
    else System.out.println("NOT EQUAL");
}
```

OUTPUT:

```
Enter string 1
MynameisArchitAgrawal
Enter string 2
MynameisArchitAgrawal
EQUAL STRINGS

Process finished with exit code 0
|
```

```
Enter string 1
eihaoiefhapeihf
Enter string 2
paeihfpaiehfea
NOT EQUAL

Process finished with exit code 1
|
```

10. Creating a class named CharSort

CODE:

```
//package com.company;

import java.util.*;

public class CharSort {
    public static void main(String args[]){

        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter the string you want to sort");
        String str = Sc.nextLine();
        int x, y, l=str.length();
        char temp;
        char[]a = str.toCharArray();
        str = "";
        for(x = 0;x < l-1; x++) {
            for(y = 0; y < l-1-x ; y++) {
                if(a[y]>a[y+1]) {
                    temp = a[y];
                    a[y] = a[y+1];
                    a[y+1] = temp;
                }
            }
            str = a[l-1-x] + str;
        }
        str = a[0]+str;
        System.out.println("Sorted String is: " +str);
    }
}
```

OUTPUT:

```
Enter the string you want to sort
akhebfaksjbfa
Sorted String is: aaabbefhjkks

Process finished with exit code 0
|
```

11.Creating a class named IsAnagram

CODE:

```
//package com.company;

import java.util.*;

public class IsAnagram{
```

```
public static void main(String args[]) {
    Scanner Sc=new Scanner(System.in);
    System.out.println("Enter String 1");
    String s1=Sc.nextLine();
    System.out.println("Enter String 2");
    String s2=Sc.nextLine();
    int a1[]=new int[256];
    int a2[]=new int[256];

    if(s1.length() != s2.length()) {
        System.out.println("The two strings are not anagrams.");
        System.exit(1);
    }
    int x;
    for(x = 0; x < s1.length(); x++) {
        a1[s1.charAt(x)]++;
        a2[s2.charAt(x)]++;
    }
    int c = 0;
    for(x = 0;x < 256;x++) {
        if(a1[x] != a2[x]) {
            c++;
            break;
        }
    }
    if(c == 0) System.out.println("The two strings are anagrams");
    else System.out.println("The two strings are not anagrams");
}
```

OUTPUT:

```
Enter String 1
hello world
Enter String 2
dlrow olleh
The two strings are anagrams

Process finished with exit code 0
|
```

```
Enter String 1
hello world
Enter String 2
world helll
The two strings are not anagrams.

Process finished with exit code 1
|
```

12.Creating a class named SingleCharacters

CODE:

```
//package com.company;

import java.util.*;
public class SingleCharacters {
    public static void main(String args[]){
        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter a string");
        String str = Sc.nextLine();
        int a[] = new int[256];
        int x;
        for(x = 0; x < str.length();x++) {
            a[str.charAt(x)]++;
        }
        int count=0;
        for(x = 0; x < 256; x++) {
            if(a[x] == 1) count++;
        }
        System.out.println("Number of single occurring characters is
"+count);
    }
}
```

OUTPUT:

```
Enter a string
helloworld
Number of single occurring characters is 5

Process finished with exit code 0
|
```

13.Creating a class named Binary

CODE:

```
package com.company;

import java.util.*;
public class Binary {
    public static void main(String args[]){
```

```
Scanner Sc = new Scanner(System.in);
String str = Sc.nextLine();
int x, c = 1;
long dec = 0;
for(x = str.length()-1; x >= 0; x--) {
    dec += (str.charAt(x)-48) * c;
    c *= 2;
}
System.out.println("The decimal number is "+dec);
}
```

OUTPUT:

```
Enter a string containing 1's and 0's
110101
The decimal number is 53

Process finished with exit code 0
|
```

14.Creating a class named Decimal

CODE:

```
//package com.company;
import java.util.*;
public class Decimal {
    public static void main(String args[]) {
        Scanner Sc = new Scanner(System.in);
        System.out.println("Enter an integer whose binary equivalent you want to find");
        int num = Sc.nextInt();
        System.out.print("Binary Equivalent of "+num+" = ");
        String bin = "";
        while(num > 0) {
            bin = (char)(num%2+48) + bin;
            num = num/2;
        }
        System.out.println(bin);
    }
}
```


OUTPUT:

```
Enter an integer whose binary equivalent you want to find
67
Binary Equivalent of 67 = 1000011

Process finished with exit code 0
|
```

15.Creating a class named RightTriangle

CODE:

```
//package com.company;

import java.util.*;
public class RightTriangle {
    public static void main(String args[]) {
        System.out.println("Enter the number of rows of pattern");
        Scanner Sc = new Scanner(System.in);
        int rows = Sc.nextInt();
        int r,c;
        for(r = 0; r < rows; r++) {
            for(c = 0; c <= r; c++) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}
```

OUTPUT:

```
Enter the number of rows of pattern
5
*
* *
* * *
* * * *
* * * * *

Process finished with exit code 0
|
```

