LAB ASSIGNMENT-1

Q1: Write a java program that can take a positive integer greater than 2 as input and write out the number of times one must repeatedly divide this number by 2 before getting a value less than 2.

Ans

```
import java.util.*;
public class Q1
{
  public static void main(String[] args)
  {
    Scanner sc= new Scanner(System.in);
    System.out.print("Enter positive integer greater
    than 2:");
    int n=sc.nextInt();
    int count=0;
    int x=n;
    while(n>2)
    {
      a/=2;
      count+=1;
    }
   System.out.println("The no. of time the no."+
   x+"repeatedly divide by 2 is:"+count);
   sc.close();
  }
}
```

Name: Suryamadhab Moharana Regd. Number: 2341013398

Output:

Enter positive integer greater than 2: 67.
The number of times one must repeatedly divide by 2 is 6.

Question 2 : The body mass Index (BMI) is commonly used by health and nutrition professionals to estimate human body fat in populations. It is computed by taking the individual's weight (mass) in kilograms and dividing it by the square of height in metres. ie. BMI = Weight(kg) / Height^2(m^2). Write a Java program by using conditional statement to show the category for a given BMI.

Category	BMI
Less than 18.5	Underweight
18.5 to 24.9	Normal weight
25.0 to 29.9	Overweight
30.0 to more	Obese.

Ans.
```
import java.util.*;
public class Q2
{
  public static void main(String[] args)
  {
    Scanner sc=new Scanner(System.in);
    System.out.print("Enter the Weight(in kg):");
    double weight=sc.nextInt();
    System.out.print("Enter the Height(in m):");
    double height=sc.nextInt();
    double bmi= weight/(Math.pow(h,2));
    if (bmi< 18.5)
       System.out.println("Underweight");
    else if (bmi>=18.5 && bmi<24.9)
       System.out.println("Normal weight");
```

```
        else if (bmi >= 25.0 && bmi < 29.9)
            System.out.println("Overweight");
        else
            System.out.println("Obese");
        sc.close();
    }
}
```

Output:

```
Enter the weight in kg: 96
Enter the height in m: 1.4
Obese
```

Question 3: WAP to check whether a no. is a spy number or not.

For ex: Let a no. be 132.

$1+3+2=6$

$1*3*2=6$

Ans.
```
import java.util.*;
public class Q3
{
  public static void main(String[] args)
  {
    Scanner sc=new Scanner(System.in);
    System.out.print("Enter a number:");
    int n=sc.nextInt();
    int s=0, p=1, x=n;
    while(n>0)
    {
      int r=n%10;
      sum+=r;
      p*=r;
      n/=10;
    }
    if(s==p)
      System.out.println(x+" is a spy number");
    else
      System.out.println(x+" is not a spy number");
  }
}
```

Output: Enter a number: 132

132 is a spy number.

Q4: WAP that outputs all possible strings formed by using the characters 'c', 'a', 'r', 'b', 'o', 'n' exactly once.

Ans.

```
public class Q4
{
  public static void main (String[] args)
  {
    String s = "carbon";
    for (int i=0; i<6; i++)
    {
      for (int j=0; j<6; j++)
      {
        for (int k=0; k<6; k++)
        {
          for (int l=0; l<6; l++)
          {
            for (int m=0; m<6; m++)
            {
              for (int n=0; n<6; n++)
              {
                if (i!=j && i!=k && i!=l && i!=m && i!=n
                  && j!=k && j!=l && j!=m && j!=n
                  && k!=l && k!=m && k!=n
                  && l!=m && l!=n
                  && m!=n)
                  System.out.println(s.charAt(i)+" "
                + s.charAt(j)+" "+s.charAt(k)+" "+s.charAt(l)
                +" "+s.charAt(m)+" "+s.charAt(n));
              }
            }
          }
        }
      }
    }
  }
}
```

Name: Suryamadhab Mohanta Regd. Number: 2341013398

Output :

carbon
carbno
caronb
carnbo
carnob

(720 times)

Question: Write a java method to calculate the sum of digits of a given number until the no. is a single digit. The method signature is as follows.
public static int sum-of-Digits(int n).

ex. 9294 = 9+2+9+4 = 24
24 = 2+4 = 6

Ans.
```
import java.util.*;
public class Qs
{
  public static void main (String[] args)
  {
    Scanner sc=new Scanner (System.in);
    System.out.print("Enter a number:");
    int n=sc.nextInt();
    System.out.println("Sum of the digits of "+n+
    "until the no. is a single digit is "+sum-of-
    Digits(n));
    sc.close();
  }
  public static int sum-of-Digits(int n)
  {
    int sum=0;
    while (n>0)
    {
      int r=n%10;
      sum+=r;
      n/=10;
    }
    if (sum>=10)
    {
```

```
            return sum_Of_Digits(sum);
        }
        else
        {
            return (sum);
        }
    }
}
```

Output:

Enter a number = 9294
Sum of digits of 9294 until the number is a single digit is 6.

Question 6: Write a java method, isOdd() that takes an integer i and returns true if and only if i is odd. You method can't use the multiplication, modulus or division operators. The method signature is as follows:
public static boolean isOdd(int n).

Ans.

```
import java.util.*;
public class Q6
{
  public static void main(String[] args)
  {
    Scanner sc= new Scanner(System.in);
    System.out.print("Enter a number:");
    int n= sc.nextInt();
    System.out.println(n+" is odd: "+isOdd(n));
    sc.close();
  }
  public static boolean(int n)
  {
    int x= n&1;
    if (b==1)
    {
      return true;
    }
    else
    {
      return false;
    }
  }
}
```

Output: Enter a number: 37
37 is odd: true

Question 7: Write a java program to find maximum and minimum and how many times they both occur in an array of n elements. Find out the positions where the maximum first occurs and minimum last occurs.

Ans:-

```
import java.util.*;
public class Q7
{
  public static void main (String[] args)
  {
    Scanner sc= new Scanner (System.in);
    System.out.print("Enter no. of elements of
    Array: ");
    int a[]= new int[n];
    System.out.println("Enter elements of array:");
    for (int i=0; i<a.length; i++)
    {
      a[i]= sc.nextInt();
    }
    int min=a[0];
    int max=a[0];
    for (int i=1; i<a.length; i++)
    {
      if (a[i] > max)
        max= a[i];
      if (a[i] < max)
        min= a[i];
    }
```

```
System.out.println("Maximum element of Array is"+max+" and it has occured "+ count(a,max) + "times");
System.out.println("Minimum element of Array is "+min+" and occurs "+count(a,min) +"times");
System.out.println("First occurance of maximum element is at position"+indmax(a,max));
System.out.println("Last occurance of minimum element is at position "+indmin(a,min));
sc.close();
}
public static int count(int a[], int x)
{
int count=0;
for(int i=0; i<a.length; i++)
{
if(a[i]==x)
{
count+=1;
}
}
return count;
}
public static int indmin(int a[], int min)
{
int minc=0;
```

```
        for (int i=0 ; i<a.length ; i++)
        {
            if (min == a[i])
            {
                minc++;
            }
        }
        return (minc+1);
    }
    public static ind max (int a[], int max)
    {
        int maxc=0;
        for (int i=a.length-1 ; i>=0 ; i--)
        {
            maxc
            if (max == a[i])
            {
                maxc=i;
            }
        }
        return (maxc+1);
    }
}
```

Output:

Enter number of elements of Array: 5
Enter elements of array: 12 14 12 14 13.
Maximum element of Array is: 14 and occurs 2 times.
Minimum element of Array is 12 and occurs 2 times.
First occurance of Maximum element is at position 2.
Last occurance of minimum element is at position 3.

Question 8: Write a java program to print M by N array in tabular format. And display sum of the elements of the array.

Ans:

```
import java.util.*;
public class Q8
{
  public static void main(String[] args)
  {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter no. of row:");
    int r = sc.nextInt();
    System.out.println("Enter no. of column:");
    int c = sc.nextInt();
    int a[][] = new int[r][c];
    int sum=0;
    System.out.println("Enter elements of 2D
    array:");
    for(int i=0; i<m; i++)
    {
      for(int j=0; j<n; j++)
      {
        int[i]
        a[i][j] = sc.nextInt();
      }
    }
    System.out.println("Array is:");
    for(int i=0; i<a.length; i++)
```

```
        {
            for (int j=0; j<a[i].length; j++)
            {
                System.out.print(a[i][j]+" ");
            }
            System.out.println();
        }
        for (int i=0; i<a.length
        int sum=0;
        for (int i=0; i<a.length; i++)
        {
            for (int j=0; j<a.length; j++)
            {
                sum+=a[i][j];
            }
        }
        System.out.println("The sum of elements
        of 2D Array is: "+sum);
        sc.close();
    }
}
```

Output:

```
Enter no. of row: 3
Enter no. of column: 3
Enter elements of 2-D array: 1 2 3 2 3 4 3 4 5
Array is:
1 2 3
2 3 4
3 4 5
The sum of elements of 2D Array is 27.
```

Name: Suryamadhab Mohanana Regd. Number: 2341013398

Question 9: Write a method that sums all numbers in the major diagonal in a n * n matrix of double values using following ~~order~~ header:
public static double sumMajorDiagonal(double[][] m)
Write a program that reads 4-by-4 matrix and displays the sum of all its elements on the major diagonal.

Ans.

```
import java.util.*;
public class Q9
{
  public static void main(String[] args)
  {
    Scanner sc=new Scanner(System.in);
    double a[][] = new double[4][4];
    System.out.println("Enter 4 X 4 matrix:");
    for(int i=0; i<4; i++)
    {
      for(int j=0; j<4; j++)
      {
        a[i][j] = sc.nextDouble();
      }
    }
    System.out.println("Sum of elements in the
    major diagonal is " + sumMajorDiagonal(a));
    sc.close();
  }
```

```
public static double sumMajorDiagonal(double m[][])
{
    double sum=0;
    for(int i=0; i<4; i++)
    {
        for(int j=0; j<4; j++)
        {
            if(i==j)
            {
                sum+=a[i][j];
            }
        }
    }
    return sum;
}
}
```

Output:

Enter 4x4 matrix:

1 2 3 4.0
5 6.5 7 8
9 10 11 12
13 14 15 16

Sum of elements in the major diagonal is 34.5.

Question 10: Write a method that returns the sum of all elements in a specified column in a matrix using ~~her~~ following header:
public static double sumColumn(double[][] m;
int columnIndex).
Write a java program that reads 3×4 matrix and display the sum of each column.

Ans.

```
import java.util.*;
public class Q10
{
  public static void main (String[] args)
  {
    Scanner sc=new Scanner (System.in);
    double a[][]=new double [3][4];
    System.out.println("Enter a 3X4 matrix:");
    for(int i=0; i<3; i++)
    {
      for (int j=0; j<4; j++)
      {
        a[i][j]= sc.nextDouble();
      }
    }
    System.out.println("Sum of elements at
    column " + k+1 + "is" + sumColumn(a, r
```

```
    for (int k=0; k<4; k++)
    {
      System.out.println("Sum of elements of
      column "+(k+1)+" is "+sumColumn(a,k));
    }
    sc.close();
  }
  public static double sumColumn(double a[][],
                                 int columnIndex)
  {
    double sum=0;
    for (int i=0; i<4; i++)
    {
      for (int j=0; j<4; j++)
      {
        if (j==columnIndex)
        {
          sum+=m[i][j];
        }
      }
    }
    return sum;
  }
}
```

Output

```
Enter a 3-by-4 matrix
1.5  2  3  4
5.5  6  7  8
9.5  1  3  1
```

Sum of the elements at column 1 is 16.5
Sum of the elements at column 2 is 9.0
Sum of the elements at column 3 is 13.0
Sum of the elements at column 4 is 13.0

20.3.24.