

Programming in Java, Test-2 Set-1 (Each Q carries 5 marks)

1. WAP to count the number of lines in text file. Name of file is input through user.

```
import java.io.*;
import java.util.*;
class S3_1 {
    public static void main(String[] args) {
        int i;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter file name: ");
        String fName = sc.next();
        try ( FileInputStream fin = new FileInputStream(fName)) {
            int countLines=0;
            do{
                i = fin.read();
                if(i=='\n')countLines++;
            }while(i!=-1);
            System.out.println("Number of lines are "+(countLines+1));
        } catch (IOException e) {
            System.out.println("Exception caught");
        }
    }
}
```

2. Write a program in java that takes value of n from user and print the sum of following series:
 $1^2+2^2+3^2+\dots+n^2$. Further, it should generate exception if n is 0 or negative.

```
import java.util.*;
class MyException extends Exception{
    int n;
    MyException(int n){
        this.n =n;
    }
    public String toString(){
        return n+" is a negative number";
    }
}
class S3_2 {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter vaue of n: ");
        int n=sc.nextInt();
        try{
            if(n<=0)throw new MyException(n);
            int sumSeries=0;
            for(int i=1;i<=n;i++){
                sumSeries=i*i+sumSeries;
            }
            System.out.println("Sum of series is: "+sumSeries);
        }catch(MyException e){
            System.out.println(e);
        }
    }
}
```

3. WAP having recursive method sum () to calculate the sum of digits of a number which is passed as a parameter to the method.

```

class S3_3 {
    static int getSumofDigits(int n) {
        if (n < 10) {
            return n;
        } else {
            return getSumofDigits(n / 10) + n % 10;
        }
    }
    public static void main(String[] args) {
        int num = 9234;
        System.out.println("Sum of digits of " + num + " is " + getSumofDigits(num));
    }
}

```

4. Implement method 'multiply' of class Matrix with 3 instance variables (int r, c, arr[][]), with signature: 'Matrix multiply (Matrix m2)'. Assume constructor already used in Matrix class as Matrix(int arr[][]).

```

class Matrix {
    int r;
    int c;
    int arr[][];
    Matrix(int arr[][]) {
        r = arr.length;
        c = arr[0].length;
        this.arr = arr;
    }
    Matrix multiply(Matrix m2) {
        int m3arr[][] = new int[r][m2.c];
        if (c == m2.r) {
            for (int i = 0; i < r; i++) {
                for (int j = 0; j < m2.c; j++) {
                    for (int k = 0; k < c; k++) {
                        m3arr[i][j] = m3arr[i][j] + (arr[i][k] * m2.arr[k][j]);
                    }
                }
            }
            return new Matrix(m3arr);
        } else {
            return this;
        }
    }
}
class S3_4 {
    public static void main(String[] args) {
        int marr1[][] = {{2, 3, 4}, {1, 2, 0}};
        int marr2[][] = {{2, 1}, {3, 2}, {7, 4}};
        Matrix m1 = new Matrix(marr1);
        Matrix m2 = new Matrix(marr2);
        Matrix m3 = m1.multiply(m2);
    }
}

```

Programming in Java, Test-2 Set-2 (Each Q carries 5 marks)

1. A do while loop asks the users to enter first name of the person and his/her height. If the name has more than 15 characters, (Hint: length() function) or if the height is more than 7 feet it should raise an

exception and exit from the loop. If both inputs are OK, then it should convert the height in feet to inches and print the name and height of the person. WAP to implement the loop.

```
import java.util.*;
class myException extends Exception {
    String name;
    float height;
    myException(String name, float height) {
        this.name = name;
        this.height = height;
    }
    public String toString() {
        return "Either " + name + " has more than 15 characters or height " + height + " is more than 7 feet";
    }
}
class S4_1 {
    public static void main(String[] args) {
        String name = "";
        float height = 0;
        Scanner sc = new Scanner(System.in);
        try {
            do {
                System.out.println("Enter First Name");
                name = sc.next();
                System.out.println("Enter height in feet");
                height = sc.nextFloat();
                if(name.length()>15 || height>7)throw new myException(name,height);
                System.out.println(name+"'s height: "+(height*12)+ " inches.");
            } while (true);
        } catch (myException e) {
            System.out.println("Exception: "+e);
        }
    }
}
```

2. Write a program having method 'isPrime' to find whether the given number is prime or not.

```
import java.util.*;
class S4_2 {
    static boolean isPrime(int n) {
        for(int i=2;i<=n/2;i++){
            if(n%i==0)return false;
        }
        return true;
    }
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter number");
        int n=sc.nextInt();
        if(isPrime(n))
            System.out.println(n + " is Prime");
        else
            System.out.println(n + " is not Prime");
    }
}
```

3. WAP to display number of characters in a file. Name of file is taken through command Line argument.

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

```

class S4_3 {
    public static void main(String[] args) {
        int i;
        String fName = "MyFile1.txt";
        try ( FileInputStream fin = new FileInputStream(args[0])) {
            int countchars=0;
            do{
                i = fin.read();
                countchars++;
            }while(i!=-1);
            System.out.println("Total characters: "+(countchars));
        } catch (IOException e) {
            System.out.println("Exception caught");
        }
    }
}

```

4. Create a class TwoDim which contains x and y coordinates as int. Define the parameterized constructor and use toString method to print the co-ordinates. Now reuse this class in ThreeDim adding a new dimension as z of type int. Define the constructor for the subclass and override the method toString() in the subclass. Implement main() to show dynamic method dispatch.

```

class TwoDim {
    int x;
    int y;
    TwoDim(int x, int y) {
        this.x = x;
        this.y = y;
    }
    public String toString() {
        return "Coordinates: " + x + ", " + y;
    }
}

class ThreeDim extends TwoDim {
    int z;
    ThreeDim(int x, int y, int z) {
        super(x, y);
        this.z = z;
    }
    public String toString() {
        return super.toString() + ", " + z;
    }
}

class S4_4 {
    public static void main(String[] args) {
        ThreeDim t1=new ThreeDim(2,3,4);
        TwoDim d=t1;
        System.out.println(d);
    }
}

```