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

1. WAP to generate the following output for a given number n=3 say

```
0
1 2
3 4 5
```

```
import java.util.*;
class S1_1 {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter number of lines: ");
      int lines = sc.nextInt();
      int sNum = 0;
      for (int i = 1; i <= lines; i++) {
         for (int j = 1; j <= i; j++) {
            System.out.print(sNum++ + " ");
         }
         System.out.println("");
      }
   }
}
```

2. WAP to read file A.txt and copy the text in B.txt file after removing the vowels.

```
import java.io.*;
class S1_2 {
   public static void main(String[] args) {
      int i;
      //System.out.println("Working Directory = " + System.getProperty("user.dir"));
      try ( FileInputStream fin = new FileInputStream("A.txt");
            FileOutputStream fout = new FileOutputStream("B.txt")) {
         do {
            i = fin.read();
            if (!(i == 'a' || i == 'e' || i == 'i' || i == 'o' || i == 'u')) {
               fout.write(i);
            }
         } while (i != -1);
      } catch (IOException e) {
         System.out.println("IO Exception");
      }
   }
}
```

3. Write a method to read name of the student through command Line. If first letter of the name is not a capital letter then throw an exception.

```
class MyException extends Exception {
}
class S1_3 {
   public static void main(String[] args) {
      String s = "computer";//args[0]
      try {
         char ch=s.charAt(0);
         if(!(ch>='A' && ch<='Z'))
            throw new MyException();
      } catch (MyException e) {
         System.out.println("First Letter not Capital");
      }
   }
}
```

Or

```
class MyException extends Exception {
   String s;
   MyException(String s){
      this.s=s;
   }
   public String toString(){
      return "First letter of '"+s+"' is not capital letter";
   }
}
class S1_3b {
   public static void main(String[] args) {
      String s = "computer";//args[0]
      try {
         char ch=s.charAt(0);
         if(!(ch>='A' && ch<='Z'))
            throw new MyException(s);
      } catch (MyException e) {
         System.out.println(e);
      }
   }
}
```

4. Design a Class Rational to store numbers in form of a/b & which also add two rational numbers.

```
class Rational {
   int n;
   int d;
   public Rational(int n, int d) {
```

```
        this.n = n;
        this.d = d;
    }
    public String toString() {
        return n + "/" + d;
    }
    Rational add(Rational r2) {
        int fn = (n * r2.d) + (d * r2.n);
        int fd = d * r2.d;
        return new Rational(fn, fd);
    }
}
class S1_4 {
    public static void main(String[] args) {
        Rational r1=new Rational(1,2);
        Rational r2=new Rational(1,4);
        System.out.println(r1.add(r2));
    }
}
```

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

1. WAP to evaluate S=1!-2!+3!-4!+…(n terms)

```
import java.util.*;
class S2_1 {
    static int factorial(int n){
        if(n==0)
            return 1;
        else
            return factorial(n-1)*n;
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter value of n: ");
        int n = sc.nextInt();
        int sum1 = 0;
        for (int i = 1; i <= n; i++) {
            if(i%2==0)
                sum1=sum1-factorial(i);
            else
                sum1=sum1+factorial(i);
        }
        System.out.println("Sum of series is "+sum1);
    }
}
```

2. Write a program that accepts the names of two text files through command line, reads their contents and prints the contents of first file followed by the contents of second file on the screen.

```
import java.io.*;
class S2_2 {
   public static void main(String[] args) {
      int i;
      //System.out.println("Working Directory = " + System.getProperty("user.dir"));
      //try ( FileInputStream fin1 = new FileInputStream("A.txt"); //args[0]
            //FileInputStream fin2 = new FileInputStream("B.txt")) {//args[1]
         try ( BufferedReader fin1 = new BufferedReader(new FileReader("A.txt")); //args[0]
            BufferedReader fin2 = new BufferedReader(new FileReader("B.txt"))) {//args[1]
         do {
            i = fin1.read();
            System.out.print((char) i);
         } while (i != -1);
         do {
            i = fin2.read();
            System.out.print((char) i);
         } while (i != -1);
      } catch (IOException e) {
         System.out.println("IO Exception");
      }
   }
}
```

3. Create a user defined exception class MyException and use this class to signal an error condition if the number is negative. Write a program to compute the square root of a number using user defined method MySqrt() which raises exception of type MyException for negative number.

```
class MyException extends Exception {
   float i;
   MyException(float i){
      this.i=i;
   }
   public String toString(){
      return "Number '"+i+"' is negative";
   }
}
class S2_3 {
   static void MySqrt(float f){
      try{
         if(f>0)
            System.out.println(Math.sqrt(f));
         else
            throw new MyException(f);
      }catch(MyException e){
         System.out.println(e);
      }
```

```
    }
    public static void main(String[] args) {
        MySqrt(-5);
    }
}
```

4. WAP to read n elements and display the position of the largest element in the array.

```
import java.util.*;
class S2_4 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter number of elements in array: ");
        int num = sc.nextInt();
        System.out.println("Enter positive numbers");
        int a[]=new int[num];
        int mNum =-1;
        int mIndex=-1;
        for(int i=0;i<num;i++){
            a[i] = sc.nextInt();
            if(a[i]>mNum){
                mNum=a[i];
                mIndex=i;
            }
        }
        System.out.println("Max number is "+mNum+" at index "+mIndex);
    }
}
```