1(A). Finding Prime numbers between 1 and n

```
//JAVA PROGRAM TO PRINT PRIME NUMBERS FROM 1 TO N

import java.util.*;

class Primegen

{

   public static void main(String args[])throws Exception

   {

      int n,i,j,fc;

      Scanner sc = new Scanner(System.in);

      System.out.print("\n\n\t ENTER THE VALUE FOR N....");

      n=sc.nextInt();

      System.out.print("\n\n\t THE PRIME NUMBERS BETWEEN 1 AND " + n +" ARE...");

      for(i=1;i<=n;i=i+1)

      {

         for(j=1,fc=0;j<=i;j=j+1)

         {

            if(i%j==0)

            {

               fc=fc+1;

            }

         }

         if(fc==2)

         {

            System.out.print(" " +i);

         }

      }

   }

}
```

1(B). Printing all the real solutions of the Quadratic equation

```
// JAVA PROGRAM TO PRINT THE ROOTS OF A QUADRATIC EQUATION

import java.util.*;

class Quadratic {

    public static void main(String args[]) throws Exception {

        int a, b, c;

        double d, r, r1, r2, p, q, z;

        Scanner sc = new Scanner(System.in);

        System.out.print("\n\n\t ENTER THE COEFFICIENT OF X*X...");

        a = sc.nextInt();

        System.out.print("\n\n\t ENTER THE COEFFICIENT OF X...");

        b = sc.nextInt();

        System.out.print("\n\n\t ENTER THE CONSTANT TERM...");

        c = sc.nextInt();

        d = (b * b) - (4 * a * c);

        if (d == 0) {

            r = -(b / (2 * a));

            System.out.print("\n\n\t THE ROOTS ARE REAL AND EQUAL");

            System.out.print("\n\n\t THE ROOT IS..." + r);

        } else if (d > 0) {

            p = -(b / (2 * a));

            q = (Math.sqrt(d)) / (2 * a);

            r1 = p + q;

            r2 = p - q;

            System.out.print("\n\n\t THE ROOTS ARE REAL AND DISTINCT");

            System.out.print("\n\n\t THE ROOTS ARE..." + r1 + "AND" + r2);

        } else {

            z = Math.abs(d);
```

```
            p = -(b / (2 * a));
            q = (Math.sqrt(z)) / (2 * a);
            System.out.print("\n\n\t THE ROOTS ARE IMAGINARY");
            System.out.print("\n\n\t THE FIRST ROOT IS..." + p + "+" + q + "i");
            System.out.print("\n\n\t THE SECOND ROOT IS..." + p + "-" + q + "i");
        }
    }
}
```

2(A).To find the factorial of a given number

```
//JAVA PROGRAM TO PRINT THE FACTORIAL OF A GIVEN NUMBER

import java.util.*;

class Factorial

   {

   public static void main(String args[]) throws Exception

   {

      Scanner sc = new Scanner(System.in);

      int n, i, fact;

      System.out.print("Enter the number: ");

      n = sc.nextInt();

      fact = 1;

      for (i = 1; i <= n; i = i + 1)

      {

         fact = fact * i;

      }

      System.out.println("Factorial of " + n + " is: " + fact);

   }

}
```

2(B).To find whether given number is prime or not

```
//JAVA PROGRAM TO PRINT PRIME NUMBERS FROM 1 TO N

import java.util.*;

class Primechk

{

   public static void main(String args[]) throws Exception

   {

      int n,i,fc;

      Scanner sc = new Scanner(System.in);

      System.out.print("\n\n\t ENTER NUMBER....");

      n = sc.nextInt();

      fc=0;

      for (i = 1; i <= n; i++)

      {

         if (n%i == 0)

         {

            fc = fc + 1;

         }

      }

      if (fc == 2)

      {

         System.out.print("\n\n\t"+n+" IS PRIME NUMBER");

      }

      else

      {

         System.out.print("\n\n\t"+n+" IS NOT A PRIME NUMBER");

      }

   }

}
```

2(C).To print N terms of Fibonacci series

```
//JAVA PROGRAM TO PRINT N TERMS OF FIBINACCI SERIES

import java.util.*;

class Fibgen

{

   public static void main(String args[]) throws Exception

   {

      int t1, t2, t3, n, count;

      Scanner sc = new Scanner(System.in);

      System.out.print("Enter the number of terms in series....");

      n = sc.nextInt();

      t1 = 0;

      t2 = 1;

      if (n == 1)

      {

         System.out.print(t1);

      }

      else if (n == 2)

      {

         System.out.print(t2);

      }

      else

      {

         System.out.print(t1 + " " + t2);

         t3 = t1 + t2;

         System.out.print(" " +t3+ " ");

         count = 3;

         while (count <= n)
```

```
            {
               t1 = t2;
               t2 = t3;
               t3 = t1 + t2;
               count = count + 1;
               System.out.print(" "+ t3 +" ");
            }
         }
      }
   }
```

3(A). TO FIND THE SUM OF THE INDIVIDUAL DIGITS OF A NUMBER

```
//JAVA PROGRAM TO FIND THE SUM OF THE INDIVIDUAL DIGITS OF A NUMBER
import java.util.*;
class Inddig
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      int n, r, sum;
      System.out.println("Enter a number:");
      n = sc.nextInt();
      sum = 0;
      while (n > 0)
      {
         r = n % 10;
         sum = sum + r;
         n = n / 10;
      }
      System.out.print("Sum of digits is: " + sum);
   }
}
```

3(B).Arithmetic calculator using switch case menu

```
// Java program to perform arithmetic operations using switch case.

import java.util.*;

class Simpcalc

{

   public static void main(String[] args) throws Exception

   {

      int ch;

      double n1, n2, result;

      Scanner sc = new Scanner(System.in);

      System.out.print("Enter First number: ");

      n1 = sc.nextDouble();

      System.out.print("Enter Second number: ");

      n2 = sc.nextDouble();


System.out.print("\n1.Addition\n2.Subtraction\n3.Multiplication\n4.Division\n5.Modulus\n
6.Exit\n");

      System.out.println("Enter your choice: \n");

      ch = sc.nextInt();

      switch (ch)

      {

      case 1:

         result = n1 + n2;

         System.out.print("Addition of two numbers: " + result);

         break;

      case 2:

         result = n1 - n2;

         System.out.print("Subtraction of two numbers: " + result);

         break;
```

```
        case 3:
           result = n1 * n2;
           System.out.print("Multiplication of two numbers: " + result);
           break;
        case 4:
           result = n1 / n2;


           System.out.print("Division of two numbers: " + result);
           break;
        case 5:
           result = n1 % n2;
           System.out.print("Modulus of two numbers: " + result);
           break;
        case 6:
           System.exit(0);
           break;
        default:
           System.out.print("Invalid choice");
           break;
        }
     }
  }
```

4(A). Product of two matrices

```
// java program for multiplication of two matrices

import java.util.*;

class Matrixmul

{

   public static void main(String args[]) throws Exception

   {

      int m, n, p, q, i, j, k, a[][], b[][], c[][];

      Scanner sc = new Scanner(System.in);

      System.out.print("\n\n\t ENTER THE NUMBER OF ROWS IN THE FIRST MATRIX...");

      m = sc.nextInt();

      System.out.print("\n\n\t ENTER THE NUMBER OF COLUMNS IN THE FIRST MATRIX...");

      n = sc.nextInt();

      System.out.print("\n\n\t ENTER THE NUMBER OF ROWS IN THE SECOND MATRIX...");

      p = sc.nextInt();

      System.out.print("\n\n\t ENTER THE NUMBER OF COLUMNS IN THE SECOND
MATRIX...");

      q = sc.nextInt();

      a = new int[m][n];

      b = new int[p][q];

      c = new int[m][q];

      if (n == p)

      {

         System.out.print("\n\n\t ENTER THE ELEMENTS OF THE FIRST MATRIX...");

         for (i = 0; i < m; i = i + 1)

         {

            for (j = 0; j < n; j = j + 1)

            {

               System.out.print("\n\n\t ENTER THE ELEMENT-a[" + i + "][" + j + "]...");
```

```
        a[i][j] = sc.nextInt();
    }
}
System.out.print("\n\n\t ENTER THE ELEMENTS OF THE SECOND MATRIX...");
for (i = 0; i < p; i = i + 1)
{
    for (j = 0; j < q; j = j + 1)
    {
        System.out.print("\n\n\t ENTER THE ELEMENT-b[" + i + "][" + j + "]...");
        b[i][j] = sc.nextInt();
    }
}
System.out.print("\n\n\t THE ELEMENTS OF THE FIRST MATRIX ARE...\n");
for (i = 0; i < m; i = i + 1)
{
    for (j = 0; j < n; j = j + 1)
    {
        System.out.print(" " + a[i][j]);
    }
    System.out.print("\n");
}
System.out.print("\n\n\t THE ELEMENTS OF THE SECOND MATRIX ARE...\n");
for (i = 0; i < p; i = i + 1)
{
    for (j = 0; j < n; j = j + 1)
    {
        System.out.print(" " + b[i][j]);
    }
    System.out.print("\n");
```

```
    }
    for (i = 0; i < m; i = i + 1)
    {
       for (j = 0; j < q; j = j + 1)
       {
          c[i][j] = 0;
       }
    }
    for (i = 0; i < m; i = i + 1)
    {
       for (j = 0; j < q; j = j + 1)
       {
          for (k = 0; k < n; k = k + 1)
          {
             c[i][j] = c[i][j] + (a[i][k] * b[k][j]);
          }
       }
    }
    System.out.print("\n\n\t THE PRODUCT OF TWO MATRICES IS...\n");
    for (i = 0; i < m; i = i + 1)
    {
       for (j = 0; j < q; j = j + 1)
       {
          System.out.print(" " + c[i][j]);
       }
       System.out.print("\n");
    }
 }
 else
```

```
        {
            System.out.print("\n\n\t MATRIX MULTIPLICATION IS NOT POSSIBLE DUE TO
DIMENSIONS");
        }
    }
}
```

4(B). Method Overloading

```
//JAVA PROGRAM TO IMPLEMENT METHOD OVERLOADING

import java.util.*;

class Overload

{

   public static void main(String args[]) throws Exception

   {

      int s, x, ch;

      float ln, br, y;

      double a, b, c, z;

      Scanner sc = new Scanner(System.in);

      do

      {

         System.out.print("\n\n\t-----MENU-----");

         System.out.print("\n\n\t 1...AREA OF SQUARE");

         System.out.print("\n\n\t 2...AREA OF RECTANGLE");

         System.out.print("\n\n\t 3...AREA OF TRIANGLE");

         System.out.print("\n\n\t 4...EXIT");

         System.out.print("\n\n\t ENTER YOUR CHOICE...");

         ch = sc.nextInt();

         switch (ch)

         {

         case 1:

            System.out.print("\n\n\t ENTER THE SIDE OF THE SQUARE...");

            s = sc.nextInt();

            x = area(s);

            System.out.print("\n\n\t THE AREA OF SQUARE IS..." + x);

            break;
```

```
        case 2:
           System.out.print("\n\n\t ENTER THE LENGTH OF THE RECTANGLE...");
           ln = sc.nextFloat();
           System.out.print("\n\n\t ENTER THE BREADTH OF THE RECTANGLE...");
           br = sc.nextFloat();

           y = area(ln, br);
           System.out.print("\n\n\t THE AREA OF RECTANGLE IS..." + y);
           break;
        case 3:
           System.out.print("\n\n\t ENTER THE LENGTH OF SIDE-1...");
           a = sc.nextDouble();
           System.out.print("\n\n\t ENTER THE LENGTH OF SIDE-2...");
           b = sc.nextDouble();
           System.out.print("\n\n\t ENTER THE LENGTH OF SIDE-3...");
           c = sc.nextDouble();
           z = area(a, b, c);
           System.out.print("\n\n\t THE AREA OF TRIANGLE IS..." + z);
           break;
        case 4:
           System.exit(0);
        default:
           System.out.print("\n\n\t INVALID CHOICE");
        }
    } while (ch >= 1 && ch <= 3);
}
public static int area(int s)
{
    int p;
```

```
        p = s * s;
        return (p);
    }
    public static float area(float ln, float br)
    {
        float p;
        p = ln * br;
        return (p);
    }
    public static double area(double a, double b, double c)
    {
        double s, p;
        s = (a + b + c) / 2;
        p = Math.sqrt(s * (s - a) * (s - b) * (s - c));
        return (p);
    }
}
```

4(C). Method Overriding

```
//JAVA PROGRAM TO IMPLEMENT METHOD OVERRIDING

import java.util.*;

class Person

{

   public void display()

   {

      System.out.print("\n\n\t PERSON");

   }

}

class Doctor extends Person

{

   public void display()

   {

      System.out.print("\n\n\t DOCTOR");

   }

}

class Override

{

   public static void main(String args[])

   {

      Person p = new Person();

      Doctor d = new Doctor();

      p.display();

      d.display();

   }

}
```

5(A). Creating Student Class

```
//JAVA PROGRAM TO CREATE A STUDENT CLASS

import java.util.*;

class Students

{

   String rollno;

   String name;

   String branch;

   double phoneno;

   public void accept_det()

   {

      Scanner sc = new Scanner(System.in);

      System.out.print("\n\n\tEnter the rollno: ");

      rollno = sc.next();

      System.out.print("\n\n\tEnter the name: ");

      name = sc.next();

      System.out.print("\n\n\tEnter the branch: ");

      branch = sc.next();

      System.out.print("\n\n\tEnter the phoneno: ");

      phoneno = sc.nextDouble();

   }

   public void display_det()

   {

      System.out.print("\n\n\tThe entered roll no is: " + rollno);

      System.out.print("\n\n\tThe entered name is: " + name);

      System.out.print("\n\n\tThe entered branch is: " + branch);

      System.out.print("\n\n\tThe entered phoneno is: " + phoneno +"\n\n");

   }
```

```
}
class Demo
{
   public static void main(String args[]) throws Exception
   {
      Scanner sc = new Scanner(System.in);
      Students s[];
      int n;
      System.out.print("\n\n\tEnter the number of students: ");
      n = sc.nextInt();
      s = new Students[n];
      for (int i = 0; i < n; i++)
      {
         s[i] = new Students();
      }
      for (int i = 0; i < n; i++)
      {
         s[i].accept_det();
      }
      for (int i = 0; i<n; i++)
      {
         s[i].display_det();
      }
   }
}
```

5(B). Use of Inheritance, using Final

```
//JAVA PROGRAM TO ILLUSTRATE THE USAGE OF FINAL KEYWORD WITH A CLASS

final class Simple1

{

   public void display()

   {

      System.out.print("\n\n\t BASE CLASS");

   }

}

class Simple2 extends Simple1

{

   public void display()

   {

      System.out.print("\n\n\t DERIVED CLASS");

   }

}

class Finaldemo3

{

   public static void main(String args[])

   {

      Simple2 s = new Simple2();

      s.display();

   }

}
```

5(C). Abstract Class

```
//JAVA PROGRAM TO CREATE AN ABSTRACT CLASS AND EXTEND CLASSES FROM IT

import java.util.*;

abstract class Shape

{

   int x, y;

   abstract public void area();

}

class Rectangle extends Shape

{

   public void accept() throws Exception

   {

      Scanner sc = new Scanner(System.in);

      System.out.print("\n\n\t ENTER THE LENGTH OF THE RECTANGLE... ");

      x = sc.nextInt();

      System.out.print("\n\n\t ENTER THE BREADTH OF THE RECTANGLE... ");

      y = sc.nextInt();

   }

   public void area()

   {

      System.out.print("\n\n\t THE AREA OF THE RECTANGLE IS... " + (x * y));

   }

}

class Triangle extends Shape

{

   public void accept() throws Exception

   {

      Scanner sc = new Scanner(System.in);
```

```
        System.out.print("\n\n\t ENTER THE BASE OF THE TRIANGLE... ");
        x = sc.nextInt();
        System.out.print("\n\n\t ENTER THE HEIGHT OF THE TRIANGLE... ");
        y = sc.nextInt();
    }
    public void area()
    {
        System.out.print("\n\n\t THE AREA OF THE TRIANGLE IS... " + (0.5 * x * y));
    }
}
class Circle extends Shape
{
    public void accept() throws Exception
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("\n\n\t ENTER THE RADIUS OF THE CIRCLE... ");
        x = sc.nextInt();
    }
    public void area()
    {
        System.out.print("\n\n\t THE AREA OF CIRCLE IS... " + (3.14 * x * x));
        }
    }
    class Shapedemo
    {
        public static void main(String args[]) throws Exception
        {
            Rectangle r = new Rectangle();
            Triangle t = new Triangle();
```

```
        Circle c = new Circle();
        r.accept();
        r.area();
        t.accept();
        t.area();
        c.accept();
        c.area();
    }
}
```

6(A). Creating a User defined Exception

```
//JAVA PROGRAM TO CREATE A USER DEFINED EXCEPTION

import java.util.*;

class Simple extends Exception

{

   Simple(String s)

   {

      super(s);

   }

}

class Simpledemo

{

   public static void main(String args[])

   {

      int n;

      Scanner sc = new Scanner(System.in);

      System.out.print("\n\n\t ENTER THE AGE OF THE PERSON...");

      n = sc.nextInt();

      if (n < 18)

      {

         try

         {

            throw new Simple("AGE SHOULD BE GREATER THAN 18");

         } catch (Exception e) {

            System.out.print(e);

         }

      }

      else
```

```
        {
            System.out.print("\n\n\t ELIGIBLE FOR VOTING");
        }
    }
}
```

6(B). Splitting a File into n-parts

```
//JAVA PROGRAM FOR SPLITTING A FILE
import java.io.*;
import java.util.*;
class Split
{
   public static void main(String args[]) throws Exception
   {
      String f, s;
      Scanner sc = new Scanner(System.in);
      System.out.print("\n\n\t ENTER THE FILE NAME... ");
      f = sc.next();
      BufferedReader br1 = new BufferedReader(new FileReader(f));
      int lc = 0;
      System.out.print("\n\n\t THE CONTENTS OF THE FILE ARE: ");
      while ((s = br1.readLine()) != null)
      {
         System.out.print("\n\n\t" + s);
         System.out.print("\n");
         lc++;
      }
      System.out.println("\n\n\t NUMBER OF LINES IN THE FILE IS: " + lc);
      int nof;
      System.out.print("\n\n\t ENTER THE NUMBER OF FILES: ");
      nof = sc.nextInt();
      System.out.println("\n\n\tNUMBER OF FILES TO BE GENERATED IS: " + nof);
      br1.close();
      BufferedReader br2 = new BufferedReader(new FileReader(f));
```

```
String sl;
int p = lc / nof;
for (int j = 1; j <= nof; j++)
{
   FileWriter fw = new FileWriter("F:/File" + j + ".txt");
   BufferedWriter bw = new BufferedWriter(fw);
   for (int i = 1; i <= p; i++)
   {
      sl = br2.readLine();
      if (sl != null)
      {
         bw.write(sl);
         if (i != p)
         {
            bw.newLine();
         }
      }
   }
   bw.close();
}
System.out.print("\n\n\t THE CONTENTS OF FILE-1 ARE: \n");
BufferedReader br3 = new BufferedReader(new FileReader("F:/File1.txt"));
while ((s = br3.readLine()) != null)
{
   System.out.print("\n\n\t" + s);
   System.out.print("\n");
}
System.out.print("\n\n\t THE CONTENTS OF FILE-2 ARE: \n");
BufferedReader br4 = new BufferedReader(new FileReader("F:/File2.txt"));
```

```
        while ((s = br4.readLine()) != null)
        {
           System.out.print("\n\n\t" + s);
           System.out.print("\n");
        }
     }
  }
```

7(A). Displaying the information about the given File

```
//JAVA PROGRAM TO DISPLAY THE INFORMATION ABOUT A FILE
import java.util.*;
import java.io.*;
class Fileinfo
{
   public static void main(String args[]) throws Exception
   {
      String s;
      Scanner sc = new Scanner(System.in);
      System.out.print("\n\n\t ENTER THE FILE NAME ....");
      s = sc.next();
      File f1 = new File(s);
      if (f1.exists())
      {
         System.out.print("\n\n\t FILE EXISTS");
      } else {
         System.out.print("\n\n\t FILE DOES NOT EXIST");
      }
      if (f1.canRead())
      {
         System.out.print("\n\n\t FILE IS READABLE");
      } else {
         System.out.print("\n\n\tFILE IS NOT READABLE");
      }
      if (f1.canWrite())
      {
         System.out.print("\n\n\t FILE IS WRITEABLE");
```

```
        } else {
            System.out.print("\n\n\t FILE IS NOT WRITABLE");
        }
        System.out.print("\n\n\t THE LENGTH OF THE FILE IS.... " + f1.length());
    }
}
```

7(B). Counting the no.of characters,words and lines in a file

```
//JAVA PROGRAM TO DISPLAY THE CONTENTS OF A FILE ALONG WITH LINE NUMBER

import java.util.*;

import java.io.*;

class Lineread

{

   public static void main(String args[]) throws Exception

   {

      String s, l;

      Scanner sc = new Scanner(System.in);

      System.out.print("\n\n\t ENTER THE FILE NAME...");

      s = sc.next();

      LineNumberReader lnr = new LineNumberReader(new FileReader(s));

      while ((l = lnr.readLine()) != null)

      {

         System.out.print("\n\n\t LINE-" + lnr.getLineNumber() + ":" + l);

      }

   }

}
```

8.Creating a zero division error Exception Or

Division Exception

```
//JAVA PROGRAM FOR HANDLING EXCEPTIONS
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
import javax.swing.*;
/* <applet code="Division" width="500" height="500">
</applet>
*/
public class Division extends Applet implements ActionListener
{
   Label l1, l2, l3;
   TextField tf1, tf2, tf3;
   Button b;
   public void init()
   {
      l1 = new Label("ENTER THE FIRST NUMBER");
      l2 = new Label("ENTER THE SECOND NUMBER");
      l3 = new Label("RATIO OFTWO NUMBERS IS");
      tf1 = new TextField();
      tf2 = new TextField();
      tf3 = new TextField();
      b = new Button("DIVIDE");
      add(l1);
      add(tf1);
      add(l2);
      add(tf2);
```

```
        add(l3);
        add(tf3);
        add(b);
        b.addActionListener(this);
    }
    public void actionPerformed(ActionEvent ae)
    {
        String s1 = tf1.getText();
        String s2 = tf2.getText();
        int a = Integer.parseInt(s1);
        int b = Integer.parseInt(s2);
        int c = 0;
        try
        {
            c = a / b;
        }
        catch (Exception e)
        {
            JOptionPane.showMessageDialog(this, "ARITHMETIC EXCEPTION ","EXCEPTION
",JOptionPane.ERROR_MESSAGE);
        }
            String z = Integer.toString(c);
            tf3.setText(z);
        }
    }
```