**Weak1**

1.Try debug step by step with small program of about 10 to 15 lines which contains at least one if else condition and a for loop. Aim:To write a java program that uses recursive procedure to print the nth value in the Fibonacci sequence.

Program:

```
class Demo
{
int fib(int n)
{
if(n==1)
return (1);
else if(n==2)
return (1);
else
return (fib(n-1)+fib(n-2));
}
}
class Fib1
{
public static void main(String args[])
{
Demo ob=new Demo();
System.out.println("The 10th fibonacci element is " + ob.fib(10));
}
}
```

Input/output Processing:

The 10th Fibonacci element is 55=

2.Write a java program that prints all real solutions to the quadratic equation ax2+bx+c=0. Read in a, b, c and use the quadratic formula.

Aim:To write a java program that prints all real solutions to quadratic equation ax2+bx+c= 0

Program:

```
class Quadratic
{
public static void main(String args[])
{
double a=5,b=10,c=2,r1,r2,d;
d=(b*b)-(4*a*c);
if(d==0)
{
System.out.println("Roots are real and equal");
System.out.println("The roots are : " + (-b/(2*a) ) );
}
else if(d<0)
System.out.println("The roots are imaginary");
```

```
else if(d>0)
{
r1=-b+(Math.sqrt(d))/(2*a);
r2=-b-(Math.sqrt(d))/(2*a);
System.out.println("root1=" + r1);
System.out.println("root2=" + r2);
}
}
}
```

Input/output Processing:
Root1 = -9.225403330758517
Root2 = -10.774596669241483

3.The Fibonacci sequence is defined by the following rule. The first two values in the sequence are 1 and 1. Every subsequent value is the sum of the two values preceding it. Write a java program that uses both recursive and non-recursive functions

Aim:To write a java program that prompts the user for an integer and then prints out all prime numbers up to that integer.
Program:

```
import java.io.*;
class Prime
{
public static void main(String args[])throws IOException
{
int i,j,a=0,n;
BufferedReader br= new BufferedReader(new InputStreamReader (System.in) );
System.out.println("Enter range : " );
n = Integer.parseInt(br.readLine() );
for(i=2;i<=n;i++)
{
a=0;
for(j=2;j<i;j++)
{
if(i%j==0)
a=1;
}
if(a==0)
System.out.print(i + "\t");
}
}
}
```

Input/output Processing:
Enter range: 20
2
3
5
7

11
13
17

WEEK 2

1.Write a java program to multiply two given matrices.

Aim:To write a java program to multiply two given matrices.
Program:

```
import java.io.*;
class MatMul
{
public static void main(String args[])throws IOException
{
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter the row and columns for matrix A:");
int r1=Integer.parseInt(br.readLine());
int c1=Integer.parseInt(br.readLine());
System.out.println("Enter the row and columns for matrix B:");
int r2=Integer.parseInt(br.readLine());
int c2=Integer.parseInt(br.readLine());
int i, j, k;
int a[ ][ ]=new int[r1][c1];
int b[ ][ ]=new int[r2][c2];
int c[ ][ ]=new int[r1][c2];
if(c1!=r2)
System.out.println("Matrix multiplication is not possible");
else
{
System.out.println("Enter the elements for matrixA:");
for(i=0;i<r1;i++)
{
for(j=0;j<c1;j++)
{
a[i][j]=Integer.parseInt(br.readLine());
}
}
System.out.println("Enter the elements for matrix B:");
for(i=0;i<r2;i++)
{
for(j=0;j<c2;j++)
{
b[i][j]=Integer.parseInt(br.readLine());
}
}
for(i=0;i<r1;i++)
{
```

```
for(j=0;j<c2;j++)
{
c[i][j]=0;
for(k=0;k<c1;k++)
{
c[i][j]=c[i][j]+a[i][k]*b[k][j];
}
}
}
}
for(i=0;i<r1;i++)
{
System.out.println();
for(j=0;j<c2;j++)
{
System.out.print(c[i][j]+"\t");
}
}
}
}
```

Input/output Processing:

```
Enter the row and columns for matrix A:
2
2
Enter the row and columns for matrix B:
2
2
Enter the elements for matrix A:
1
2
3
4
Enter the elements for matrix B:
2
3
4
5
Resultant matrix is:
10 13
22
```

2.Write a java program to implement method overloading and constructors overloading.

Aim: To write a java program to illustrate method overloading.

Program:

```
import java.io.*;
class Me
{
```

```
int sum(int a,int b)
{
return (a+b);
}
int sum(int a,int b,int c)
{
return (a+b+c);
}
float sum(float a,float b)
{
return (a+b);
}
float sum(int a,float f1)
{
return (a+f1);
}
}
class MeOver
{
public static void main(String args[ ]) throws IOException
{
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter three int values:");
int a=Integer.parseInt(br.readLine());
int b=Integer.parseInt(br.readLine());
int c=Integer.parseInt(br.readLine());
System.out.println("Enter the two float values:");
float f1=Float.parseFloat(br.readLine());
float f2=Float.parseFloat(br.readLine());
Me ob=new Me();
System.out.println("The sum of two integer is:" + (ob.sum(a,b)) );
System.out.println("The sum of three integer is:" + (ob.sum(a,b,c)) );
System.out.println("The sum of two float values is:" + (ob.sum(f1,f2)) );
System.out.println("The sum of one int and one float value is:" + (ob.sum(a,f2)));
}
}
```

Input/output Processing:
Enter three int values: 3 2 1
Enter the two float values: 1.5 2.5
The sum of two integers is: 5
The sum of three integers is: 6
The sum of two float values is: 4.0
The sum of one int and one float value is: 5.5

3.Write a java program to implement method overriding.

Aim: To write a Java program to implement
Method overriding

```
class Vehicle{
  void run(){System.out.println("Vehicle is running");}
}
//Creating a child class
class Bike extends Vehicle{
  public static void main(String args[]){
  //creating an instance of child class
  Bike obj = new Bike();
  //calling the method with child class instance
  obj.run();
  }
}
```

WEEK 3

1. Write a java program to check whether a given string is palindrome.

Aim: To write a java program that checks whether a given string is palindrome or not.
Program:

```
import java.io.*;
class Palindrome
{
public static void main(String args[]) throws IOException
{
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
System.out.print("Enter the string : ");
String s1=br.readLine();
StringBuffer sb = new StringBuffer( s1 );
sb.reverse();
String s2 = sb.toString();
System.out.println(“The reversed String is : “ + s2);
if( s1.equals(s2) )
System.out.println("Given String is palindrome");
else
System.out.println("Given String is not a palindrome");
}
}
```

Input / Output Processing:

```
Enter the string: Ramesh
The reversed String is: hsemaR
Given String is not a palindrome
Enter the string: Madam
The reversed String is: madaM
Given String is palindrome
```

2. Write a java program for sorting a given list of names in ascending order.

Aim: To write a java program to sort a list of names in ascending order.

Program:

```
import java.io.*;
class NameSort
{
public static void main(String args[ ]) throws IOException
{
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter how many names you want");
int n=Integer.parseInt(br.readLine() );
String a[ ]=new String[n];
int i,j;
System.out.println("Enter " + n + " names : ");
for(i=0;i<n;i++)
a[i]=br.readLine();
for(i=0;i<n;i++)
{
for(j=i+1;j<n ;j++)
{
if(a[i].compareTo(a[j])>0)
{
String temp=a[i];
a[i]=a[j];
a[j]=temp;
}
}
}
System.out.println("After sorting the names are");
for(i=0;i<n;i++)
System.out.println(a[i]);
}
}
```

Input / Output Processing:

```
Enter how many names you want: 5
Ramesh
Ravi
Rakesh
Ramu
Ramana
After sorting the names are:
Rakesh
Ramana
Ramesh
Ramu
RaVi
```

WEEK 4

Write a program that creates a user interface to perform integer division. The user enters two

numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 and Num2 were not integers, the program would throw a Number Format Exception. If Num2 were zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

Aim: To write a program that creates a user interface to perform integer divisions. The user enters
two numbers in the textfields,Num1 and Num2.The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException Display the exception in a message dialog box.
Program:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class ARDemo extends JFrame implements ActionListener
{
JTextField tf1, tf2, tf3;
JButton b;
JLabel l;
ARDemo()
{
Container c = getContentPane();
c.setLayout(new FlowLayout());
l=new JLabel("Enter Numbers and press divide button");
tf1=new JTextField("",5);
tf2=new JTextField("",5);
tf3=new JTextField("",5);
b=new JButton("Divide");
c.add(l);
c.add(tf1);
c.add(tf2);
c.add(b);
c.add(tf3);
b.addActionListener(this);
}
public void actionPerformed(ActionEvent ae)
{
if(ae.getActionCommand()=="Divide")
{
try
{
int n1=Integer.parseInt(tf1.getText());
int n2=Integer.parseInt(tf2.getText());
int n=n1/n2;
tf3.setText(""+n);
}
catch(ArithmeticException e1)
```

```
{
JOptionPane.showMessageDialog (null,"Arthimetic Exception");
}
catch(NumberFormatException e2)
{
JOptionPane.showMessageDialog(null,"NumberFormatException");
}
}
public static void main(String args[])
{
ARDemo ob = new ARDemo();
ob.setSize(800,600);
ob.setVisible(true);
ob.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE;
}
}
```

WEEK 5

Write a java program that creates menu which appears similar to the menu of notepad application
of the Microsoft windows or any editor of your choice.

AIM: To write a java program that creates menu which appears similar to the menu of notepad application
of the Microsoft windows or any editor of your choice.

```
import javax.swing.*;
import java.awt.event.*;

public class Notepad implements ActionListener{
JFrame f;
JMenuBar mb;
JMenu file,edit,help;
JMenuItem cut,copy,paste,selectAll;
JTextArea ta;

Notepad(){
f=new JFrame();

cut=new JMenuItem("cut");
copy=new JMenuItem("copy");
paste=new JMenuItem("paste");
selectAll=new JMenuItem("selectAll");

cut.addActionListener(this);
copy.addActionListener(this);
paste.addActionListener(this);
```

```
selectAll.addActionListener(this);

mb=new JMenuBar();
mb.setBounds(5,5,400,40);

file=new JMenu("File");
edit=new JMenu("Edit");
help=new JMenu("Help");

edit.add(cut);edit.add(copy);edit.add(paste);edit.add(selectAll);


mb.add(file);mb.add(edit);mb.add(help);

ta=new JTextArea();
ta.setBounds(5,30,460,460);

f.add(mb);f.add(ta);

f.setLayout(null);
f.setSize(500,500);
f.setVisible(true);
}

public void actionPerformed(ActionEvent e) {
if(e.getSource()==cut)
ta.cut();
if(e.getSource()==paste)
ta.paste();
if(e.getSource()==copy)
ta.copy();
if(e.getSource()==selectAll)
ta.selectAll();
}

public static void main(String[] args) {
   new Notepad();
}
}
```

Week-6

1. Write a java program that reads a file name from the user, and then displays information about
whether the file exists, whether the file is readable, whether the file is writable, the type of file and length of the bytes.

Aim:To write a java program that reads a file name from the user then displays information

about
whether the file exists, whether the file is readable, whether the file is writable, type of file and the
length of the file in bytes.
Program:

```
import java.io.*;
class FileProp
{
public static void main(String args[]) throws IOException
{
BufferedReader br = new BufferedReader(new InputStreamReader (System.in));
System.out.println("Enter file name :");
String fname = br.readLine();
File f = new File (fname);
System.out.println ("File name: " + f.getName ());
System.out.println ("Path:"+ f.getPath ());
System.out.println ("Absolute Path:"+ f.getAbsolutePath ());
System.out.println ("Parent:"+ f.getParent ());
System.out.println ("Exists:"+ f.exists ());
if ( f.exists() )
{
System.out.println ("Is writable: "+ f.canWrite ());
System.out.println ("Is readable: "+ f.canRead ());
System.out.println ("Is executable: "+ f.canExecute ());
System.out.println ("Is directory: "+ f.isDirectory());
System.out.println ("File size in bytes: "+ f.length());
}
}
}
```

Input / Output Processing:

```
Enter file name :
ramesh
File name: ramesh
Path:ramesh
Absolute Path:G:\ramesh\ramesh
Parent:null
Exists:false
```

2.Write a java program that displays the number of characters, lines and words in a text file.

Aim:To write a java program that displays the number of characters, lines and words in a text file.
Program:

```
import java.io.*;
class FileStat
{
public static void main(String args[]) throws IOException
{
```

```
int pre=' ' , ch , ctr=0 , L=0 , w=1;
String fname;
BufferedReader br=new BufferedReader(new InputStreamReader (System.in));
System.out.print("Enter a file name: ");
fname=br.readLine();
FileInputStream fin = new FileInputStream(fname);
while((ch=fin.read())!=-1)
{
if(ch!=' ' && ch!='\n')
ctr++;
if(ch=='\n')
L++;
if(ch==' ' && pre!=' ')
w++;
pre=ch;
}
System.out.println("Char count="+ctr);
System.out.println("Word count="+(w+(L-1)));
System.out.println("Line count="+L);
}
}
```

Input / Output Processing:

```
Enter a file name: a.txt
Char count=10
Word count=2
Line count=1
```

3.Write a java program that reads a file and displays the file on the screen with line number before each line.

Aim: To write a java program that reads a file and displays the file on the screen, with a line number before each line.

Program:

```
import java.io.*;
class FileRead
{
public static void main(String args[]) throws IOException
{
int ch,ctr=1;
String fname;
BufferedReader br = new BufferedReader(new InputStreamReader (System.in));
System.out.print("Enter a file name: ");
fname=br.readLine();
FileInputStream fin =new FileInputStream(fname);
System.out.print(ctr+"");
while((ch=fin.read())!=-1)
{
System.out.print((char)ch);
```

```
if(ch=='\n')
{
ctr++;
System.out.print(ctr+"");
}
}
fin.close();
}
}
```
Input / Output Processing:
Enter a file name: a.txt
1 hi welcome
2

Week-7
a. Suppose that table named table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas.

Aim:To write a java program to display the table using JTable component. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas.
Program:
```
import javax.swing.*;
import java.awt.*;
import java.io.*;
import java.util.*;
class MyTable extends JFrame
{
JTable tab;
JScrollPane scrollPane;
Vector<String> initData = new Vector<String>();;
String data[][] ;
String cols[];
File file = null;
FileInputStream fis = null;
BufferedInputStream bis = null;
Scanner sc = null;
public MyTable()
{
Container c = getContentPane ();
c.setLayout (new FlowLayout ());
try
{
file = new File("D:\\jqr\\Table.txt");
fis = new FileInputStream(file);
bis = new BufferedInputStream(fis);
sc = new Scanner( new InputStreamReader (bis) );
```

```
while (sc.hasNextLine())
{
initData.add( sc.nextLine());
}
StringTokenizer st = new StringTokenizer( initData.get(0), ",");
cols = new String[ st.countTokens() ];
data = new String[ initData.size()-1][st.countTokens()];
for( int i = 0; i< initData.size(); i++)
{
if( i== 0)
{
int j= 0;
while( st.hasMoreTokens() )
{
cols[j] = st.nextToken();
j++;
}
}
else
{
int k=0;
StringTokenizer str = new StringTokenizer( initData.get(i),",");while( str.hasMoreTokens() )
{
data[i-1][k] = str.nextToken();
k++;
}
}
}
fis.close();
bis.close();
}
catch (FileNotFoundException e)
{
e.printStackTrace();
}
catch (IOException e)
{
e.printStackTrace();
}
JTable tab = new JTable (data, cols);
scrollPane = new JScrollPane(tab);
c.add(scrollPane);
}
}
class TableDemo
{
public static void main(String args[])
{
```

```
MyTable ob = new MyTable();
ob.setTitle("Table Demo");
ob.setSize(600,400);
ob.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE);
ob.setVisible( true );
}
}
```

b. Write a java program that connects to a database using JDBC and does add, delete, modify and
retrieve operations.

Aim:ToWrite a java program that connects to a database using JDBC and does add, delete, modify and
retrieve operations.

```
/ Java program to illustrate
// selecting from Database
import java.sql.*;

public class select
{
   public static void main(String args[])
   {
      String id = "id1";
      String pwd = "pwd1";
      try
      {
         Class.forName("oracle.jdbc.driver.OracleDriver");
         Connection con = DriverManager.getConnection("
               jdbc:oracle:thin:@localhost:1521:orcl", "login1", "pwd1");
         Statement stmt = con.createStatement();

         // SELECT query
         String q1 = "select * from userid WHERE id = '" + id +
                           "' AND pwd = '" + pwd + "'";
         ResultSet rs = stmt.executeQuery(q1);
         if (rs.next())
         {
            System.out.println("User-Id : " + rs.getString(1));
            System.out.println("Full Name :" + rs.getString(3));
            System.out.println("E-mail :" + rs.getString(4));
         }
         else
         {
            System.out.println("No such user id is already registered");
         }
         con.close();
```

```
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}
```

Output :
User-Id : id1
Full Name : geeks for geeks
E-mail :geeks@geeks.org

Week-8
A.Write a java program to handle keyboard events.

```
/* Java Program to demonstrate the event actions associated with a keyboard */
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
class Keyboard_Event implements KeyListener,ActionListener
{
    static JFrame frame;
    static JTextField output;
    static JTextField input;
    //Driver function
    public static void main(String args[])
    {
        //Create a frame
        frame=new JFrame("Keyboard Event");
        frame.setBackground(Color.white);
        frame.setSize(500,500);
        frame.setLayout(null);
        //Create a text field for output
        output=new JTextField();
        output.setBounds(0,0,500,50);
        frame.add(output);
        //Create a text field for input
        input=new JTextField();
        input.setBounds(0,400,500,50);
        frame.add(input);
        //Create an exit button
        JButton exit=new JButton("Exit");
        exit.setBounds(220,200,60,30);
        frame.add(exit);
        //Create an object of the class
        Keyboard_Event obj=new Keyboard_Event();
        //Associate KeyListener with input
```

```
        input.addKeyListener(obj);
        //Associate ActionListener with exit
        exit.addActionListener(obj);
        frame.setVisible(true);
    }
    //function to dispose the frame when exit button is clicked
    @Override
    public void actionPerformed(ActionEvent e)
    {
        frame.dispose();
    }
    /*function to display the unicode of key released
      and the character if it is a letter or a digit*/
    @Override
    public void keyReleased(KeyEvent e)
    {
        output.setText("");
        output.setText("Key Released : "+e.getKeyCode());
        if(Character.isLetter(e.getKeyChar()))
            keyTyped(e);
        }
}
```