

**Domenico Valle**

# **Esercizi JAVA**

## Indice generale

SELEZIONE.....	3
Casa al mare.....	3
Bisestile.....	5
Giorno successivo.....	7
Quattro operazioni.....	10
ITERAZIONE.....	12
Somma n numeri con do ... while.....	12
Somma n numeri con while.....	14
Somma n numeri con for.....	16
Somma n numeri con controllo su valore sentinella, utilizzando un ciclo do ... while.....	18
Somma n numeri con controllo su valore sentinella, utilizzando un ciclo while.....	20
Successione di Fibonacci: trovare il primo termine > N.....	22
VETTORI.....	24
Caricamento e stampa di un vettore.....	24
Caricamento vettore; stampare i valori di posizione pari.....	26
Caricamento vettore; stampare i valori pari.....	28
Dati due vettori costruire un terzo vettore. ....	30
MATRICI.....	32
Caricamento e stampa di una matrice.....	32
Visualizzare una matrice quadrata NxN per diagonali (nel senso della diag.principale).....	34
File.....	37
Gestione file ad accesso diretto.....	37



```
/*
 * Una società immobiliare vende appartamenti al mare ai seguenti prezzi :
 distanza dal mare <= 200 m          2000 euro al mq
 distanza dal mare > 200 m e <= 1000 m      1500 euro al mq
 distanza dal mare > 1000 m          1000 euro al mq
 Dati in input la superficie dell'appartamento e la distanza dal mare,
 calcolare e stampare il prezzo dell'appartamento.
 *
 */
package casa_mare;

import javax.swing.*;

/**
 *
 * @author
 */
public class Main
{

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args)
    {
        int mq, d, costo = 0;
        mq = Integer.parseInt(JOptionPane.showInputDialog("metri quadrati appartamento"));
        d = Integer.parseInt(JOptionPane.showInputDialog("distanza dal mare (in metri)"));

        if (d <= 200)
        {
            costo = mq * 2000;
        }
        else
        {
            if (d <= 1000)
            {
                costo = mq * 1500;
            }
            else
            {
                costo = mq * 1000;
            }
        }
        System.out.println("Prezzo dell'appartamento : " + costo);
        System.exit(0);
    }
}
```

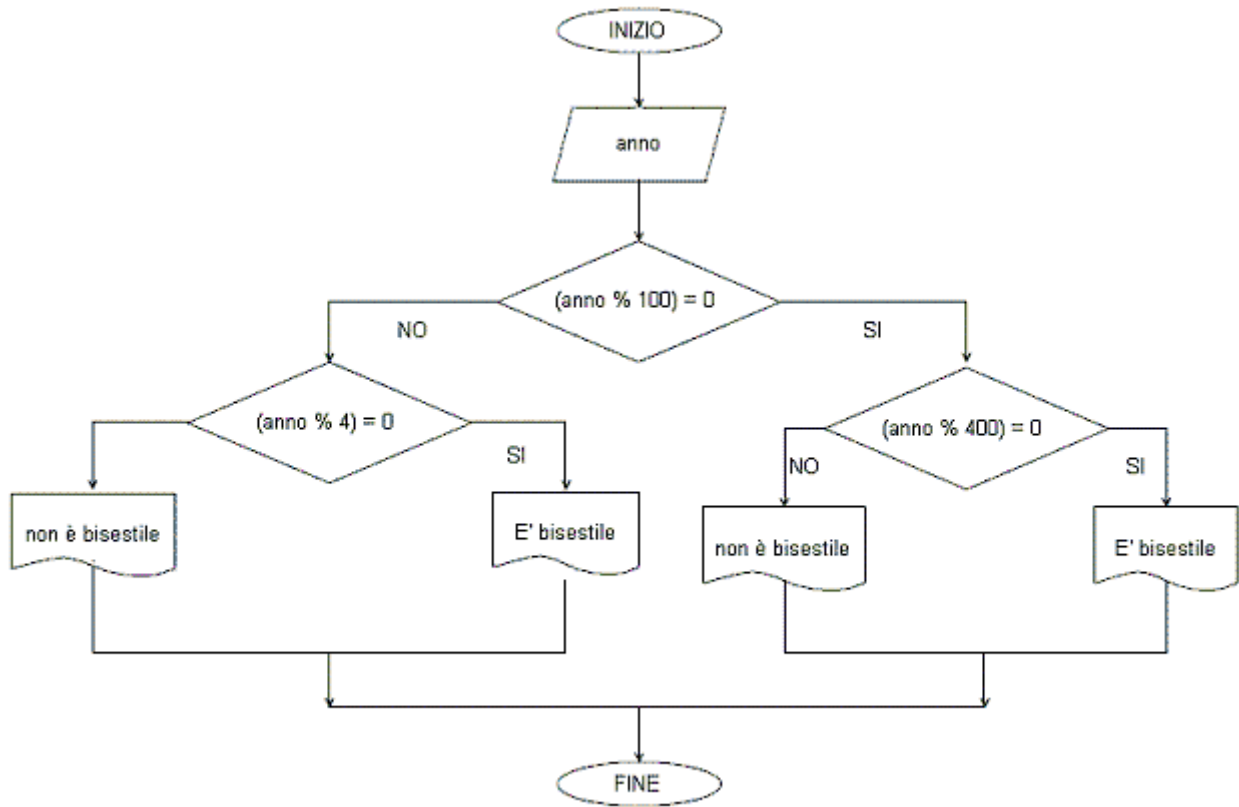
**Bisestile**

/\*

\* Dato in input un valore indicante un anno, stabilire se è bisestile.

\*

\*/



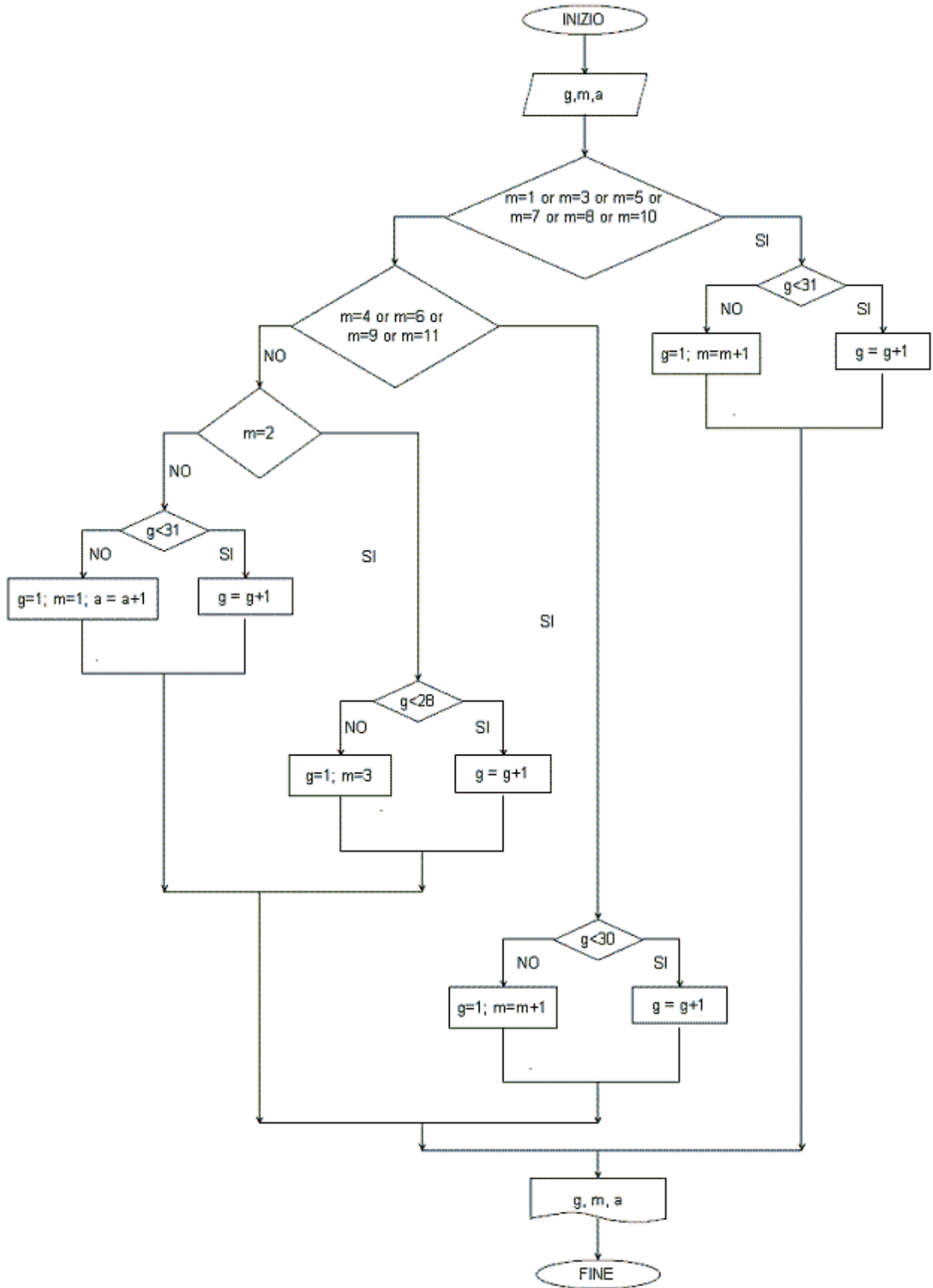
```
/*
 * Dato in input un valore indicante un anno, stabilire se è bisestile.
 *
 */

package bisestile;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int anno;
        anno = Integer.parseInt(JOptionPane.showInputDialog("anno"));
        if (anno%100 == 0)
        {
            if (anno % 400 == 0)
            {
                System.out.println("l'anno: " + anno + " è bisestile");
            }
            else
            {
                System.out.println("l'anno: " + anno + " non è bisestile");
            }
        }
        else
        {
            if (anno % 4 == 0)
            {
                System.out.println("l'anno: " + anno + " è bisestile");
            }
            else
            {
                System.out.println("l'anno: " + anno + " non è bisestile");
            }
        }
        System.exit(0);
    }
}
```

**Giorno successivo**

/\*  
 \* In input una data come giorno, mese, anno, in output il giorno successivo, per anno non bisestile.  
 \*/



```
/*
 * In input una data come giorno, mese, anno, in output il giorno successivo,
 * per anno non bisestile.
 */

package giorno_successivo;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

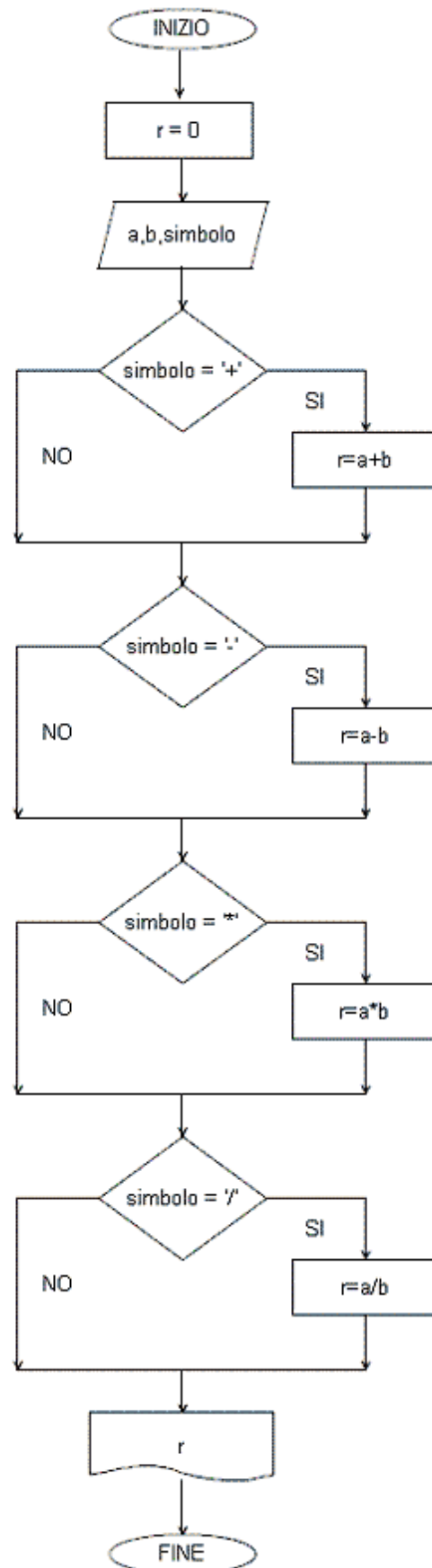
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int g,m,a;
        int gattuale,mattuale,aattuale;
        g = Integer.parseInt(JOptionPane.showInputDialog("giorno"));
        m = Integer.parseInt(JOptionPane.showInputDialog("mese"));
        a = Integer.parseInt(JOptionPane.showInputDialog("anno"));
        gattuale=g;
        mattuale=m;
        aattuale=a;
        if (m==1 || m==3 || m==5 || m==7 || m==8 || m==10 )
        {
            if (g<31)
            {
                g++;
            }
            else
            {
                g=1;
                m++;
            }
        }
        else
        {
            if (m==4 || m==6 || m==9 || m==11 )
            {
                if (g<30)
                {
                    g++;
                }
                else
                {
                    g=1;
                    m++;
                }
            }
            else
            {
                if (m==2)
```



```
{
    if (g<28)
    {
        g++;
    }
    else
    {
        g=1;
        m++;
    }
}
else
{
    if (g<31)
    {
        g++;
    }
    else
    {
        g=1;
        m=1;
        a++;
    }
}
}
}
System.out.println("data attuale   : "+ gattuale + "/" + mattuale + "/" + aattuale +
    "\ngiorno successivo: "+ g + "/" + m + "/" + a );
}
}
```

## Quattro operazioni

Dati in input 2 numeri e il simbolo di una delle quattro operazioni aritmetiche calcolare il risultato dell'operazione indicata.



```
/*
 * Dati in input 2 numeri e il simbolo di una delle quattro operazioni aritmetiche calcolare il
 risultato dell'operazione indicata.
 *
 */

package quattro_operazioni;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

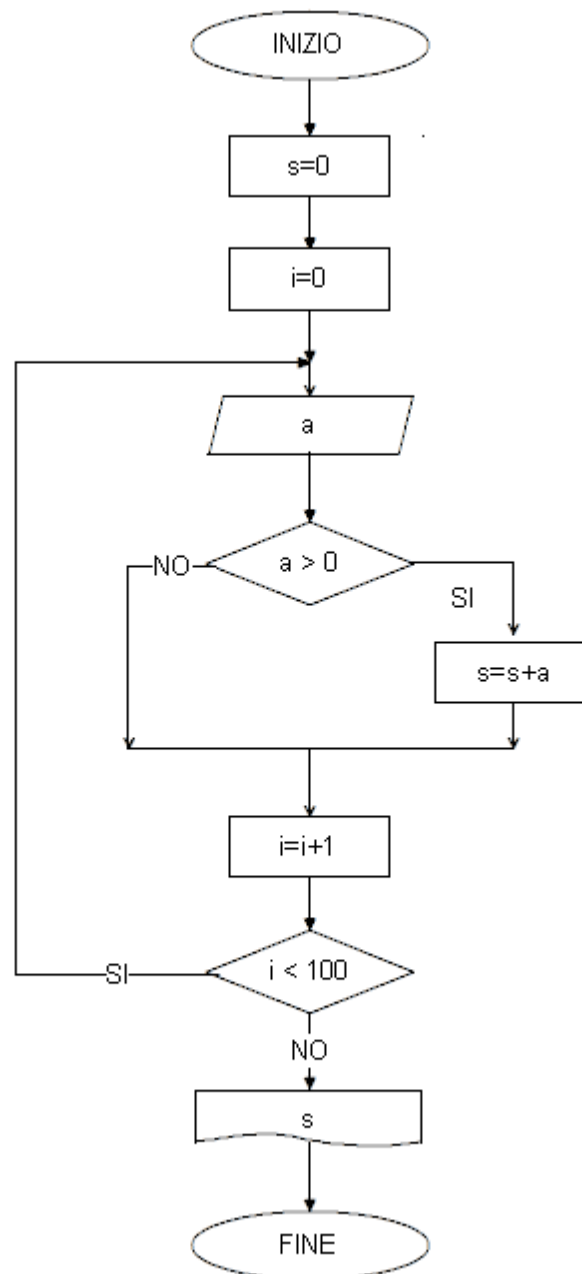
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int a,b,r=0;
        String simbolo;
        a = Integer.parseInt(JOptionPane.showInputDialog("I operando (A) "));
        b = Integer.parseInt(JOptionPane.showInputDialog("II operando (B) "));
        simbolo = JOptionPane.showInputDialog("operatore (simbolo) ");
        if (simbolo.equals("+"))
        {
            r = a+b;
        }
        if (simbolo.equals("-"))
        {
            r = a-b;
        }
        if (simbolo.equals("*"))
        {
            r = a*b;
        }
        if (simbolo.equals("/"))
        {
            r = a/b;
        }

        System.out.println("il risultato di " + a + " "+ simbolo + " "+b+ " è: "+ r );
    }

}
```

**ITERAZIONE****Somma n numeri con do ... while**

```
/*  
 * In input 100 valori interi; calcolare la somma dei valori positivi con un ciclo do...while.  
 */
```



```
/*
 * In input 100 valori interi; calcolare la somma dei valori positivi con un ciclo do...while.
 */

package input_100_somma_positivi_do;
import javax.swing.*;

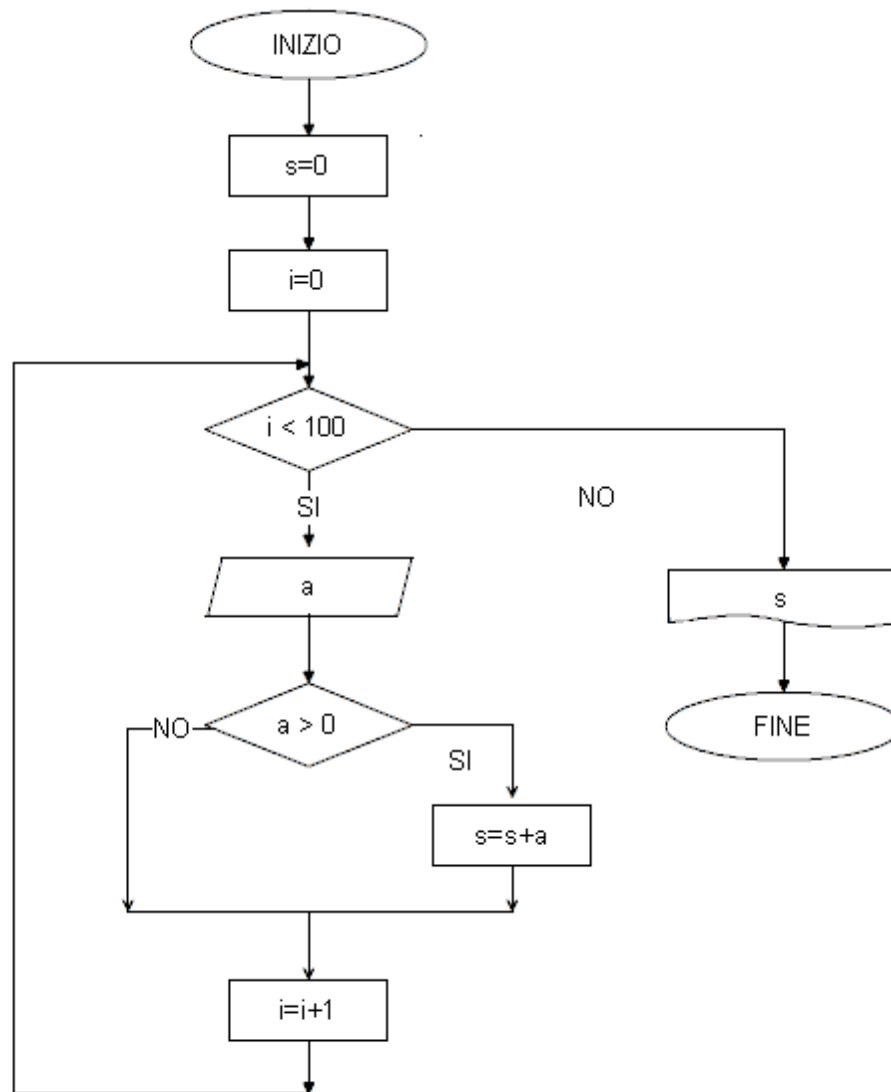
/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int s=0;
        int i=0;
        int a;

        do
        {
            a=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore num. "+i ));
            if (a >0)
            {
                s+=a;
            }
            i++;
        }while(i<100);
        System.out.println("La somma dei valori positivi inseriti è: "+s);
    }
}
```

**Somma n numeri con while**

```
/*  
* In input 100 valori interi; calcolare la somma dei valori positivi con un ciclo while.  
*/
```



```
/*
 * In input 100 valori interi; calcolare la somma dei valori positivi con un ciclo while.
 */

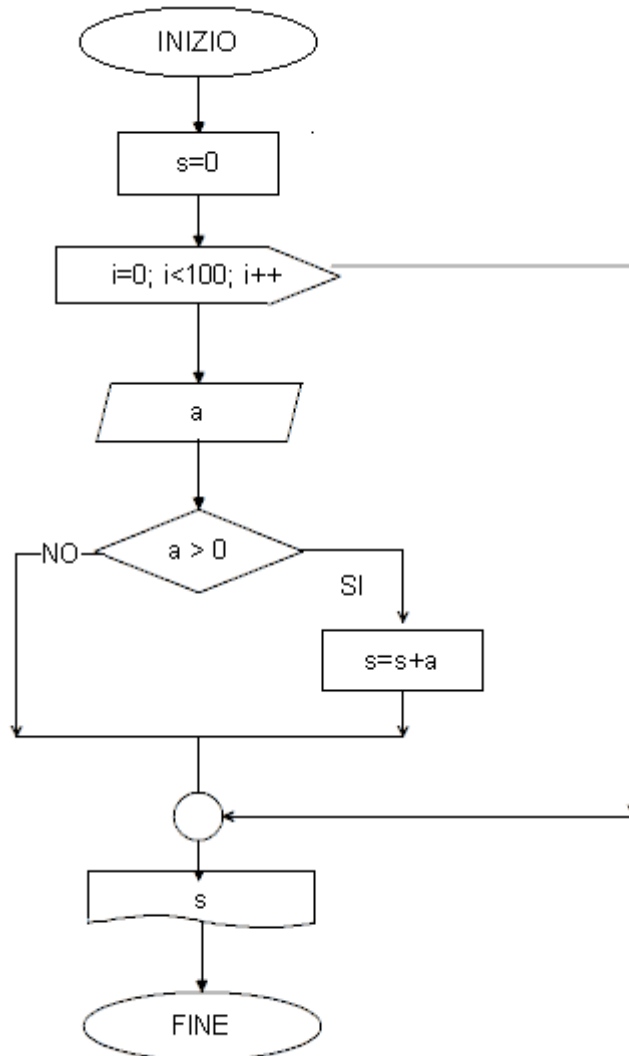
package input_100_somma_positivi_while;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int s=0;
        int i=0;
        int a;

        while(i<100)
        {
            a=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore n. "+i));
            if (a>0)
            {
                s+=a;
            }
            i++;
        }
        System.out.println("La somma dei valori positivi inseriti è: "+s);
    }
}
```

**Somma n numeri con for**

```
/*  
 * In input 100 valori interi; calcolare la somma dei valori positivi con un ciclo for.  
 *  
 */
```





```
/*
 * In input 100 valori interi; calcolare la somma dei valori positivi con un ciclo for.
 *
 */

package input_100_somma_positivi_for;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int s=0;
        int a;

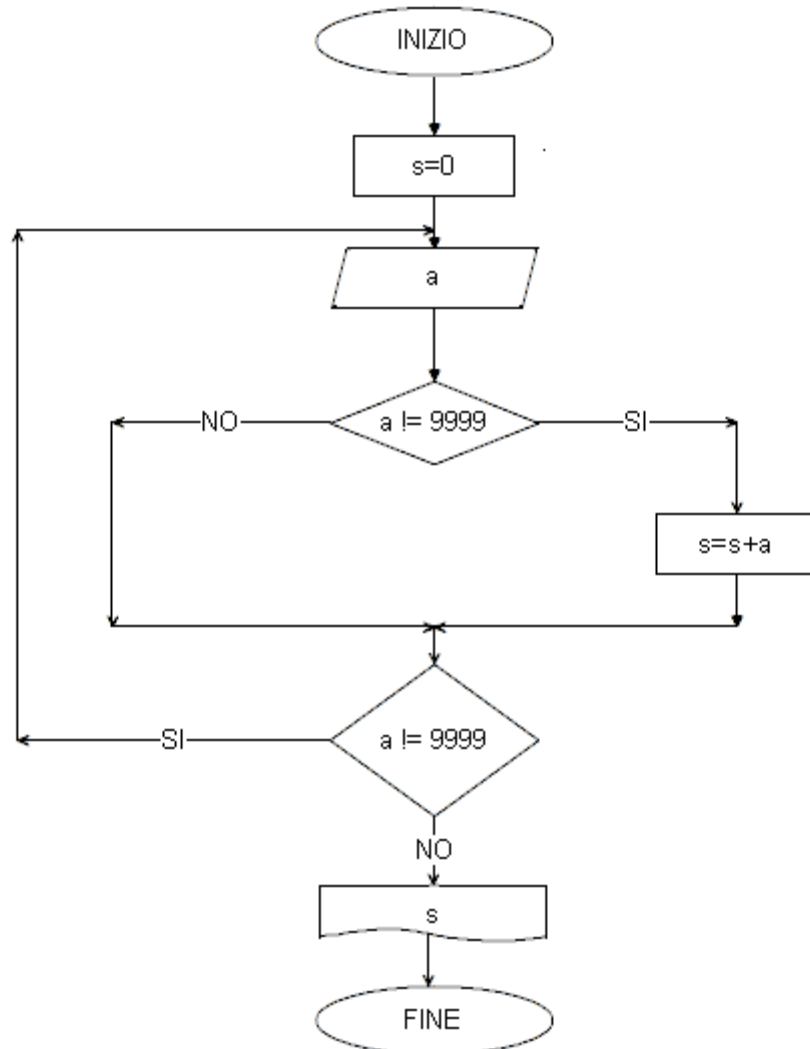
        for(int i=0;i<100;i++)
        {
            a=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore n. "+i));
            if (a>0)
            {
                s+=a;
            }
        }
        System.out.println("La somma dei valori positivi inseriti è: "+s); }
}
```

**Somma n numeri con controllo su valore sentinella, utilizzando un ciclo do ... while**

/\*

\*In input una serie di valori; l'input ha termine quando viene inserito il valore 9999 (sentinella);  
sommare i valori inseriti, escludendo il valore sentinella.(do...while)

\*/



```
/*
 *In input una serie di valori; l'input ha termine quando viene inserito il valore 9999 (sentinella);
   sommare i valori inseriti, escludendo il valore sentinella.(do...while)
 */

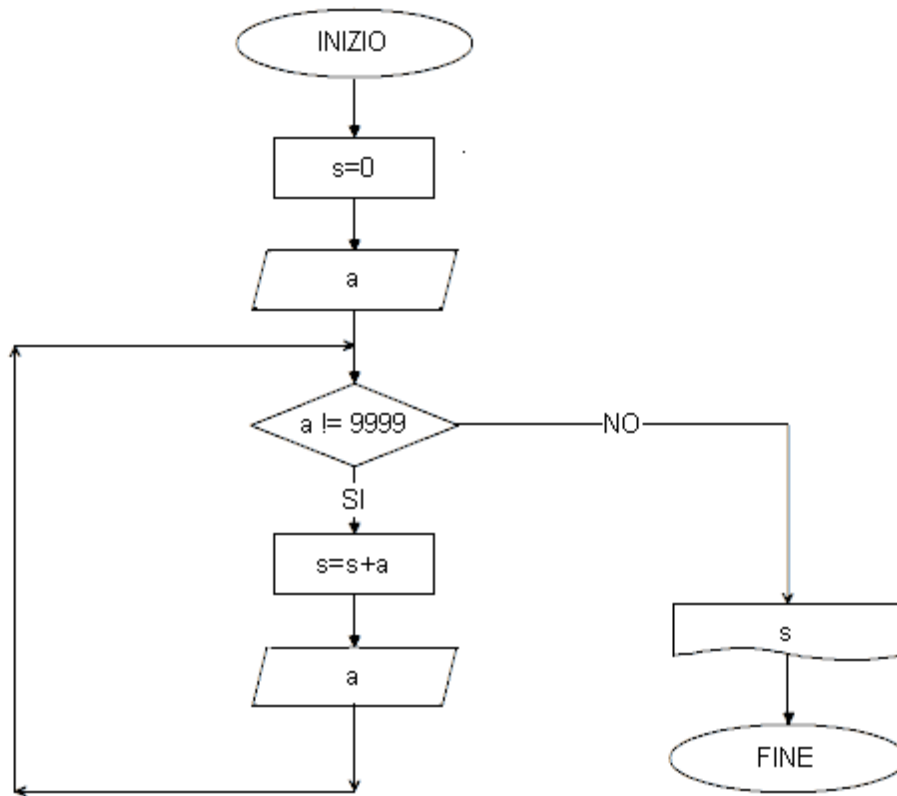
package input_valori_con_sentinella_do_while;
import javax.swing.*;

/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int s=0;
        int i=0;
        int a;
        do
        {
            a=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore num. "+i ));
            if (a!=9999)
            {
                s+=a;
                i++;
            }
        }while(a!=9999);
        System.out.println("La somma dei valori positivi inseriti è: "+s);
        System.exit(0);
    }
}
```

**Somma n numeri con controllo su valore sentinella, utilizzando un ciclo while**

```
/*  
 * In input una serie di valori; l'input ha termine quando viene inserito il valore 9999  
 (sentinella); sommare i valori inseriti, escludendo il valore sentinella.(while)  
 */
```



```
/*
 * In input una serie di valori; l'input ha termine quando viene inserito il valore 9999
 (sentinella); sommare i valori inseriti, escludendo il valore sentinella.(while)
 */

package input_valori_con_sentinella_while;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int s=0;
        int i=0;
        int a;

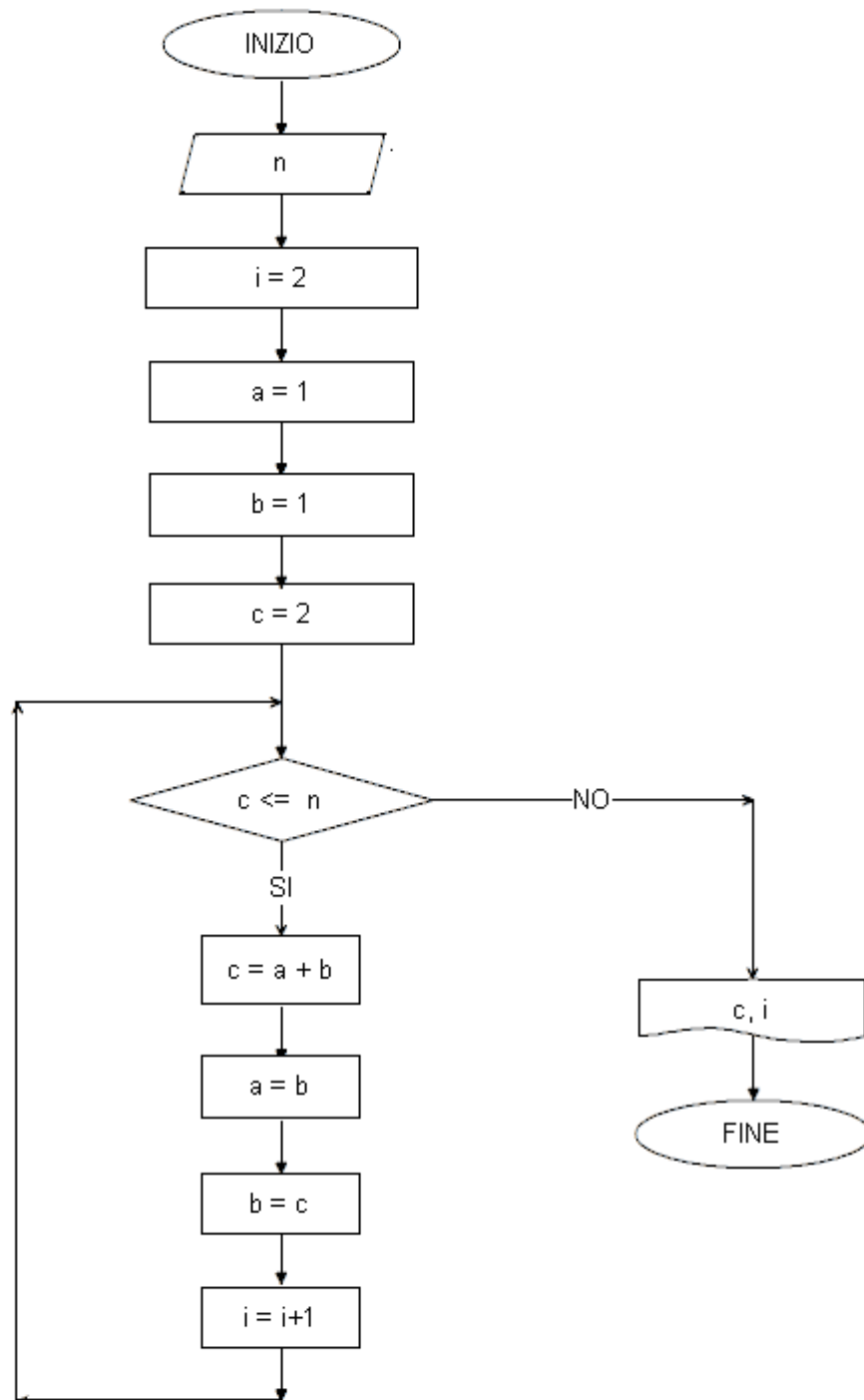
        a=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore num. "+i ));
        while(a!=9999)
        {
            s+=a;
            i++;
            a=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore num. "+i ));
        }
        System.out.println("La somma dei valori positivi inseriti è: "+s);
    }
}
```

**Successione di Fibonacci: trovare il primo termine  $> N$** 

/\*

\* Data la successione di Fibonacci, stabilire qual è il primo termine della successione  $>$  di un valore dato N.

\*/



```
/*
 * Data la successione di Fibonacci, stabilire qual è il primo termine della successione > di un
valore dato N.
 */

package fibonacci_1;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

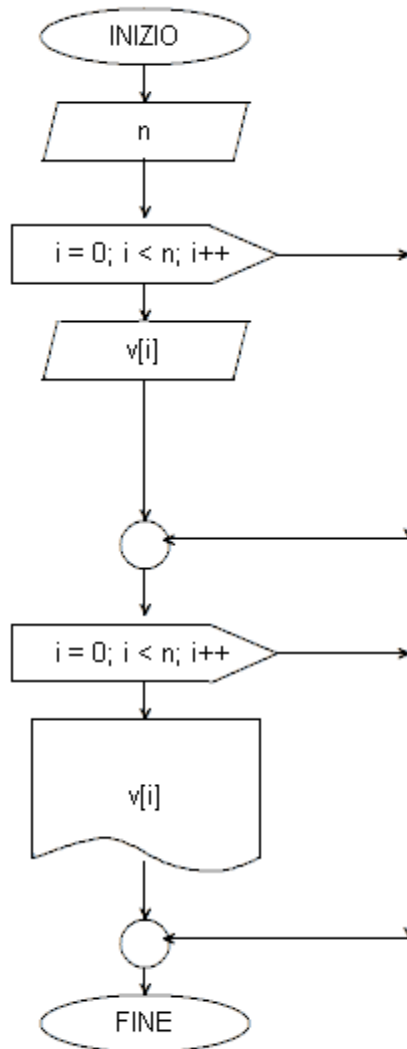
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int i=2;
        int a=1;
        int b=1;
        int c=2;
        int n;

        n=Integer.parseInt(JOptionPane.showInputDialog("Inserisci numero N "));

        while(c<=n)
        {
            c = a+b;
            a=b;
            b=c;
            i++;
        }
        System.out.println("Il I termine maggiore di " +n + " è: "+ c + " dopo "+i + " termini");
        System.exit(0);
    }
}
```

**VETTORI****Caricamento e stampa di un vettore**

```
/*  
 * caricamento e stampa di un vettore  
 */
```





```
/*
 * caricamento e stampa di un vettore
 */

package carica_stampa;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int n;

        n=Integer.parseInt(JOptionPane.showInputDialog("Inserisci numero N "));
        int v[] = new int[n];

        //----- Carica -----
        for(int i=0; i<n ;i++)
        {
            v[i]=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore "+i+ "-esimo"));
        }

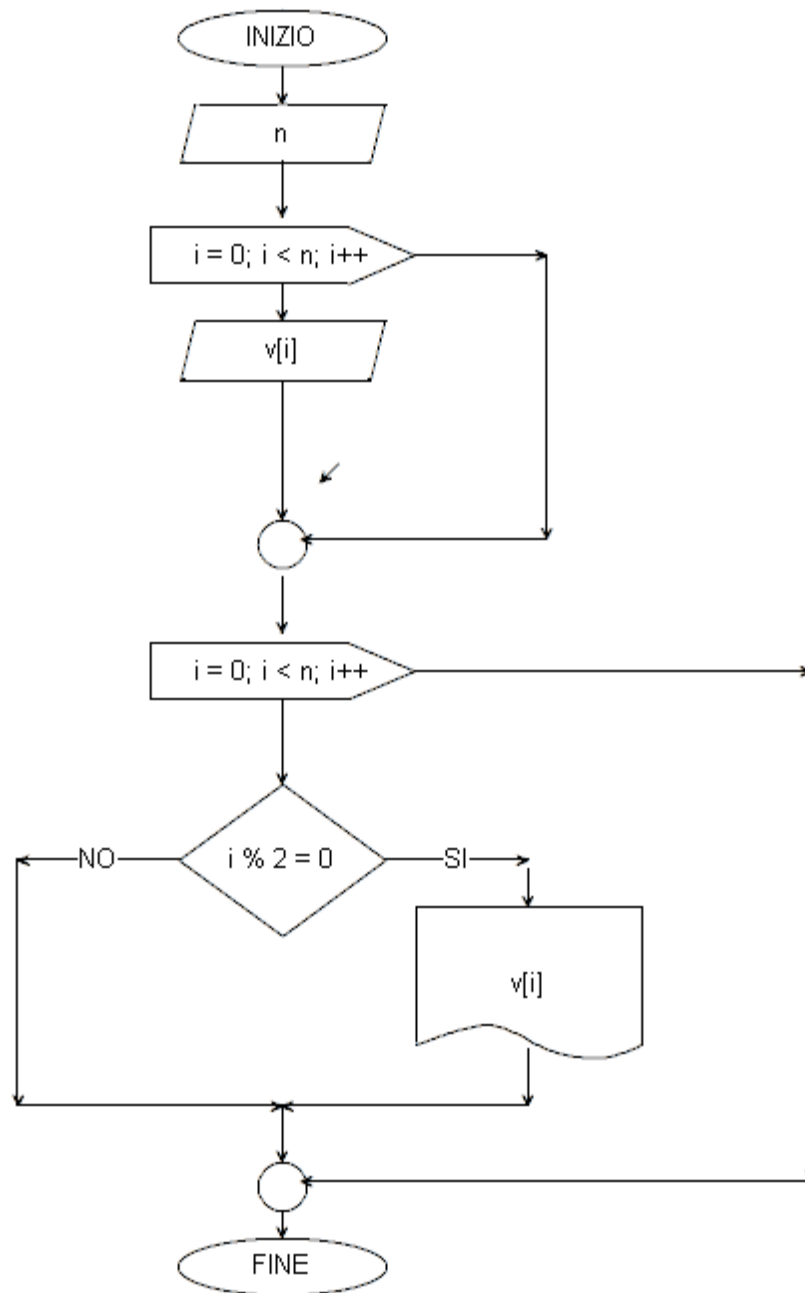
        //----- Stampa -----
        for(int i=0; i<n ;i++)
        {
            System.out.println("elemento v["+i+"] = "+ v[i]);
        }
        System.exit(0);
    }
}
```

**Caricamento vettore; stampare i valori di posizione pari**

/\*

\* Caricamento vettore; stampare i valori di posizione pari

\*/



```
/*
 * Caricamento vettore; stampare i valori di posizione pari
 */

package posizione_pari;

import javax.swing.JOptionPane;

/**
 *
 * @author DValle
 */
public class Main {

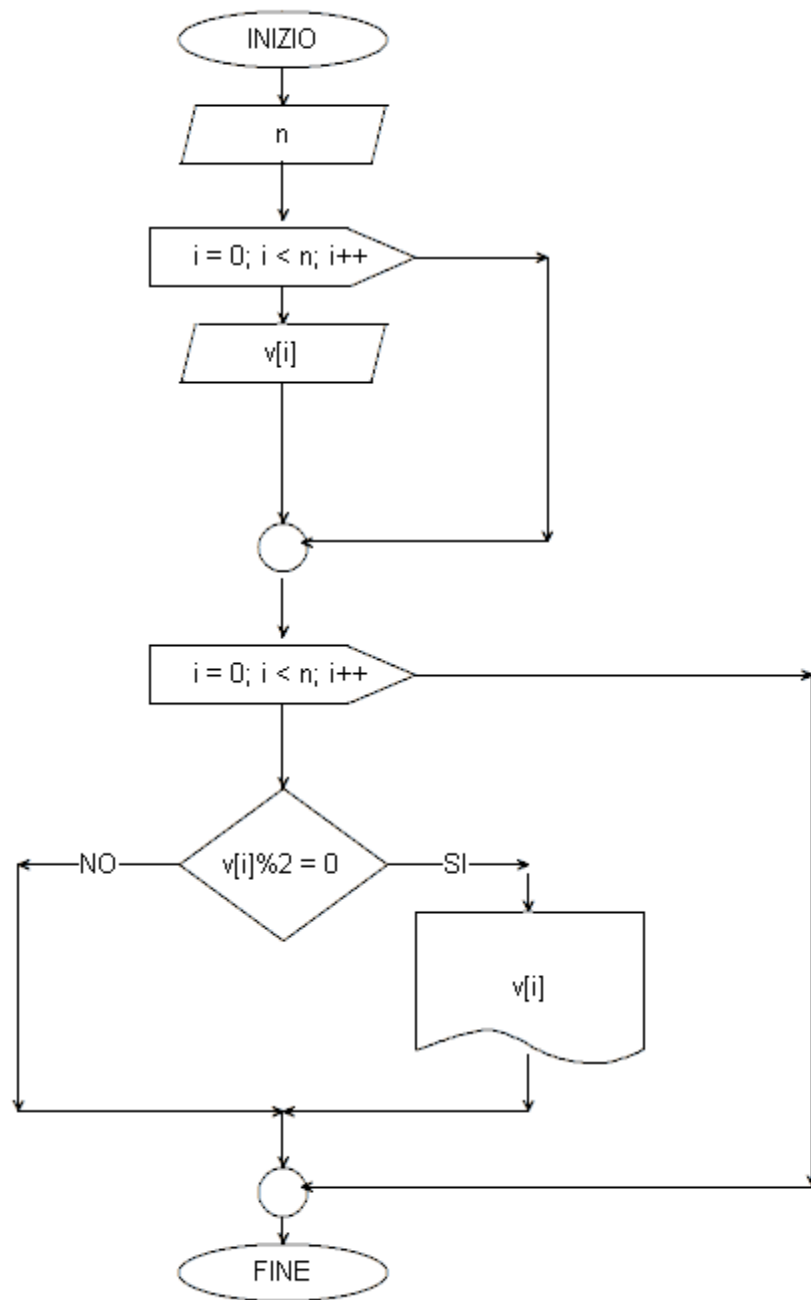
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int n;

        n=Integer.parseInt(JOptionPane.showInputDialog("Inserisci numero N "));
        int v[] = new int[n];

        for(int i=0; i<n ;i++)
        {
            v[i]=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore "+i+ "-esimo"));
        }
        for(int i=0; i<n ;i++)
        {
            if (i%2 == 0)
            {
                System.out.println("elemento v["+i+"] = "+ v[i]);
            }
        }
        System.exit(0);
    }
}
```

**Caricamento vettore; stampare i valori pari**

```
/*  
* caricamento vettore; stampare i valori pari  
*/
```



```
/*
 * caricamento vettore; stampare i valori pari
 */

package valori_pari;

import javax.swing.JOptionPane;

/**
 *
 * @author DValle
 */
public class Main {

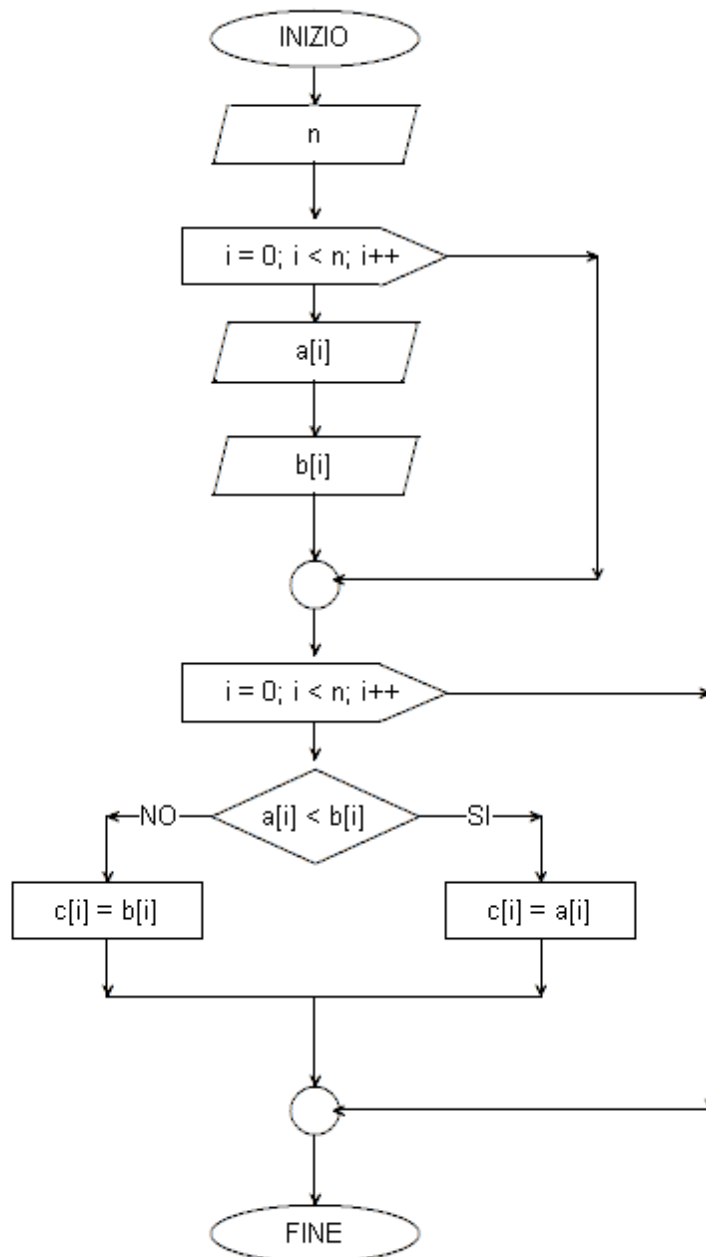
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int n;

        n=Integer.parseInt(JOptionPane.showInputDialog("Inserisci numero N "));
        int v[] = new int[n];

        for(int i=0; i<n ;i++)
        {
            v[i]=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore "+i+ "-esimo"));
        }
        for(int i=0; i<n ;i++)
        {
            if (v[i]%2 == 0)
            {
                System.out.println("elemento v["+i+"] = "+ v[i]);
            }
        }
        System.exit(0);
    }
}
```

**Dati due vettori costruire un terzo vettore.**

/\*  
Dati due vettori, costruire un terzo vettore contenente in ogni elemento il valore più piccolo dei corrispondenti elementi dei vettori di input  
\*/



```
/*
  Dati due vettori, costruire un terzo vettore contenente in ogni elemento il valore più
  piccolo dei corrispondenti elementi dei vettori di input
  */

package elemento_minimo_terzo_vettore;

import javax.swing.JOptionPane;

/**
 *
 * @author DValle
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int n;

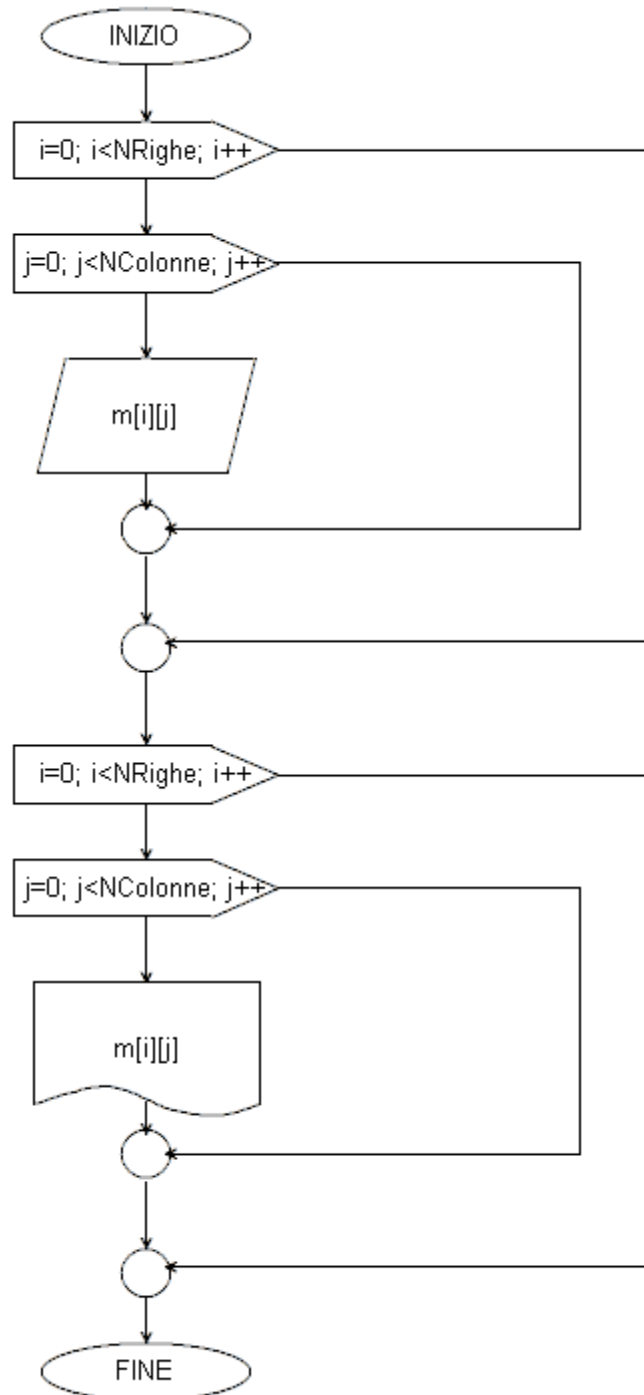
        n=Integer.parseInt(JOptionPane.showInputDialog("Inserisci numero N "));
        int a[] = new int[n];
        int b[] = new int[n];
        int c[] = new int[n];

        for(int i=0; i<n ;i++)
        {
            a[i]=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore "+i+ "-esimo"));
            b[i]=Integer.parseInt(JOptionPane.showInputDialog("Inserisci il valore "+i+ "-esimo"));
        }
        for(int i=0; i<n ;i++)
        {
            if (a[i]<b[i])
            {
                c[i]=a[i];
            }
            else
            {
                c[i]=b[i];
            }
        }
        for(int i=0; i<n ;i++)
        {
            System.out.println(a[i]+" "+b[i]+ " "+c[i]);
        }
    }
}
```

## MATRICI

### Caricamento e stampa di una matrice

```
/*  
 * Caricamento e stampa di una matrice  
 */
```





```
/*
 * Caricamento e stampa di una matrice
 */

package matrice_carica_stampa;

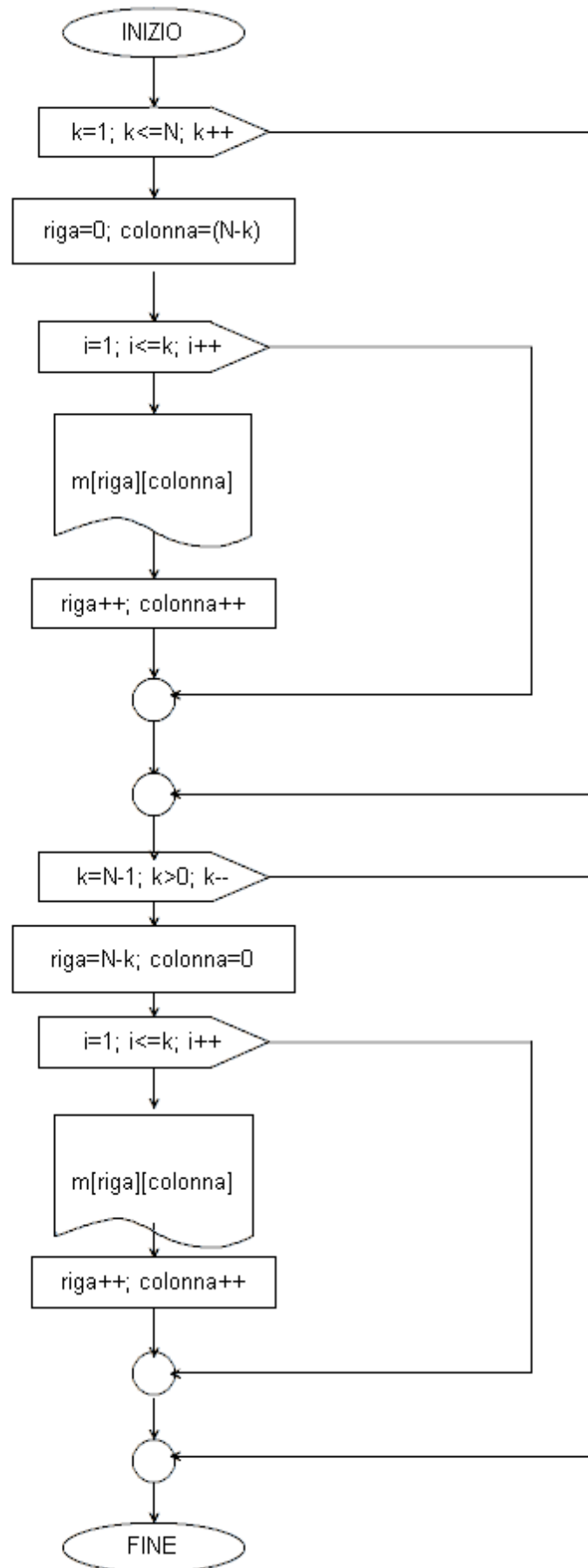
import javax.swing.JOptionPane;

/**
 *
 * @author mimmo
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int m[][] = new int[3][4];
        String messaggio;
        // ----- carica matrice -----
        for (int i=0;i<m.length;i++)
        {
            for(int j=0;j<m[0].length;j++)
            {
                messaggio = "elemento riga "+ i + " colonna "+j;
                m[i][j] = Integer.parseInt(JOptionPane.showInputDialog(messaggio));
            }
            System.out.println();
        }

        // stampa matrice
        for (int i=0;i<m.length;i++)
        {
            for(int j=0;j<m[0].length;j++)
                System.out.print(" "+ m[i][j]);
            System.out.println();
        }
        System.exit(0);
    }
}
```

**Visualizzare una matrice quadrata NxN per diagonali (nel senso della diag.principale)**



```
/*
 * Visualizzare una matrice quadrata NxN per diagonali (nel senso della diagonale principale)
 *
 */

package matriceperdiagonali_1;
import javax.swing.JOptionPane;
import java.text.*;
/**
 *
 * @author mimmo
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        int n;
        n = Integer.parseInt(JOptionPane.showInputDialog("Dimensioni matrice"));
        int m[][] = new int[n][n];

        // ----- carica
        int a=0;
        for (int i=0;i<m.length;i++)
            for (int j=0;j<m.length;j++)
                {
                    a++;
                    m[i][j]=a;
                }

        // ----- stampa
        for (int i=0;i<m.length;i++)
            {
                for (int j=0;j<m.length;j++)
                    {
                        //NumberFormat è una classe astratta per cui non si
                        //può avere new NumberFormat
                        NumberFormat formatter = NumberFormat.getInstance();
                        formatter.setMaximumIntegerDigits(2);
                        formatter.setMinimumIntegerDigits(2);
                        System.out.print(formatter.format(m[i][j])+" ");
                    }
                System.out.println();
            }
        System.out.println();
        //----- visualizza per diagonali ( ;
        int riga=0, colonna=0;
        //-- dall'angolo in alto a destra alla diagonale principale
        for (int k=1;k<=m.length;k++)
            {
                riga=0;
                colonna = m.length-k;
                for (int i=1;i<=k;i++)
                    {
```

```
    NumberFormat formatter = NumberFormat.getInstance();
    System.out.print(formatter.format(m[riga][colonna])+" ");
    riga++;
    colonna++;
}
System.out.println();
}
//-- dalla diagonale principale-1 all'angolo in basso a sinistra
for (int k=m.length-1;k>0;k--)
{
    riga=m.length-k;
    colonna = 0;
    for (int i=1;i<=k;i++)
    {
        NumberFormat formatter = NumberFormat.getInstance();
        System.out.print(formatter.format(m[riga][colonna])+" ");
        riga++;
        colonna++;
    }
    System.out.println();
}
System.exit(0);
}
}
```

## File

### Gestione file ad accesso diretto

```
/*
 * Libro.java
 * Attributi e costruttore dell'oggetto "libro" ; metodi per leggere e scrivere un record "libro"
 */

package appfile;
import java.io.*;
/**
 *
 * @author DValle
 */
public class Libro {

    private String titolo;
    private int prezzo;

    /** Creates a new instance of Libro */
    public Libro() { }

    public Libro(String titolo, int prezzo)
    {
        this.titolo = titolo;
        this.prezzo = prezzo;
    }

    public int getPrezzo()
    {
        return prezzo;
    }

    public String getTitolo()
    {
        return titolo;
    }
}
```

```
public void scrivi(RandomAccessFile a)
{
    try
    {
        a.writeUTF(titolo);
        a.writeInt(prezzo);
    } catch (Exception e) {}
}

public void leggi(RandomAccessFile a)
{
    try
    {
        titolo = a.readUTF();
        prezzo = a.readInt();
    } catch (Exception e) {}
    return;
}

}



---


/*
 * Main.java
 * Creazione e lettura di un file random i cui record contengono "oggetti" di tipo "Libro"
 */

package appfile;
import java.io.*;
import javax.swing.*;
/**
 *
 * @author DValle
 */
public class Main {

    /** Creates a new instance of Main */
    public Main() {
```

```

}

public static void main(String[] args) {

    Libro p;
    String tito;
    int pre;
    try
    {
        File f = new File("c:\\libri.dat");
        RandomAccessFile ra = new RandomAccessFile(f,"rw");
        if (f.exists()) ra.seek(ra.length());

        for (int i=0; i<3; i++)
        {
            tito = "titolo - " +(i+ra.length()); //JOptionPane.showInputDialog("titolo");
            pre = 100+(i+(int)ra.length());
            //Integer.parseInt( JOptionPane.showInputDialog("prezzo"));
            tito = riempi(tito,30);
            p = new Libro(tito,pre);
            p.scrivi(ra);
        }
        ra.seek(0);
        JOptionPane.showMessageDialog(null, " "+ra.getFilePointer()+" "+ ra.length());
        p= new Libro();
        while(ra.getFilePointer() < ra.length() )
        {
            p.leggi(ra);
            String x="";
            x = p.getTitolo() +" "+p.getPrezzo()+"\n";
            JOptionPane.showMessageDialog(null, x);
        }
        ra.close();
    }catch(Exception e){
        JOptionPane.showMessageDialog(null, "errore" +e.getMessage() );
    }
    System.exit(0);
}

```

```
public static String riempi(String s, int l)
{
    for (int i=s.length();i<l;i++)
        s = s + " ";
    return s;
}
}
```