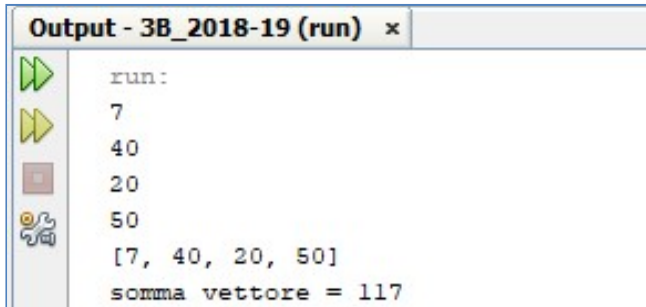


Scheda di lavoro (gestione vettori: applicazioni Console e con interfaccia grafica)

03/05/2019

Confronta le due diverse applicazioni per la gestione di un vettore di interi

Applicazione Console

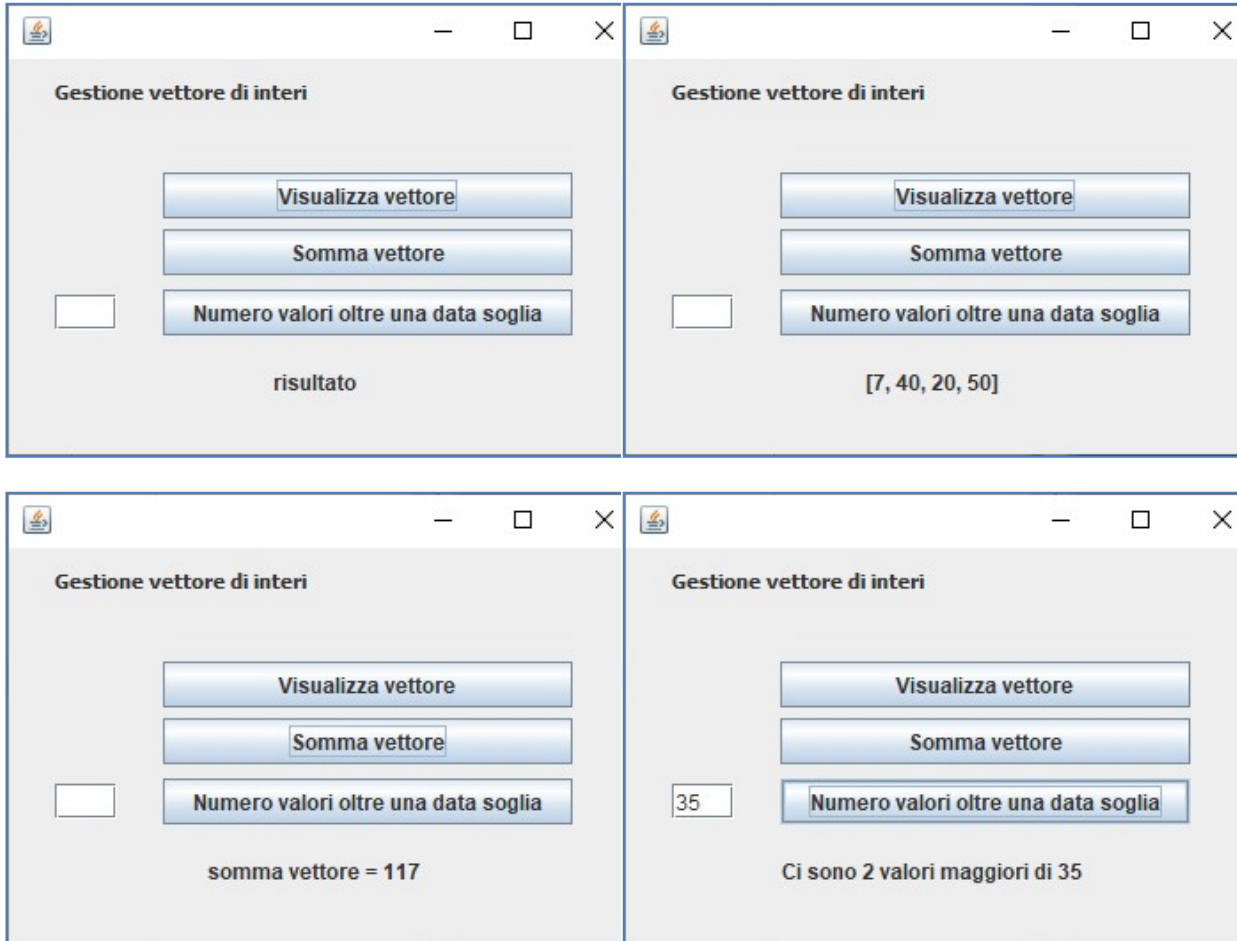


```
Output - 3B_2018-19 (run) x
run:
7
40
20
50
[7, 40, 20, 50]
somma vettore = 117
```

```
1 package pkg3b_2018.pkg19;
2 import java.util.*;
3
4 public class Main {
5     public static void main(String[] args) {
6         int V[] = {7,40,20,50};
7         int s; //somma vettore
8         visualizzaVettore(V);
9         System.out.println(Arrays.toString(V));
10        s = sommaVettore(V);
11        System.out.println("somam vettore = " + s);
12    }
13
14    static void visualizzaVettore (int x []) {
15        for (int i=0; i<x.length; i++) {
16            System.out.println(x[i]);
17        }
18    }
19
20    static int sommaVettore (int x []) {
21        int totale = 0;
22        for (int i=0; i<x.length; i++) {
23            totale = totale + x[i];
24        }
25        return totale;
26    }
27 }
```

Scheda di lavoro (gestione vettori: applicazioni Console e con interfaccia grafica di due valori interi)
03/05/2019

Applicazione con interfaccia grafica



```
1 package pkg3b_2018.pkg19;
2 import java.util.Arrays;
3
4 public class NewJFrameVettore extends javax.swing.JFrame {
5     int V[] = {7,40,20,50};
6
7     public NewJFrameVettore() {...3 lines }
8
9     @SuppressWarnings("unchecked")
10    // <editor-fold defaultstate="collapsed" desc="Generated Code">
11    private void initComponents() {...79 lines } // </editor-fold>
12
13    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
14        // visualizza vettore
15        jLabel2.setText(Arrays.toString(V));
16    }
17
18    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
19        // somma vettore
20        int s; //somma vettore
21        s = sommaVettore(V);
22        jLabel2.setText("somma vettore = " + s);
23    }
24
25    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
26        // Numero valori oltre una data soglia
27        int soglia;
28        int numero;
29        String tmp;
30        tmp = jTextField1.getText();
31        soglia = Integer.valueOf(tmp);
32        numero = contaSopraSoglia(V,soglia);
33        jLabel2.setText("Ci sono " + numero + " valori maggiori di " + soglia);
34    }
35 }
```