Fondamenti di Informatica (Stat. Gest.)
Prova scritta del 4 giugno 2009

Cognome	Nome
Corso di laurea	

1) (10 punti) **Completare la definizione della classe Java seguente, scrivendo i metodi e le porzioni di metodo mancanti (al posto dei .....) .**

```
import javax.swing.JOptionPane;

public class ElementiArray {
   public static double[] leggiSequenza() {
      int nEl=Integer.parseInt(JOptionPane.showInputDialog("Quanti elementi?"));

      double[] a = new .......

      ............

      return a;
   }

   public static void stampaSequenza( ...... ) {
      ............
   }

/*
AGGIUNGERE QUI UN METODO contaEsterni, CHE CALCOLA E RESTITUISCE (senza
visualizzare nulla) IL NUMERO DI ELEMENTI DELLA SEQUENZA ESTERNI ALL'INTERVALLO
DI TOLLERANZA SPECIFICATO
QUESTO METODO DEVE ESSERE RICHIAMABILE COME MOSTRATO NEL main
*/

   public static void main(String[] args) {
      double[] seq;
      double soglia = 2.5;
      seq = leggiSequenza();
      stampaSequenza(seq);

      int numElementiInCode = contaEsterni(seq, soglia);
      System.out.println("L'array contiene " + numElementiInCode +
                         " elementi al di fuori dell'intervallo di tolleranza");
   }
}
```

il metodo contaEsterni, quando è invocato sui parametri formali ***s*** e **γ**, deve restituire il numero di elementi nella sequenza ***s*** che sono esterni all'intervallo di ampiezza $\mathbf{2\gamma\sigma}$, centrato sulla media aritmetica degli elementi della sequenza, dove σ è la deviazione standard degli elementi della sequenza.

$$m = \frac{1}{n}\sum_{i=1}^{n} s_i \qquad \sigma = \sqrt{\frac{\sum_{i=1}^{n}(s_i - m)^2}{n}} \qquad \text{intervallo} = [m - \gamma\sigma; m + \gamma\sigma]$$

2) (8 punti) **Descrivere l'algoritmo di ricerca binaria, mostrandone anche la codifica in Java. Dimostrare la complessità computazionale dell'algoritmo.**

3) (6 punti) **Si discuta il concetto di *parametro* di un metodo Java, e la maniera in cui vengono utilizzati i parametri specificati nella chiamata.**

4) (6 punti) **Si descriva il concetto di *costruttore* nella programmazione a oggetti**