

Esercizi aggiuntivi su tipi di dato, stringhe e uso di metodi statici

(Fondamenti di Informatica 1 – Walter Didimo)

Soluzioni

Esercizio 1 Dire cosa visualizza il seguente frammento di codice. Verificare poi la correttezza della risposta, scrivendo ed eseguendo un programma che lo realizza.

```
boolean a = 10 > 2;
System.out.println ("a = " + a);
boolean b = !a && a;
boolean c = !a || a;
System.out.println ("b && c = " + (b && c));
boolean d = Math.abs(-10) <= 10 && Math.abs(-10) >=10;
System.out.println ("d = " + d);
boolean e = !((a && b) || (!c && d));
System.out.println ("!((a && b) || (!c && d)) = " + e);
```

Soluzione

La visualizzazione appare come segue:

```
a = true
b && c = false
d = true
!((a && b) || (!c && d)) = true
```

Esercizio 2 Scrivere un programma che fa inserire all'utente tre numeri reali e che visualizza all'utente true se essi sono stati inseriti in ordine crescente di valore e false altrimenti.

Soluzione

```
class Esercizio2{
    public static void main (String[] args){
        InputWindow in = new InputWindow ();
        OutputWindow out = new OutputWindow ();
        double a = in.readDouble ("Inserire primo numero");
        double b = in.readDouble ("Inserire secondo numero");
        double c = in.readDouble ("Inserire terzo numero");
        boolean risultato = (a<b) && (b<c);
        out.writeln (risultato);
    }
}
```

Esercizio 3 Scrivere un programma che fa inserire all'utente una stringa str e che visualizza true se str contiene tutte le 5 vocali dell'alfabeto italiano e false altrimenti.

Soluzione

```
class Esercizio3{
    public static void main (String[] args){
        InputWindow in = new InputWindow ();
        String str = in.readString ("Inserire una stringa");
        String s = str.toLowerCase();
        boolean risultato = (s.indexOf("a") >=0) &&
            (s.indexOf("e") >=0) &&
            (s.indexOf("i") >=0) &&
            (s.indexOf("o") >=0) &&
            (s.indexOf("u") >=0);
        System.out.println (risultato);
    }
}
```

Esercizio 4 Dire cosa visualizza il seguente frammento di codice. Verificare poi la correttezza della risposta, scrivendo ed eseguendo un programma che lo realizza.

```
Integer i1 = new Integer (10);
Integer i2 = new Integer (10);
System.out.println (i1 == i2);
System.out.println (i1.equals(i2));
System.out.println (i1.intValue()==i2.intValue());
```

Soluzione

La visualizzazione appare come segue:

```
false
true
true
```

Esercizio 5 La classe Retta (che viene fornita già implementata), permette di creare oggetti che rappresentano rette nel piano. Un oggetto di tipo Retta è caratterizzato da due punti, ciascuno specificabile con una coppia di coordinate cartesiane. E' possibile creare un oggetto Retta attraverso il seguente costruttore:

- `Retta (double p1x, double p1y, double p2x, double p2y);`

dove la retta rappresentata dall'oggetto è quella che passa per i punti $p1 = (p1x,p1y)$, $p2 = (p2x,p2y)$. Inoltre, un oggetto Retta dispone del seguente metodo:

- `boolean equals (Retta r);`

che restituisce true se l'oggetto Retta ricevente rappresenta la stessa retta dell'oggetto Retta r passato come parametro.

Scrivere un semplice programma Java che:

1. Acquisisce dall'utente tre punti p1, p2 e p3.

2. Verifica se i tre punti sono collineari (cioè se giacciono su una stessa retta) oppure no; nel primo caso visualizza all'utente il messaggio "punti collineari: true", mentre nel secondo caso visualizza all'utente il messaggio "punti collineari: false".

Soluzione

Il programma deve verificare se tre punti sono collineari utilizzando le funzionalità offerte dalla classe Retta. Una soluzione possibile consiste nel definire due oggetti Retta, uno che rappresenta la retta per i punti p1 e p2, e l'altro che rappresenta la retta per i punti p1 e p3. A questo punto, se le due rette create sono la stessa retta, vuol dire che i punti p1, p2 e p3 sono collineari. Questa ultima verifica si può fare con il metodo equals della classe Retta. Ecco di seguito il codice di una classe che (insieme alla classe Retta) realizza il programma richiesto.

```
class PuntiCollineari{
    public static void main (String[] args){
        InputWindow in = new InputWindow ();
        OutputWindow out = new OutputWindow ();

        // inserimento p1, p2 e p3

        double p1x,p2x,p3x,ply,p2y,p3y;

        p1x = in.readDouble ("p1: coordinata X");
        ply = in.readDouble ("p1: coordinata Y");

        p2x = in.readDouble ("p2: coordinata X");
        p2y = in.readDouble ("p2: coordinata Y");

        p3x = in.readDouble ("p3: coordinata X");
        p3y = in.readDouble ("p3: coordinata Y");

        // creazione della retta per p1 e p2
        // e della retta per p1 e p3

        Retta r12 = new Retta (p1x,ply,p2x,p2y);
        Retta r13 = new Retta (p1x,ply,p3x,p3y);

        // verifica collinearità di p1, p2 e p3

        out.writeln ("punti collineari: " + r12.equals(r13));
    }
}
```