

# Programmazione 1, Laboratorio 5

Claudio Ferretti

(novembre 2014)

1. Riprendere il main() e il metodo inverti() dell'Esercizio 1 del Laboratorio 4, per usare al loro interno l'istruzione:

```
System.println("s->" + System.identityHashCode(s))
```

Nel main() inserire tale istruzione per ogni stringa s passata come argomento a inverti(), e per la stringa restituita da inverti().

Nel metodo inverti() fare la stessa cosa per ogni stringa ricevuta come argomento, e per la stringa da ritornare come valore restituito dal metodo.

Ad esempio, metodi originali:

---

```
public static void main(String[] args){
    String sPassata;
    //... seguire specifiche dell'esercizio originale ...
    String sRicevuta = inverti(sPassata);
    //...
}
public static String inverti(String sInput){
    String sOutput;
    //...
    return sOutput;
}
```

---

metodi richiesti ora:

---

```
public static void main(String[] args){
    String sPassata;
    //... seguire specifiche dell'esercizio originale ...
    System.out.println("sPassata->" + System.identityHashCode(sPassata));
}
```

```

        String sRicevuta = inverti(sPassata);
        System.out.println("sRicevuta->" + System.identityHashCode(sRicevuta));
        //...
    }
    public static String inverti(String sInput){
        System.out.println("sInput->" + System.identityHashCode(sInput));
        String sOutput;
        //...
        System.out.println("sOutput->" + System.identityHashCode(sOutput));
        return sOutput;
    }

```

---

Saper spiegare i nuovi messaggi a video:

il metodo `System.identityHashCode(s)` produce, per la stringa `s`, un numero che indica in quale area di memoria si trovano i caratteri della stringa (“riferimento”).

2. Scrivere, ed invocare opportunamente dal `main()`, un metodo `alterna()` che:
  - riceve come parametri un intero positivo `n`, due stringhe `sA` e `sB` e un booleano `primaB`,
  - restituisce una stringa prodotta concatenando in modo alternato `sA` e `sB` per `n` volte, partendo da `sB` solo se l’ultimo parametro riceve valore `true`.

Esempio: `alterna(2, "ab", "xy", false) -> "abxyabxy"`

3. Riprendere l’esercizio precedente, e modificare il programma in modo che se `primaB==true` la stringa `sA` venga inserita nel risultato con i suoi caratteri invertiti rispetto all’originale.

N.B. E’ opportuno usare il metodo `inverti()` del primo esercizio del precedente laboratorio.

Esempio: `alterna(2, "ab", "xy", true) -> "xybaxyba"`

4. Scrivere un programma che:
  - (a) nel `main`, chiede all’utente una stringa `sIn`, poi
  - (b) offre all’utente la scelta tra le opzioni `'A'` e `'B'`
  - (c) se viene scelto `'A'`: viene stampato il numero di vocali presenti in `sIn`

- (d) altrimenti viene stampato il numero di caratteri in sIn che NON sono vocali
- (e) in ENTRAMBI in casi, si stampano poi i caratteri presenti in sIn ma escludendo gli spazi

Si richiede di chiudere le operazioni indicate negli ultimi tre punti in tre distinti metodi, invocandoli poi in modo opportuno.

5. Scrivere, ed invocare opportunamente dal main(), un metodo checkSeq() che:

riceve una stringa sIn,

restituisce un valore booleano, che sarà true solo se: le vocali presenti in sIn sono in sequenza non decrescente.

Esempio: "rtp/xw"->true, "a,bd.eg/ei"->true, "a,bd.ig/ei"->false

6. Scrivere un programma che chiede all'utente due numeri n e m, e poi visualizza una stringa che rappresenta un panorama montano con n cime ampie m, precedute e separate da valli ampie m, in questo modo:

con n==2, m==1: \_/'\\_'/'\

con n==3, m==4: \_\_\_\_/''''\\_\_\_\_\_/''''\\_\_\_\_\_/''''\

Farlo creando e invocando dei semplici metodi cima(dim) e valle(dim), che ricevono un intero e restituiscono una stringa,

ad esempio: cima(2)->"'", valle(3)->"\_\_\_".