

Programmazione e Algoritmi 1 - Prova pratica

A.A. 2024/25 — Compito del 17/1/2025 – Versione Java

prof. Gianluca Amato

Esercizio 1 (8 punti)

Scrivere un programma che prende in input da terminale due numeri interi n ed m e disegna una griglia formata da $n \times m$ rettangoli, ognuno formato da 5 spazi (vedere esempio sotto). Al centro di ogni rettangolo deve essere scritto un numero progressivo che parte da 1 e si incrementa di 1 a ogni rettangolo, secondo l'ordine consueto di lettura. Quello che segue è un esempio d'interazione con il programma. In rosso trovate quello scritto dall'utente, in nero l'output del programma. I bordi della griglia (realizzati con il carattere meno e la barra verticale) sono obbligatori.

```
Inserisci n (numero righe): 3
Inserisci m (numero colonne): 4
```

```
-----
| 1 | 2 | 3 | 4 |
-----
| 5 | 6 | 7 | 8 |
-----
| 9 | 10 | 11 | 12 |
-----
```

Esercizio 2 (6 punti)

Scrivere un metodo statico `caratteri_in_comune` che prende in input due stringhe e restituisce come risultato il numero di caratteri che le due stringhe hanno in comune. Ad esempio, se la funzione è chiamata con gli argomenti `"ciccio"` e `"orco"`, il risultato è 2, in quanto le due stringhe hanno in comune i caratteri `"c"` e `"o"`.

Notare che il risultato non dipende da quante volte un carattere è ripetuto nella prima o nella seconda stringa. Si ricordi che per accedere al carattere in posizione i della stringa s si può usare il metodo `s.charAt(i)`.

Esercizio 3 (3 punti)

Scrivere alcuni test con il framework `jUnit` per verificare il corretto funzionamento di `caratteri_in_comune` dell'esercizio precedente. Si considerino i seguenti casi:

- l'esempio fornito nell'esercizio;
- almeno un esempio in cui una delle due stringhe è vuota;
- almeno un esempio in cui le due stringhe sono non vuote ma non hanno caratteri in comune.

Programmazione e Algoritmi 1 - Prova scritta

A.A. 2024/25 — Compito del 17/1/2025 – Versione Java

prof. Gianluca Amato

Esercizio 4 (5 punti)

Scrivere un metodo statico `mascheraVettore` che prende come parametro un array di interi `m` e un valore intero `v`. Il metodo deve restituire un nuovo array che ha le stesse dimensioni di `m` e che contiene `true` nelle posizioni contenenti il valore `v`, e `false` altrimenti. L'array `m` non deve essere modificato.

Esercizio 5 (5 punti)

Spiegare la differenza esistente tra linguaggi a basso e ad alto livello, e tra interpreti e compilatori.

Esercizio 6 (7 punti)

Si consideri il seguente codice Java, e se ne scriva la traccia di esecuzione negli appositi moduli.

```
1  public class Prog {
2      public static String fun(String l, int n, String r) {
3          if (n == l.length())
4              return r;
5          else if (n % 2 == 0)
6              return fun(l, n + 1, l.charAt(n) + r);
7          else
8              return fun(l, n + 1, r);
9      }
10
11     public static void main(String[] args) {
12         String x = fun("ciao", 0, "");
13         System.out.println(x);
14     }
15 }
```

Si ricordi che `l.charAt(n)` restituisce il carattere in posizione `n` della stringa `l`.