

**Modulo di Laboratorio di Programmazione I (CdL Informatica) &  
Corso di Laboratorio di Programmazione (CdL Matematica)  
a.a. 2007-2008**

Prova Totale - 30 Gennaio 2008

NOME

COGNOME

MATRICOLA

CORSO DI LAUREA

Scrivere in stampatello **Nome**, **Cognome** e **Matricola** su ogni foglio consegnato.  
Si ricorda di **riconsegnare il testo del compito** insieme al proprio elaborato scritto.

**Nota Bene:** Le soluzioni degli esercizi **non** devono contenere alcun metodo **main** né alcuna istruzione di stampa né istruzioni **switch**, **break**, **continue**, etc. La soluzione per l'Esercizio 3 **non** deve contenere alcuna istruzione di ciclo (**for**, **while**, etc.). La non osservanza di tali requisiti comporterà l'annullamento della prova scritta dello studente.

**Esercizio 1.** Si consideri il seguente frammento di codice in Java:

```
String[] a = b;
int i = 0, j = 0; boolean p = false;
for (i = 0; i < a.length; i++) {
    j = a[i].length()-1; p = false;
    while (j > 0 && !p) {
        if (a[i].charAt(j) == a[i].charAt(j-1))
            p = true;
        j--;
    }
}
```

Determinare il valore finale delle variabili  $i$ ,  $j$  e  $p$  per i seguenti valori di  $b$ :

- 1)  $b = \{ "grrr", "kjyy", "abaabb", "p" \};$
- 2)  $b = \{ "qps", "well", "ggghch" \}.$

**Esercizio 2.** Scrivere un metodo statico iterativo che, dato un array bidimensionale  $a$  di caratteri, restituisce un array monodimensionale  $b$  di stringhe tale che  $b[i]$  è la stringa ottenuta concatenando i caratteri nella riga  $a[i]$  da sinistra a destra. Ad esempio, dato  $a = \{ \{ 'a', 'k', 'w' \}, \{ 'j', 'q', 'p', 'b' \}, \{ 'z', 'x' \} \}$ , il metodo restituisce l'array  $b = \{ "akw", "jqpb", "zx" \}.$

Quali modifiche occorre apportare alla soluzione proposta affinché l'array  $b$  restituito sia tale che ogni  $b[i]$  sia la stringa ottenuta concatenando i caratteri nella riga  $a[i]$  da destra a sinistra?

**Esercizio 3.** Scrivere una versione ricorsiva del metodo definito per risolvere l'Esercizio 2.

**Esercizio 4.** Un *bambino* può essere caratterizzato da nome, cognome, età e sesso. Scrivere una classe **Bambino** con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Inoltre, definire un metodo per modificare l'età di un bambino ed un metodo che restituisce una stringa che descrive un bambino.

**Esercizio 5.** Un *alunno* è un bambino caratterizzato anche dal nome della scuola che frequenta, la classe, la sezione e dall'aver o meno intolleranze alimentari. Scrivere una classe `Alunno` con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Inoltre, definire un (unico) metodo per cambiare il nome della scuola, la classe e la sezione frequentata, ed un metodo che restituisce una stringa che descrive un alunno.

**Esercizio 6.** Un *campo estivo* può essere caratterizzato tramite il nome, l'indirizzo, il nome del responsabile e l'elenco dei bambini iscritti al campo estivo. Scrivere una classe `CampoEstivo`, il cui costruttore imposta il numero massimo di bambini che possono essere iscritti ad un campo estivo. Oltre ai metodi che restituiscono i valori delle variabili istanza, definire i seguenti metodi:

- un metodo che, dato un bambino  $b$  ed un intero  $k$ , aggiunge  $b$  nell'elenco degli iscritti ad un campo estivo, a condizione che l'età di  $b$  sia maggiore di  $k$ ;
- un metodo che, dati due interi  $m$  ed  $n$  (con  $m < n$ ), restituisce l'elenco di tutti i bambini iscritti ad un campo estivo la cui età è compresa tra  $m$  ed  $n$ ;
- un metodo che, data una stringa  $s$ , restituisce l'elenco dei bambini iscritti ad un campo estivo che sono alunni della scuola il cui nome è denotato da  $s$ .

**Esercizio 7.** Siano date le seguenti classi:

```
class A {
    private String s;

    public A (String t) {
        this.s = t;
    }
    public String leggiS() {
        return this.s;
    }
    public String m (A a) {
        if (this.s.length()==a.s.length())
            return this.s + a.s;
        else
            return a.s + this.s;
    }
}

class B extends A {
    private int n;

    public B (String t, int k) {
        super(t);
        this.n = k;
    }
    public int leggiN() {
        return this.n;
    }
    public String m (A a) {
        return super.m(a) + this.n;
    }
}
```

Dati gli oggetti seguenti:

```
B obj1 = new B ("firenze",7);
A obj2 = new A ("venezia");
A obj3 = obj1;
```

determinare qual è il valore restituito dalle seguenti chiamate di metodo:

- 1) `obj1.m(obj2)`;
- 2) `obj2.m(obj3)`;
- 3) `obj3.m(obj1)`.