

**Laboratorio di Programmazione I (CdL Informatica) &
Laboratorio di Programmazione (CdL Matematica)
a.a. 2005-2006**

Prova Totale - 14 Dicembre 2005

NOME

COGNOME

MATRICOLA

CORSO DI LAUREA

Nota Bene: Le soluzioni degli esercizi **non** devono contenere alcun metodo **main** né alcuna istruzione di stampa né istruzioni di **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, r = 0; boolean p = false;
for (i = 0; i < a.length; i++) {
    j = 0; p = false;
    while (j < a[i].length && !p) {
        if (a[i][j].equals(s)) {
            r++;
            if (r >= k) p = true;
        }
        j++;
    }
    r = 0;
}
```

Determinare il valore finale delle variabili **i**, **j**, **r** e **p** per i seguenti valori di **b**, **s** e **k**:

- 1) **b** = {{"ab", "abc", "ab"}, {"ab", "ab", "kkr"}, {"ccc", "ab"}}, **s** = "ab" e **k** = 2;
- 2) **b** = {{"g", "ff", "rmn", "g"}, {"rp", "g", "g", "pr", "rpg"}}, **s** = "g" e **k** = 2.

Esercizio 2. Scrivere un metodo che, dati un array bidimensionale *a* di stringhe ed un intero $k > 0$, restituisce un array monodimensionale di stringhe *b* tale che $b[i]$ è la prima stringa di $a[i]$, la cui lunghezza è un multiplo di *k*. Se in $a[i]$ non esistono stringhe la cui lunghezza è un multiplo di *k*, allora $b[i]$ è la stringa "*". Ad esempio, dati $a = \{\{"treno", "tram", "bus"\}, \{"scuola", "natale"\}, \{"uno", "bis", "ter"\}\}$ e $k = 2$, il metodo restituisce l'array $b = \{"tram", "scuola", "*"\}$.

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

Esercizio 4. La *presentazione della tesi di un laureando* davanti ad una commissione di laurea può essere descritta tramite il nome e la matricola del laureando, il titolo della tesi, il nome del relatore della tesi ed il tempo a disposizione per la presentazione. Scrivere una classe **Presentazione** con gli opportuni costruttori ed i metodi che restituiscono i valori delle variabili istanza. Inoltre, definire un metodo per modificare il tempo a disposizione ed un metodo che restituisce una stringa che descrive la presentazione di una tesi.

Esercizio 5. La *presentazione della tesi di un laureando in Scienze dell'Informazione* è caratterizzata anche dal titolo di una tesina, dal nome del relatore della tesina e dal tempo a disposizione per la presentazione della tesina. Scrivere una classe **PresentazioneScInf** con gli opportuni costruttori ed i metodi che restituiscono i valori delle variabili istanza. Infine, definire un metodo che restituisce il tempo a disposizione per la presentazione di tesi e tesina, ed un metodo che restituisce una stringa che descrive la presentazione di una tesi in Scienze dell'Informazione.

Esercizio 6. Una *sessione di laurea* può essere caratterizzata dal tipo di sessione (Es. invernale), dall'anno (Es. 2005), dal nome del presidente della commissione di laurea e dall'elenco delle presentazioni. Scrivere una classe **SessioneLaurea**, il cui costruttore imposta il numero massimo delle presentazioni. Oltre ai metodi che restituiscono i valori delle variabili istanza, definire i seguenti metodi:

- un metodo che aggiunge una presentazione all'elenco di una sessione;
- un metodo che scambia due presentazioni in una sessione;
- un metodo che restituisce *true* se nella sessione esiste almeno una presentazione di una tesi di cui è relatore il presidente della commissione di laurea.

Esercizio 7. Siano date le seguenti classi:

```
class A {
    private String s;

    public A (String t) {
        this.s = t;
    }
    public String leggiNome() {
        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 c;

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

Dati gli oggetti seguenti:

```
A a1 = new A ("alba");
A a2 = new B ("per",7);
B b1 = new B ("lillo",5);
```

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

- 1) a1.m(b1);
- 2) a2.m(a1);
- 3) b1.m(a2).