

Modulo di Laboratorio di Programmazione I (CdL Informatica) &
Corso di Laboratorio di Programmazione (disattivato - CdL Matematica)

a.a. 2010-2011

Prova Totale - 25 Febbraio 2011

NOME

COGNOME

MATRICOLA

CORSO DI LAUREA

Scrivere in stampatello **Nome**, **Cognome** e **Matricola** su ogni foglio consegnato.

Nota Bene: Il testo del compito deve essere riconsegnato insieme al proprio elaborato scritto. 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 = true; String s = "";
for (i = 0; i < a.length; i++) {
    j = 1; p = true;
    while (j < a[i].length && p) {
        if (a[i][j-1].charAt(0) == a[i][j].charAt(0))
            j++;
        else
            p = false;
    }
    if (!p)
        s = a[i][j];
}
```

Determinare il valore finale delle variabili *i*, *j*, *p* ed *s* nei seguenti casi:

- 1) $b = \{ \{ "pfg", "p", "pn" \}, \{ "bw", "bxc", "bbbb" \} \}$;
- 2) $b = \{ \{ "abc", "ak", "azz" \}, \{ "gg" \}, \{ "c", "cy" \}, \{ "zbt", "z", "yy", "zhi" \} \}$.

Esercizio 2. Scrivere un metodo statico iterativo che, dati due array monodimensionali *a* e *b* di stringhe, restituisce un array bidimensionale *c* di stringhe tale che ogni riga $c[i]$ contiene le stringhe ottenute concatenando i caratteri di $a[i]$ (letti da sinistra a destra) con la stringa $b[i]$. Esempio: se $a = \{ "ab", "bcd", "e", "fhgs" \}$ e $b = \{ "rp", "k", "uxy", "st" \}$, il metodo restituisce $c = \{ \{ "arp", "brp" \}, \{ "bk", "ck", "dk" \}, \{ "euxy" \}, \{ "fst", "hst", "gst", "sst" \} \}$. Nota: si assuma che le lunghezze di *a* e *b* siano uguali.

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

Esercizio 4. Una *squadra* può essere caratterizzata da nome, città, sport praticato, nome del proprietario, anno di fondazione e nome del campionato a cui è iscritta. Scrivere una classe **Squadra** con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Inoltre, definire un metodo per modificare il nome del proprietario, un

metodo che, date due squadre, restituisce quella “piú antica” ed un metodo che restituisce una stringa che descrive una squadra.

Esercizio 5. Una *squadra con penalizzazione* è una squadra caratterizzata anche dai punti di penalizzazione e dal motivo della penalizzazione. Scrivere una classe `SquadraP` con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Inoltre, definire un metodo per modificare i punti di penalizzazione, un metodo che, date due squadre con penalizzazione, restituisce quella con una maggiore penalizzazione ed un metodo che restituisce una stringa che descrive una squadra con penalizzazione.

Esercizio 6. Un *campionato* può essere caratterizzato tramite nome, anno, numero di partite e l’elenco delle squadre partecipanti. Scrivere una classe `Campionato`, il cui costruttore imposta il numero massimo di squadre partecipanti. Oltre ai metodi che restituiscono i valori delle variabili istanza, definire i seguenti metodi:

- un metodo che, data una squadra *s*, aggiunge *s* nell’elenco delle squadre partecipanti ad un campionato, a patto che il nome del campionato coincida con quello del campionato a cui è iscritta *s*;
- un metodo che, data una squadra *s*, controlla se *s* partecipa ad un campionato;
- un metodo che, dato un intero *k*, restituisce l’elenco delle squadre di un campionato che hanno almeno *k* punti di penalizzazione;
- un metodo che, data una stringa *s*, restituisce l’elenco delle squadre partecipanti ad un campionato che provengono dalla città *s*.

Esercizio 7. Siano date le seguenti classi:

```
class A {
    private String top;

    public A (String s) {
        this.top = s;
    }
    public String getTop() {
        return this.top;
    }
    public String m (A a) {
        return this.top + a.top;
    }
}
```

```
class B extends A {
    private int id;

    public B (String s, int k) {
        super(s);
        this.id = k;
    }
    public int getId() {
        return this.id;
    }
    public String m (A a) {
        if (this.id > a.getTop().length())
            return this.id + super.m(a);
        else
            return super.m(a) + this.id;
    }
}
```

Dato il seguente frammento di codice:

```
A x = new B ("semestre",12);
A y = new A ("febbraio");
B z = new B ("primo",5);
String s1 = y.m(z), s2 = x.m(y), s3 = z.m(new A ("marzo"));
```

determinare il valore delle variabili `s1`, `s2` ed `s3`.