

Modulo di Laboratorio di Programmazione I (CdL Informatica)

a.a. 2009-2010

Prova Totale - 15 Febbraio 2010

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:

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

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

- 1) $b = \{\{4, -4\}, \{-3\}, \{7, 1, -1, 5\}\}$;
- 2) $b = \{\{3, -7, 7, 1\}, \{-2, 5, 0, 0, 11\}\}$.

Esercizio 2. Scrivere un metodo statico iterativo che, dato un array monodimensionale a di stringhe, restituisce un array bidimensionale b di caratteri tale che $b[i]$ è un array contenente i caratteri della stringa $a[i]$ (letti da sinistra a destra). Se $a[i]$ è la stringa vuota, l'array $b[i]$ è vuoto. Esempio: dato $a = \{"abc", "fg", "lnpop"\}$, il metodo restituisce l'array $b = \{\{'a', 'b', 'c'\}, \{'f', 'g'\}, \{'l', 'n', 'p', 'o', 'p'\}\}$. **N.B.** Si richiede che gli unici metodi della classe **String** usati nella soluzione dell'Es.2 e dell'Es.3 siano **length** e **charAt**.

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

Esercizio 4. Un *articolo di abbigliamento* può essere caratterizzato da nome, taglia, nazione in cui è stato fatto, prezzo e numero di capi disponibili. Scrivere una classe **ArticoloA** con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Inoltre,

definire un metodo per modificare il prezzo, un metodo `equals` che verifica se due articoli sono uguali ed un metodo che restituisce una stringa che descrive un articolo di abbigliamento.

Esercizio 5. Un *articolo di abbigliamento in saldo* è un articolo di abbigliamento caratterizzato anche dalla percentuale di sconto e dal numero di giorni a disposizione per un eventuale cambio. Scrivere una classe `ArticoloInSaldo` con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Inoltre, definire un metodo per modificare il numero di giorni per il cambio, un metodo che restituisce il prezzo scontato di un articolo in saldo ed un metodo che restituisce una stringa che descrive un articolo in saldo.

Esercizio 6. Un *negozio di abbigliamento* può essere caratterizzato tramite nome, indirizzo, nome del proprietario e l'elenco degli articoli di abbigliamento trattati. Scrivere una classe `NegozioA`, il cui costruttore imposta il numero massimo di articoli trattati da un negozio. Oltre ai metodi che restituiscono i valori delle variabili istanza, definire i seguenti metodi:

- un metodo che, dato un articolo di abbigliamento *a*, aggiunge *a* nell'elenco degli articoli di un negozio;
- un metodo che, dato un intero *p*, restituisce l'elenco degli articoli in saldo aventi una percentuale di sconto maggiore o uguale a *p*;
- un metodo che, data una stringa *s*, restituisce l'elenco di tutti gli articoli *Made in s*.

Esercizio 7. Siano date le seguenti classi:

```
class C {
    private String sx;

    public C (String t) {
        this.sx = t;
    }
    public String leggiSX() {
        return this.sx;
    }
    public int m (C c) {
        if (this.sx.equals(c.sx))
            return 2*this.sx.length();
        else
            return
                this.sx.length()+c.sx.length();
    }
}

class D extends C {
    private String dx;

    public D (String t, String s) {
        super(t);
        this.dx = s;
    }
    public String leggiDX() {
        return this.dx;
    }
    public int m (C c) {
        if (this.dx.length() >
            c.leggiSX().length())
            return super.m(c)+this.dx.length();
        else
            return super.m(c)-this.dx.length();
    }
}
```

Dato il seguente frammento di codice:

```
D d1 = new D ("alfa","beta");
C d2 = new C ("ALFA");
int n = d2.m(d1);
d2 = new D ("BETA","gamma");
int v = d1.m(d2);
int p = d2.m(d1);
```

determinare qual è il valore delle variabili *n*, *v* e *p*.