

LABORATORIO DI PROGRAMMAZIONE 2003-2004

Recupero - 14 Luglio 2004

NOME

COGNOME

MATRICOLA

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

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

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

Determinare il valore finale delle variabili i, j, n e p per i seguenti valori dell'array b:

- 1) {{"aba", "ac", "pqr", "afga", "akk"}, {"oops", "opla", "oh"}};
- 2) {"bye", "hi", "hello"}, {"help", "aid", "aiuto"}, {"spqr", "sport", "copy"}}.

Esercizio 2. Si consideri il seguente frammento di codice in Java:

```
class Exc1 extends Exception { }

public class C {
    public static int metodo (int[] a, int k) throws Exc1 {
        try {
            if (k < a.length-1) {
                if (a[k] < a[k+1]) return metodo(a,k+1);
                else throw new Exc1();
            }
            else return a.length;
        }
        catch (Exc1 e) {
            if (a[k] == a[k+1]) return k;
            else throw e;
        }
    }
}
```

Determinare il valore restituito dalle seguenti chiamate:

- 1) `C.metodo(a,0)` dove `a` è l'array `{3,7,8,11}`;
- 2) `C.metodo(a,0)` dove `a` è l'array `{3,7,5,9,9}`;
- 3) `C.metodo(a,1)` dove `a` è l'array `{-1,3,4,4,7}`.

Esercizio 3. Scrivere un metodo che, dati due array `a` e `b` di interi, restituisce *true* se tutti gli elementi dell'array `b` compaiono nell'array `a` nello stesso ordine in cui compaiono in `b`, altrimenti il metodo restituisce *false*. Ad esempio, dati gli array `a = {-5,4,7,-1,10,21,9,-7}` e `b = {4,-1,9,-7}`, il metodo restituisce *true*. Il metodo deve sollevare opportune eccezioni nel caso in cui la lunghezza dell'array `a` non sia maggiore o uguale a quella di `b`.

Esercizio 4. Scrivere una versione ricorsiva del metodo definito per risolvere l'Esercizio 3.

Esercizio 5. Un *bonbon* è un dolcetto che può essere caratterizzato tramite il nome (Es. Bacio), la marca (Es. Perugina), il formato (Es. mini), il peso (Es. 10gr) e se è incartato o meno. Scrivere una classe `Bonbon` con gli opportuni costruttori (ed opportune eccezioni in caso di parametri numerici non corretti) ed i metodi che restituiscono i valori delle variabili istanza. Inoltre, definire un metodo `etichetta` che restituisce una stringa con le informazioni relative ad un bonbon, ad esempio "Bacio Perugina, formato mini, peso 10gr, incartato".

Esercizio 6. Una caramella alla frutta è un bonbon caratterizzato dal gusto (Es. limone) e dal tipo (Es. morbida). Un cioccolatino è un bonbon caratterizzato dal tipo di cioccolato (Es. fondente) e dall'eventuale ripieno (Es. nocciola, liquore, etc.). Scrivere due classi `CaramellaFrutta` e `Cioccolatino` con gli opportuni costruttori ed i metodi che restituiscono i valori delle variabili istanza. Infine, definire un metodo che restituisca una stringa contenente tutte le informazioni relative ad una caramella alla frutta e ad un cioccolatino rispettivamente. Esempio: "Gelée Sperlari, formato a spicchio, peso 8gr, non incartata, gusto limone, morbida".

Esercizio 7. Un contenitore di bonbon può essere caratterizzato dalla forma (Es. barattolo), il tipo (Es. latta), la capacità (Es. 500gr) e l'insieme dei bonbon contenuti nel recipiente. Scrivere una classe `Contenitore`, il cui costruttore imposta il numero massimo di bonbon che possono esservi contenuti (con le opportune eccezioni in caso di parametri numerici non corretti). Oltre ai metodi che restituiscono i valori delle variabili istanza, definire i seguenti metodi:

- un metodo che calcola il peso totale dei bonbon presenti in un contenitore;
- un metodo che restituisce *true* se esistono almeno `k` bonbon di una data marca `m` in un contenitore;
- un metodo che restituisce l'elenco dei bonbon incartati di un dato formato `f`.