

# Modulo di Laboratorio di Programmazione (CdL Informatica)

a.a. 2021-2022

Prova Totale - 10 Febbraio 2022

NOME

COGNOME

MATRICOLA

**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**, **try-catch**, 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; char[] c = d;
int i = 0, j = 0; boolean p = true; String s = "+";
for (i = 0; i < a.length; i++) {
    j = 0; p = false;
    while (j < a[i].length() && !p) {
        if (a[i].charAt(j) != c[i]) {
            s = s + a[i].charAt(j) + s;
            p = !p;
        }
        j++;
    }
}
```

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

- 1)  $b = \{ "ff", "hfxhx", "yywx" \}$  e  $d = \{ 'f', 'x', 'y' \}$ ;
- 2)  $b = \{ "a", "ggkn", "", "ww" \}$  e  $d = \{ 'z', 'g', 'v', 'w' \}$ .

**Esercizio 2.** Scrivere un metodo *statico* iterativo che, dati un array bidimensionale di stringhe *a* ed un array monodimensionale di interi *c*, restituisce un array monodimensionale di booleani *b* tale che  $b[i]$  vale *true* se nella riga  $a[i]$  esistono *almeno*  $c[i]$  coppie di stringhe adiacenti che cominciano con lo stesso carattere, altrimenti  $b[i]$  vale *false*.

Esempio: se  $a = \{ \{ "ac", "k2", "krs", "km", "kw", "z" \}, \{ "bg", "rmn", "rsa", "rw", "r" \}, \{ "lpq", "mR", "mlp" \} \}$  e  $c = \{ 3, 2, 2 \}$ , il metodo restituisce l'array  $b = \{ true, true, false \}$ . Si assuma che tutte le stringhe di *a* siano non vuote.

**N.B.** i) Il metodo deve sollevare un'opportuna eccezione se le lunghezze di *a* e *c* non sono uguali. ii) Occorre passare alla riga successiva *non appena* si trovano in  $a[i]$  le  $c[i]$  coppie di stringhe che soddisfano la condizione richiesta.

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

**Esercizio 4.** Un *treno* può essere caratterizzato tramite numero, tipologia (e.g. IC, Regionale, etc.), distanza percorsa (espressa in numero di km), nome della stazione di partenza e nome della stazione di arrivo. Scrivere una classe **Treno** con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Definire inoltre un metodo che modifica la tipologia; un metodo che, dati due treni, verifica se hanno la stessa tipologia e la

stessa stazione di partenza; un metodo che, dati due treni, restituisce quello con la distanza minore. Definire infine un metodo che restituisce una stringa che descrive un oggetto della classe **Treno**.

**Esercizio 5.** Un *treno notte* è un treno caratterizzato anche dal numero di vagoni letto e dalla tipologia delle cuccette. Scrivere una classe **TrenoNotte** con un opportuno costruttore ed i metodi che restituiscono i valori delle variabili istanza. Definire inoltre un metodo che, dati due treni notte, verifica se hanno la stessa tipologia di cuccette ed arrivano nella stessa stazione, ed un metodo che, dati due treni notte, restituisce quello con il numero di vagoni letto maggiore. Definire infine un metodo che restituisce una stringa che descrive un oggetto della classe **TrenoNotte**.

**Esercizio 6.** Una *stazione ferroviaria* può essere caratterizzata tramite il nome, la città in cui si trova, l'indirizzo e l'elenco dei treni che vi transitano. Scrivere una classe **Stazione**, il cui costruttore imposta il numero massimo di treni che possono transitare in una stazione. Oltre ai metodi che restituiscono i valori delle variabili istanza, definire i seguenti metodi:

- un metodo che, dato un treno  $t$ , aggiunge  $t$  nell'elenco dei treni che transitano in una stazione (il metodo deve sollevare un'opportuna eccezione in caso di elenco pieno);
- un metodo che, date due stringhe  $x$  ed  $s$ , restituisce l'elenco dei treni in transito in una stazione che hanno tipologia  $x$  e stazione di partenza  $s$ ;
- un metodo che, dato un intero  $k$ , restituisce l'elenco dei treni notte che transitano in una stazione aventi un numero di vagoni letto maggiore di  $k$ .

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

```
class A {
    private int n;

    public A (int n) {
        this.n = n;
    }
    public int getN() {
        return this.n;
    }
    public String m (A a) {
        if (this.n <= a.n)
            return this.n + "*" + a.n;
        else
            return this.n + "--" + a.n;
    }
}

class B extends A {
    private String str;

    public B (int n, String t) {
        super(n);
        this.str = t;
    }
    public String getStr() {
        return this.str;
    }
    public String m (A a) {
        if (a.getN() >= this.getN())
            return super.m(a) + this.str;
        else
            return this.str + super.m(a);
    }
}
```

Dato il seguente frammento di codice:

```
A x = new A (8), y = new B (7,"febbraio");
B z = new B (4,"esame");
String s = x.m(y), t = z.m(x), u = y.m(z);
```

determinare il valore delle variabili  $s$ ,  $t$  ed  $u$ .