

# Esame di Programmazione Internet

## CdL in Informatica per il Management

**Data Appello:**

**Tempo a Disposizione: 2h**

### INFORMAZIONI PRELIMINARI

L'esame consiste nello scrivere uno o più file \*.java compilabili. Elaborati che **non compilano il codice prodotto non** verranno presi in considerazione.

NON è consentito l'uso di testi e appunti.

La scelta dei nomi dei file, se non espressamente indicati, non è rilevante. I file possono essere salvati in directory diverse.

All'inizio di **ogni** file .java mettete come commento, Nome, Cognome, Num Matricola. Se **non** lo farete il compito sarà annullato. Esempio:

```
/*****  
* Nome: Paolino  
* Cognome: Paperino  
* Num Matricola: 12345678  
* email: paperino@unibo.it  
*****/
```

Se il testo richiede effettuare lettura di file testuali, il candidato deve crearsene alcuni di esempio da solo (usando jEdit come editor e salvando il file con estensione .txt), **rispettando le specifiche date.**

E' **consentito** invocare qualunque metodo delle API di Java.

La prova intende valutare, oltre alla verifica che lo studente ha acquisito **competenze** nell'ambito della programmazione, la capacità del candidato di **comprendere** il testo del compito (i.e., programmi che girano ma **non** fanno quanto richiesto verranno considerati insufficienti), la capacità di inserire **commenti** al codice per facilitare la comprensione dell'elaborato, la capacità di scrivere **codice pulito e ordinato**, la capacità di **ottimizzare** la soluzione (soprattutto in termini di efficienza del programma).

### NOTE TECNICHE

Nel caso (frequente) vi si dovesse bloccare il sistema di sviluppo (ex: per loop infinito), potete aprire un terminale (CTRL-ALT-t) e provare scrivere il comando "pkill java" o "pkill jedit". Se si dovesse chiudere jEdit (che e` quello che dovrebbe capitare), poi scrivere "jedit &" e il sistema dovrebbe ripartire.

### NOTA LOGISTICA

Agli studenti affetti da disturbi quali dislessia, è richiesto di completare due esercizi a scelta tra quelli proposti. Il tempo di svolgimento resta lo stesso concesso agli altri.

**Nota per lo studente:** Per sicurezza, nei commenti iniziali si scriva "**studente con DSA**".

## TESTO DEL COMPITO

### Esercizio 1

Creare un programma che legga una serie di valori numerici da tastiera (non necessariamente interi). L'inserimento termina quando viene inserito il valore 0.

Il programma dovrà calcolare e stampare la media dei quadrati di tutti i numeri inseriti. Ad esempio, se vengono inseriti 1, 2.5, -3, 0 il programma dovrà stampare  $(1^2 + (2.5)^2 + (-3)^2) / 3 = 5.416666666666667$

### Esercizio 2

Creare un programma che inizializzi un array X con N numeri interi casuali tra 0 e 9, dove N è una costante inizializzata con un valore a piacere.

Quindi creare una matrice M (array bidimensionale) N x N, con le seguenti proprietà:

- prima riga: è identica all'array X
- righe successive: ogni elemento è il risultato della somma tra i 3 seguenti elementi:
  - l'elemento che sta nella stessa colonna e riga precedente (l'elemento "sopra"), e
  - l'elemento che sta sulla riga precedente e colonna precedente (elemento "in alto a sinistra", se esiste), e
  - l'elemento che sta sulla riga precedente e colonna successiva (elemento "in alto a destra", se esiste).

Ad esempio, con N = 3 e X = {0, 9, 4} la matrice risultante è:

```
M = { { 0, 9, 4 },  
      { 9, 13, 13 },  
      { 22, 35, 26 } }
```

### Esercizio 3

Definire opportunamente in Java la seguente situazione. Uno smartphone è caratterizzato da un identificativo del modello (stringa), un prezzo (intero per semplicità) e una marca, che può assumere esclusivamente i valori "Apple", "Samsung", "Oppo", "Huawei". Tutte le proprietà sono necessarie per descrivere uno smartphone e immutabili.

Scrivere il codice di una classe `GestoreSmartphone` che espone un metodo che prende in input un vettore di smartphone e stampa a video il numero di smartphone per ogni marca. I dettagli della stampa non sono rilevanti.

Suggerimento: il metodo `.values()` su una Enumeration restituisce un vettore con i possibili valori.

Aggiungere alla classe `GestoreSmartphone` un main per testare il funzionamento dei metodi.

### Esercizio 4

Definire un metodo ricorsivo che prende in input un vettore di numeri interi e restituisce *true* se sono tutti uguali o *false* in caso contrario.

E' ammesso fare overloading del metodo.