


Esame di Programmazione Internet

CdL in Informatica per il Management

Tempo a Disposizione: 1h30 secondo parziale (solo esercizio 4) – 3h esame completo

Leggete BENE tutte le istruzioni PRIMA di contattare il docente. NON scrivere su questo foglio e RESTITUIRLO sempre a fine compito (anche se vi ritirate)

- Prendete posto e **loggatevi su EOL** all'indirizzo **http://eol.unibo.it**. Cliccate sull'appello con la data odierna. Se non riuscite a connettervi, contattate il docente
- Cliccate sul tasto **"Windows"**  della tastiera, appariranno diverse opzioni:
 - **Nautilus**: esplora file
 - **Firefox/Chromium**: browser
 - **jEdit, editor java**
 - **gedit**: editor di testo

Aprire **jEdit** (non gedit) e **attendete istruzioni** dal docente. Se non si apre, contattate il docente. **NOTA: Spesso capita che jEdit sembra "scompare" dallo schermo. In questo caso premere ALT+TAB: per passare da un'applicazione all'altra e tornare a jEdit**

- Quando tutti sono pronti, il docente **carica il testo dell'esame su EOL** e dà il via all'esame. Il tempo indicato è **tassativo**: EOL registra l'orario di consegna, eventuali **ritardi comporteranno penalizzazioni o annullamento del compito**
- L'esame consiste nella scrittura di codice ***.java compilabile**. Si consiglia di **salvare e compilare spesso** (anche se l'esercizio non è terminato). Si può fare copia-incolla del testo come commento, in modo da non dover aprire ogni volta il pdf con il testo dell'esercizio
- **NOTA: Nel caso si dovesse bloccare il sistema di sviluppo (es. loop infinito) aprite il terminale (CTRL+ALT+t) e digitate "pkill java" o "pkill jedit". Si dovrebbe chiudere jEdit: a questo punto scrivete "jedit &" e ripartite.**
- Verranno valutate, oltre alle **competenze** di programmazione, la capacità del candidato di **comprendere il testo** del compito (ad es., programmi che girano ma **non** fanno quanto richiesto sono considerati insufficienti) e di scrivere **codice pulito, ordinato e ottimizzato**.
- Nel caso di **esame parziale**, se svolgete un esercizio che non dovete svolgere tale esercizio verrà **annullato**
- Una volta terminato, fare **l'upload** solamente dei **file *.java su EOL**. **Non consegnare file *.class e nominare i file in modo chiaro (i numeri degli esercizi devono corrispondere, evitare spazi nel nome)**. Si consiglia di eseguire **l'upload almeno 5 minuti prima del termine del compito**. **Eventuali .class senza corrispondenti .java non verranno presi in considerazione!**

NON è consentito l'uso di testi, appunti cartacei e altre risorse on-line.

STUDENTI DSA:

- Gli studenti DSA che svolgono il **primo esame parziale** devono svolgere **tutti e 4 gli esercizi**, ma hanno a disposizione **3 ore** anziché 2. Quelli che svolgono il **secondo esame parziale** hanno a disposizione tutte le **3 ore**
- Gli studenti DSA che svolgono l'**esame completo** hanno a disposizione **3 ore** ma **non** devono svolgere l'**esercizio 1**.

TESTO DEL COMPITO

Esercizio 1

Leggere da tastiera delle stringhe fino a che viene inserita una stringa di lunghezza maggiore di 10 (da non considerare). Quindi stampare tutte le stringhe che contengono un numero di vocali maggiore della media. Ad esempio, se leggo le stringhe:

CIAO Marti

!?!

a

Successivamente

La media è $(5+0+1)/3 = 2$, quindi va stampata solo la stringa:

Ciao Marti

perché contiene $5 > 2$ vocali.

Esercizio 2

Caricare e stampare una matrice $N \times M$, con N e M letti da tastiera, contenente valori casuali tra -50 e 50 (inclusi). Quindi stampare tutte le righe della matrice per cui la somma degli elementi è minore del primo elemento della riga. Ad esempio, se $N = 3$, $M = 3$ e la matrice è:

2 -10 5

-1 1 8

7 0 -40

Vanno stampate la 1a riga ($2-10+5 < 2$) e la 3a riga ($7+0-40 < 7$) ma non la 2a ($-1+1+8 \geq -1$).

Esercizio 3

Definire una funzione ricorsiva che prende un array A di interi e restituisce la somma degli elementi dispari elevati al quadrato. Se non ci sono elementi dispari deve ritornare 0. E' possibile sovraccaricare la funzione ma non utilizzare funzioni ausiliarie.

Ad esempio, se $A = [2, 80, -3, 1, 60]$ deve ritornare $(-3)^2 + 1^2 = 10$

Esercizio 4

Definire in Java classi e interfacce utili per gestire la classifica dei 10 goal più belli del mondiale di calcio Qatar 2022. Si suppone l'esistenza di una votazione pubblica per cui ad ogni goal è associato il numero di voti ottenuti.

Per ogni goal si memorizza l'autore, una descrizione testuale e il numero di voti. Per ogni calciatore si memorizza il nome (stringa) e il ruolo che può essere *Portiere*, *Difensore*, *Centrocampista* o *Attaccante*.

Definire la classe QatarTop10 che rappresenta la classifica ed espone:

- un costruttore che inizializza la classifica inizialmente vuota
- un metodo per restituire il goal nella posizione i-esima (1 corrisponde al primo in classifica, 2 al secondo e così via); restituisce null se lo slot è vuoto
- un metodo per aggiungere un goal in classifica: il goal viene aggiunto nella posizione corretta in base al numero di voti ottenuti; se il numero di voti non è sufficiente per entrare in classifica non viene aggiunto; restituisce la posizione del goal oppure -1 se non è stato inserito
- un metodo che stampa un riassunto dei goal, evidenziando il numero di goal per ogni ruolo (cioè numero di goal segnati da portieri, difensori, centrocampisti e attaccanti). Il formato della stampa non è rilevante.

Implementare una classe di test che crea alcuni dati di prova (non è richiesto farli inserire all'utente) e mostra il corretto funzionamento dei metodi precedenti.