

Corso di Programmazione 1

Terza Esercitazione di Laboratorio

Esercizio 1

Scrivere un programma Java che legga da tastiera due numeri interi, corrispondenti a mese e anno, e stampi a video di quanti giorni è formato quel mese di quell'anno. Il numero corrispondente al mese deve essere compreso tra 1 e 12, e il numero corrispondente all'anno deve essere compreso tra 1900 e 2099; in caso contrario, stampare un messaggio d'errore e terminare l'esecuzione del programma (usando il metodo `exit()`).

Si tenga anche conto del fatto che l'anno può essere bisestile. Si ricorda che un anno è bisestile se è divisibile per 4 e non è divisibile per 100; tuttavia, gli anni divisibili per 400 sono invece bisestili. Quindi, ad esempio, il 1996 è stato bisestile (è divisibile per 4 ma non per 100, e non è divisibile per 400), il 1900 no (è divisibile per 4 ma anche per 100, mentre invece non è divisibile per 400), il 2000 sì (è divisibile per 400).

Esercizio 2

Scrivere un programma Java che chieda all'utente di inserire una frase tramite tastiera e, utilizzando gli opportuni metodi sulle stringhe, esegua le seguenti operazioni:

- se la frase contiene 2 o più parole:
 - stampi a video che la frase contiene due o più parole;
 - stampi a video la prima parola e la sua lunghezza;
 - stampi a video l'ultima parola e la sua lunghezza;
- se la frase contiene una sola parola:
 - stampi a video che la frase contiene una sola parola;
 - stampi a video la parola e la sua lunghezza;
- se la frase inserita è vuota:
 - stampi a video che la frase non contiene parole.

Per semplicità, si assuma che la frase non inizi e non finisca con degli spazi, e che ogni parola sia separata dalla successiva usando un solo spazio. Grazie a questa assunzione, le parole sono individuabili semplicemente cercando gli spazi.

Esercizio 3 (Voto all'esame)

Scrivere un programma Java che calcoli il risultato di uno studente all'esame di Programmazione 1. Il programma leggerà da tastiera:

- il voto ottenuto nella parte di teoria (da -8 a +8 punti);
- il voto ottenuto nella prova pratica (da 0 a 24 punti).

Memorizzati questi dati, procederà al calcolo del risultato in trentesimi, procedendo come segue. Il risultato finale è la somma dei risultati. L'algoritmo di calcolo, però, prevede una protezione "anti-smanettone", per cui se il voto finale è maggiore o uguale a 18 ma il voto della prova teorica è inferiore a 1 lo studente sarà comunque bocciato. Infatti:

- se il voto di teoria è minore o uguale a zero, e la somma dei voti di teoria e pratica è maggiore o uguale a 18, il programma stampa: "Bocciato, devi studiare di più la teoria";
- se il voto di teoria è minore o uguale a zero, e il voto di pratica è minore di 18, il programma stampa: "Bocciato, devi studiare ed esercitarti di più";
- se il voto di teoria è maggiore di zero, e la somma dei voti di teoria e pratica è minore di 18, il programma stampa: "Bocciato: devi esercitarti di più";
- se la somma dei voti di teoria e pratica vale 31 o 32, il programma stampa: "Congratulazioni: 30 e lode!".

In tutti gli altri casi il programma stampa "Il tuo voto è ", seguito dal totale derivante dalla somma dei punti di teoria e pratica.

Nota: questo **non** è l'algoritmo usato da noi per determinare veramente il voto all'esame!

Esercizio 4 (Gioco della morra cinese a due giocatori)

Scrivere un programma Java che simuli il gioco della morra cinese. Il programma dovrà prendere in ingresso due nomi con cui identificherà i due giocatori. I nomi verranno memorizzati in variabili di tipo `String`. Per ogni giocata, viene chiesto ad entrambi i giocatori di inserire una parola tra "forbice", "carta" o "pietra"; il programma dovrà stampare a video il nome del vincitore. Se la parola in ingresso non è una delle tre previste, il programma dovrà avvisare che è stato commesso un errore. Si ricordi che:

- forbice vince su carta;
- carta vince su pietra;
- pietra vince su forbice.

Nel caso in cui entrambi i giocatori scelgono la stessa stringa, la giocata è pari.

Terminata una giocata, il programma dovrà chiedere se si intende farne un'altra. In caso negativo, terminerà immediatamente l'esecuzione (senza usare il metodo `exit()`); in caso affermativo, chiederà nuovamente ad entrambi i giocatori di inserire una parola tra "forbice", "carta" o "pietra", stamperà a video il nome del vincitore, e così via.

Esercizio 5 (Gioco della morra cinese contro il computer)

Scrivere un programma Java che simuli il gioco della morra cinese, questa volta per un solo giocatore che giochi contro il computer. Il computer deciderà se scegliere "forbice", "carta" o "pietra" generando un numero casuale da 0 a 2 mediante l'espressione:

```
(int) (Math.random() * 3)
```

In particolare:

- se il numero estratto è 0 il computer sceglierà forbice;
- se il numero estratto è 1 il computer sceglierà carta;
- se il numero estratto è 2 il computer sceglierà pietra.

Una volta deciso il proprio simbolo, il computer domanderà all'utente di inserire il suo e dichiarerà quindi il vincitore. Chiederà poi se si intende fare un'altra giocata (come nell'esercizio precedente), e terminerà quando l'utente risponderà di no.

Esercizio 6

Scrivere un programma Java che implementi una semplice calcolatrice. Il programma deve visualizzare un menù per la scelta del tipo di operazione da effettuare:

- + per sommare
- per sottrarre
- * per moltiplicare
- / per dividere
- u per uscire dal programma

Deve poi chiedere in input il tipo di operazione che si vuole effettuare. Se l'utente inserisce il carattere 'u' allora il programma termina (senza usare il metodo `exit()`); se inserisce un carattere diverso da +, -, *, /, u, stampa un messaggio d'errore e mostra nuovamente il menù. Se il carattere inserito è +, -, *, oppure /, chiede all'utente due numeri interi tra cui eseguire l'operazione specificata; in caso di divisione, verificare che l'utente non intenda dividere per 0. Eseguita l'operazione indicata, il programma stampa a video il risultato e ripropone all'utente il menù.