

Autovalutazione – Python – 1 Strutture Dati Complesse

Introduzione

Gli esercizi proposti in questo foglio costituiscono uno strumento per valutare la propria preparazione. La soluzione dell'esercizio sarà discussa in classe assieme ai docenti.

Indicazioni importanti

- La descrizione dei singoli esercizi è preceduta da una parte comune, leggetela attentamente
- Gli esercizi vi richiederanno di implementare uno script python

Esercizi: introduzione

Supponete di avere precaricata in uno script una struttura dati complessa come nell'esempio seguente:

```
li=[  
    ("31/8/08", "Udinese", "Palermo", 3, 1 ),  
    ("31/8/08", "Sampdoria", "Inter", 1, 1 ),  
    ("31/8/08", "Cagliari", "Lazio", 1, 4 ),  
    ...  
]
```

Una struttura dati precaricata è una struttura dati a vostra disposizione in uno script, senza che dobbiate preoccuparvi di eseguire l'input dei dati. Per esempio, in un file .py potreste trovare (già scritte da altri) le istruzioni che inizializzano la lista *li* dell'esempio qua sopra.

Il contenuto della struttura dati *li* sarà chiamata **Dataset Serie A** d'ora in poi.

La struttura dati contiene tutti i risultati delle partite di calcio della stagione 2008-09 della serie A di calcio italiana. Ogni tupla della lista contiene le informazioni sul risultato di una partita. Ogni tupla è formata da 5 elementi: la data della partita, il nome della squadra che gioca in casa, il nome della squadra in trasferta, il numero di gol segnati dalla squadra di casa, il numero di gol fatti dalla squadra in trasferta. Tenete conto che la vittoria attribuisce 3 punti alla squadra vincente, il pareggio 1 punto a entrambe le squadre e la sconfitta 0 punti alla squadra perdente.

Se nel testo viene specificato un range di valori per un parametro formale, ad esempio se troverete scritto che il parametro *squadra* potrà assumere i valori ['Inter', 'Milan', 'Roma', 'Juventus', 'Napoli', ...] non occorre controllare che i valori siano passati correttamente, cioè non occorre controllare la presenza di errori tipo 'IIIInter' o 'MilanO'.

Per maggiori informazioni e per ottenere dei punteggi da confrontare con i risultati dei vostri script potete far riferimento al sito web: http://it.wikipedia.org/wiki/Serie_A_2008-2009.

A1. Serie A. Esercizio puntisquadra.

Nello script calcioA1.py contenente il Dataset Serie A precaricato, implementate il seguente programma.

Il programma deve richiedere in input all'utente una stringa contenente il nome di una squadra.

Il programma deve visualizzare a video il totale dei punti che la squadra ha conseguito alla fine del campionato. Per esempio la squadra "Udinese" ha conseguito 58 punti alla fine del campionato. Opzionale: controllate che la stringa inserita dall'utente corrisponda al nome della squadra così come è memorizzato nella struttura dati precaricata.

A2. Serie A. Esercizio statisticheSquadra.

Nello script calcioA2.py contenente il Dataset Serie A precaricato, implementate il seguente programma.

Il programma deve richiedere in input all'utente una stringa contenente il nome di una squadra.

Il programma deve stampare a video: il numero delle vittorie, il numero dei pareggi e il numero delle sconfitte. Per esempio, considerando la squadra "Udinese" dovranno essere visualizzati i seguenti dati: n. vittorie 16, n. pareggi 12, n. sconfitte 10 n. pareggi 10, n. sconfitte 12.

A3. Serie A. Classifica.

Nello script calcioA3.py contenente il Dataset Serie A precaricato, implementate il seguente programma.

Il programma NON deve richiedere informazioni in input e deve visualizzare a video un dizionario formato da diverse coppie chiave valore. In ogni coppia, la chiave deve essere una stringa che contiene il nome della squadra, mentre il valore associato alla chiave deve essere un intero che rappresenta il totale dei punti che la squadra ha conseguito alla fine del campionato. Come chiavi dovranno essere usate le stesse stringhe che nella lista precaricata contengono i nomi delle squadre (se nella lista precaricata appare “Udinese”, usate quella stringa come chiave, non usate altre stringhe del tipo “Squadra udinese” oppure “Udinese FC” ...). Come prima attività all’interno del programma vi suggeriamo di individuare l’elenco delle squadre che appariranno come chiavi del dizionario restituito.

Fase di test.

Negli esercizi proposti, avete avuto a che fare con liste o dizionari contenenti un numero elevato di dati. Per semplificare la fase di test del vostro codice, all’inizio non lavorate direttamente su strutture dati di grosse dimensioni, ma utilizzando i commenti su più righe (cioè i tripli apici o le triple virgolette) escludete porzioni di dati in modo da lavorare su una struttura dati di dimensioni ridotte (per esempio con solo 4 o 5 elementi). Una volta che il vostro programma inizierà a funzionare correttamente con le strutture dati “ridotte”, levate i commenti e lavorate sui dati originali.

Prestate attenzione perché i valori contenuti nelle liste e nei dizionari potranno essere a loro volta elementi complessi come per esempio delle tuple.

Buon lavoro

I docenti del corso