

Compito d'esame -- 21 luglio 2020 -- Compito A

Istruzioni (leggere attentamente)

Nota importante: la mancata osservanza delle seguenti regole può comportare la perdita di informazioni necessarie alla valutazione della prova d'esame.

Svolgimento degli esercizi: Per ogni esercizio avete una cartella `EsercN` che contiene un file dal nome `A_ExN.py` (dove **N** è il numero dell'esercizio) con lo scheletro della soluzione. Questo file incorpora un codice di test che proverà la vostra soluzione per un certo numero di possibili dati in input. Aprite il file con IDLE e modificate SOLO il contenuto della funzione. Eseguendo il file `.py` si otterrà il responso dei test sulla console. Non spostate i file dalla loro posizione e non create nuovi file. **Si noti che per la correzione verranno usati insieme di dati di test diversi.** E' possibile consultare la documentazione ufficiale del linguaggio Python, ma **non è possibile usare libri o appunti.**

Esercizi

- **A_Ex1(squadre, punti)** Scrivere una funzione che riceve in ingresso due liste, **squadre** e **punti**, aventi la stessa lunghezza, tali che **squadre** contenga delle stringhe che corrispondono a nomi di squadre, e **punti** contenga degli interi non negativi che indicano il punteggio in classifica di ciascuna squadra, in modo che l'intero in posizione *i* in **punti** si riferisca al punteggio della squadra in posizione *i* di **squadre**. La funzione deve restituire una lista contenente i nomi delle squadre in **squadre** in ordine di classifica sulla base dei punti indicati in **punti**. Si assuma che **squadre** non contenga mai più di una occorrenza di una stessa stringa (cioè una squadra occorre una sola volta nell'elenco), e che (per semplicità) **punti** non contenga mai più di una occorrenza dello stesso intero (cioè non ci sono squadre a pari punti). Se **squadre** e **punti** sono liste vuote, la funzione deve restituire una lista vuota. Ad esempio, se **squadre** = ['Roma', 'Lazio', 'Inter', 'Juve', 'Napoli'] e **punti** = [12, 5, 8, 10, 13], la funzione deve restituire ['Napoli', 'Roma', 'Juve', 'Inter', 'Lazio'].
- **A_Ex2(m1,m2)** Scrivere una funzione che riceve in ingresso due matrici *quadrate* **m1** ed **m2** rappresentate come liste di liste e tali che **m1** ed **m2** hanno la stesse dimensioni e contengono numeri interi e non siano mai vuote. La funzione deve restituire una matrice delle stesse dimensioni di **m1** ed **m2** che contenga in posizione *i,j* il maggiore fra gli elementi in posizione *j,i* in **m1** ed **m2**. Ad esempio se **m1** ed **m2** rappresentano rispettivamente le matrici:

$$\begin{pmatrix} 1 & 2 & 3 \\ 7 & 1 & 3 \\ 1 & 0 & 2 \end{pmatrix} \quad \begin{pmatrix} 5 & 4 & 6 \\ 0 & 3 & 9 \\ 1 & 2 & 5 \end{pmatrix}$$

La funzione deve restituire la rappresentazione tramite lista di liste della matrice

$$\begin{pmatrix} 5 & 7 & 1 \\ 4 & 3 & 2 \\ 6 & 9 & 5 \end{pmatrix}$$

- **A_Ex3(file, c1, c2)** Si considerino solo file di testo in cui sono presenti solo:
 1. caratteri alfanumerici maiuscoli o minuscoli (senza lettere accentate),
 2. spazi bianchi,
 3. caratteri di fine riga ('\n'),

Scrivere una funzione che prende in ingresso un **file** di tale forma e due caratteri **c1** e **c2** e restituisce l'insieme delle parole che compaiono in **file** in cui è presente almeno una sottostringa (eventualmente coincidente con la stringa stessa) in cui **c1** precede **c2** e **c1** e **c2** sono separati da un numero pari (anche zero) di caratteri (non necessariamente diversi da **c1** e **c2**). La funzione deve distinguere fra lettere maiuscole e minuscole. Ad esempio, se **file** contiene il testo:

Questa associazione si chiama arco acronimo di associazione romana collegi

c1='a', e **c2**='o', la funzione deve restituire {'associazione', 'arco', 'acronimo'}

- **A_Ex(file1,file2)** Scrivere una funzione che calcola le vendite ed il guadagno di vari venditori. Nel file **file1**, in formato csv, sono riportati gli oggetti ed il loro prezzo minimo di vendita dei vari proprietari nel seguente formato:

Proprietario, oggetto, prezzo_minimo

Il **file2** riporta tutte le offerte nell'ordine in cui vengono fatte nel formato:

Acquirente, oggetto, prezzo_offerto

Gli oggetti sono tutti diversi tra di loro ed ogni offerta viene accettata se il prezzo offerto è superiore od uguale a quello minimo. Quando però un oggetto viene comprato, può essere rivenduto ad un prezzo minimo di 1 in più di quello a cui è stato comprato. Quindi se io compro un oggetto a 100, poi lo rivendo solo se arriva un'offerta di almeno 101.

La funzione deve calcolare un dizionario con chiave il proprietario e valore una lista composta da: numero oggetti acquistati (negativo se ne ha venduti più che acquistati) e soldi incassati (negativo se ha speso più di quanto ha incassato). Il dizionario deve contenere solo proprietari hanno venduto o acquistato qualcosa. Si noti che un acquirente che ha effettivamente acquistato qualcosa diventa un proprietario e deve comparire nel dizionario in output.

Ad esempio, se il file1 è:

Paolo,Libro,20
Paolo,Anello,25
Marta,Cane,18
Marta,Sedia,10
Sofia,Divano,30

Ed il file2 è:

Francesco,Libro,21
Gianni,Libro,25
Marta,Anello,20

Allora il risultato deve essere: {'Paolo': [-1, 21], 'Francesco': [0, 4], 'Gianni': [1, -25]}, in quanto Paolo ha venduto il Libro a 21, Francesco lo ha comprato a 21 e poi rivenduto a 25 e Gianni lo ha comprato a 25. Marta non ha comprato l'Anello poiché la sua offerta era inferiore al prezzo minimo per l'oggetto