

Compito d'esame -- 21 gennaio 2019 -- Compito C

Istruzioni (leggere attentamente)

Nota importante: la mancata osservanza delle seguenti regole può comportare la perdita di informazioni necessarie alla valutazione dell'esercitazione.

Registrazione dei dati dello studente: PRIMA DI INIZIARE, eseguite (con un doppio click sull'icona) il programma `RegistraStudente` che si trova nella cartella `Esame`. Inserite (separatamente) *Numero di Matricola*, *Cognome* e *Nome* seguendo le istruzioni che compaiono sul terminale, e confermate i dati che avete inserito. Il programma genera il file `studente.txt` che contiene Matricola, Cognome e Nome su tre righe separate (nell'ordine indicato). Il file `studente.txt` non deve essere modificato manualmente. Verificate che i dati nel file `studente.txt` siano corretti. In caso di errore potete rieseguire il programma `RegistraStudente`.

Svolgimento degli esercizi: Per ogni esercizio avete una cartella `EsercN` che contiene un file dal nome `C_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 Spyder 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**. In caso di problemi tecnici chiedere ai docenti o ai tecnici del laboratorio.

Al termine dello svolgimento del compito d'esame **NON è necessario effettuare una procedura di consegna** dell'elaborato. Le vostre soluzioni saranno copiate direttamente dalla cartella `Esame` della macchina su cui avete lavorato. Quindi, per concludere la prova, è sufficiente chiudere il programma Spyder e lasciare la vostra postazione.

Esercizi

- **C_Ex1(s1,s2)** Scrivere una funzione che riceve in ingresso due stringhe **s1** e **s2** NON VUOTE, e restituisce la stringa ottenuta alternando le lettere di **s1**, partendo dall'inizio della stringa, con quelle di **s2**, partendo dalla fine della stringa. Se **s1** è più lunga di **s2** (oppure **s2** è più lunga di **s1**) la sottostringa rimanente in **s1** (oppure in **s2**) deve essere concatenata alla fine della stringa in output. Ad esempio, se **s1** = "abcd" e **s2** = "xyefgh", la funzione deve restituire "ahbgcfdexy" (attenzione, la parte eccedente di **s2**, "xy", non viene rovesciata). Se invece **s1** = "a" e **s2** = "b" la funzione deve restituire "ab".
- **C_Ex2(m):** Scrivere una funzione che prende in ingresso una matrice QUADRATA di numeri interi positivi **m**, rappresentata come lista di liste, e restituisce la somma dei valori sulla diagonale principale SE LA MATRICE E' DIAGONALE, -1 altrimenti. Una matrice è diagonale se contiene valori diversi da 0 solo sulla diagonale principale.

Ad esempio, data la matrice

0	0	0
0	5	0
0	0	1

La funzione deve restituire 6.

Invece, data la matrice

0	0	0
1	5	0
0	0	-1

La funzione deve restituire -1.

- **C_Ex3(l1,l2)** Scrivere una funzione che riceve in ingresso le liste **l1** ed **l2**, tali che **l1** ed **l2** siano entrambe non vuote, contengano esclusivamente stringhe ed abbiano lo stesso numero di elementi. La funzione deve restituire la lista ottenuta da **l1** aggiungendo dopo ciascuna stringa di **l1** un nuovo elemento che indichi il numero di caratteri che tale stringa ha in comune con la stringa nella posizione corrispondente in **l2** (facendo distinzione fra maiuscole e minuscole). Ad esempio, se `l1 = ["mamma", "asso", "re", "pluto"]`, ed `l2 = ["mare", "lago", "fante", "barba"]`, la funzione deve restituire `["mamma", 2, "asso", 2, "re", 1, "pluto", 0]` ("asso" e "lago" hanno 2 caratteri in comune, "a" ed "o").
- **C_Ex4(file):** Una pizzeria tiene conto delle varie comande fatte dai vari tavoli tramite un **file** di testo in formato csv, le cui righe hanno il formato:

`N_tavolo,N_clienti,Portata,Quantità,Prezzo`

dove `N_tavolo` è il numero identificativo del tavolo; `N_clienti` è il numero clienti del tavolo; `Portata` è una stringa tra "antipasto", "pizza" e "bevanda"; `Quantità` è il numero di `Portata` ordinate; `Prezzo` è il prezzo (intero) unitario della `Portata`. Un tavolo può fare più di un ordine per la stessa portata, e potete assumere che per ogni ordine di un tavolo il numero di clienti sia sempre lo stesso. Il ristorante offre uno sconto di 1€ per ogni portata in più rispetto al numero dei clienti, cioè se vengono ordinati 7 antipasti e 6 bevande da un tavolo di 5 clienti, al totale bisogna togliere 3€ (cioè, 2€ per gli antipasti ed 1€ per le bevande).

Scrivere una funzione Python che dato il file di testo nel precedente formato restituisca un dizionario che ha per chiavi il numero del tavolo (**in formato intero**) e come valore il conto totale (**in formato intero**). Ad esempio, dato il file contenente

```
1,2,antipasto,2,3
2,1,pizza,1,5
1,2,antipasto,1,4
2,1,bevanda,2,1
```

la funzione deve restituire il dizionario `{1:9, 2:6}`.