

## Compito d'esame -- 27 gennaio 2022 -- Compito B

### Istruzioni (leggere attentamente)

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

**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`.

**Tempo a disposizione:** 1 ora e 30 minuti

Per risolvere gli esercizi in modo che possano essere successivamente corretti è **necessario scrivere la soluzione di ogni esercizio nel file .py relativo**, che trovate nella cartella dell'esercitazione (ad esempio, per l'esercizio 1 scrivete il vostro programma nel file `Ex1.py`, per l'esercizio 2, nel file `Ex2.py`, e così via). Notate che ogni file incorpora del codice python per eseguire alcuni test sulla funzione. **NON** modificate questo codice, ma **SCRIVETE SOLO il contenuto della funzione**. Non spostate i file dalla loro posizione e non create nuovi file. **Si noti che per la correzione verranno usati insieme i dati di test diversi**.

E' possibile consultare la documentazione ufficiale del linguaggio Python (<file:///Library/Frameworks/Python.framework/Versions/3.9/Resources/English.lproj/Documentation/index.html>),

ma **non è possibile usare libri o appunti**. In caso di problemi tecnici chiedere ai docenti o ai tecnici del laboratorio.

In ogni esercizio, se non diversamente richiesto, potete sempre assumere che gli input forniti siano coerenti con la traccia (ad esempio, se l'esercizio chiede di dare in input alla funzione una lista non vuota di stringhe, potete sempre assumere l'input sia in tale forma e non è necessario nel codice effettuare controlli per gestire casi diversi da questo, considerando, ad esempio, il caso di lista vuota).

Per gli esercizi relativi a lettura da file, la stringa in input che identifica il file è sempre comprensiva anche della sua estensione e il file risiede sempre nella stessa directory dell'esercizio.

### Esercizi

- **Ex1(I)** Scrivere una funzione che riceve in ingresso una lista `l` contenente stringhe e la lunghezza della *più lunga* sequenza in `l` di stringhe di lunghezza sempre *decrescente*. Ad esempio, se `l=['ora', 'aldo', 'gioca', 'una', 'carta', 'e', 'antonio', 'prende', 'tre', 'dani', 'ari']`, la funzione deve restituire 3 perchè tre è la lunghezza della più lunga sequenza in `l` avente la caratteristica richiesta ('antonio', 'prende', 'tre') e non c'è in `l` una sequenza di questo tipo che abbia lunghezza 4 o superiore.

- **Ex2 (file,ins)** Scrivere una funzione che riceve in ingresso il nome di un file di testo **file**, che può contenere solo caratteri alfanumerici, spazi ed andate a capo '\n', ed un insieme **ins** di parole (stringhe alfanumeriche) e restituisce un dizionario con chiavi le parole in **ins** che compaiono almeno una volta nel testo e valore associato a ciascuna chiave la lista *ordinata in ordine decrescente* degli indici di riga (contate a partire dall'indice 1) in cui compare la parola corrispondente alla chiave. Se ad esempio il file **file** contiene il seguente testo:

Marco possiede tre palloni  
i palloni devono essere gonfiati e devono essere provati  
A Marco piacciono i suoi palloni

e **ins**={'palloni', 'Marco', 'devono', 'viaggio'} allora la funzione deve restituire {'Marco': [3, 1], 'palloni': [3, 2, 1], 'devono': [2]}, poiché 'viaggio' non compare mai nel testo.

- **Ex3(file1,file2,n)** Scrivere una funzione che prende in ingresso due file di testo in formato csv ed un valore intero **n** compreso tra 1 e 365 che rappresenta un giorno dell'anno (assumiamo l'anno non sia bisestile). Il file **file1** contiene le informazioni sugli oggetti disponibili in un negozio online all'inizio di un anno e le sue informazioni includono il nome dell'oggetto, la quantità disponibile all'inizio dell'anno ed il prezzo. Il file csv **file1** ha il seguente formato:

Oggetto,Quantità,Prezzo

Il file **file2** contiene gli acquisti e le vendite effettuate dal negozio durante l'anno ed ha il seguente formato:

Giorno,Oggetto,Quantità

Dove il Giorno è il giorno in cui è avvenuto l'acquisto/vendita, indicato con un numero intero compreso tra 1 e 365, Oggetto è uno degli oggetti già visti nel file **file1** e Quantità è un valore (positivo o negativo) che indica la quantità acquistata (se positivo) o venduta (se negativo). Se di un oggetto ne viene venduta una quantità maggiore di quella disponibile in quel momento, assumete che ne venga inviata al cliente solo la parte disponibile. Il file **file2** è ordinato per giorno e potete assumere che non ci siano mai una vendita ed un acquisto dello stesso oggetto nello stesso giorno.

Il valore **n** rappresenta il giorno alla fine del quale vogliamo conoscere la situazione. La funzione deve restituire un dizionario con chiave gli oggetti che sono stati venduti o acquistati nel periodo di interesse (da inizio anno al giorno **n** incluso) e valore una lista di due interi di cui il primo indica il ricavo complessivo (numero di pezzi venduti ed inviati per quell'oggetto moltiplicati per il prezzo) ed il secondo il ricavo mancato (numero di pezzi venduti ma che non si è potuto inviare perché mancava la disponibilità moltiplicati per il prezzo), tutti calcolati alla fine del giorno **n**. Ad esempio, se il file **file1** contiene:

Iphone,10,500  
Ipad,6,700  
Ps5,5,500

il file **file2** contiene:

5,Ipad,-5  
5,Iphone,-12  
7,Ipad,-3  
11,Iphone,6  
13,Ps5,-7

ed **n** vale 10 allora la funzione deve restituire {'Ipad': [4200, 1400], 'Iphone': [5000, 1000]}.