

**Simulazione prova d'esame – 21 dicembre 2021**  
(Compito d'esame del 20 gennaio 2021 -- Compito C)

## Istruzioni (leggere attentamente)

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

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

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'esame (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 di 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 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(11,12)** Scrivere una funzione che riceve in ingresso due liste di stringhe **11** e **12** e restituisce l'insieme delle stringhe di **11** che sono anagramma di una stringa di **12** e le stringhe di **12** che sono anagramma di una stringa di **11**. Ad esempio, se la lista **11** vale ['arco', 'pippo', 'pluto', 'corda'] ed **12** vale ['palla', 'darco', 'ocra', 'casa'], la funzione deve restituire l'insieme {'arco', 'corda', 'darco', 'ocra'} (come al solito negli insiemi l'ordine non è rilevante).
- **Ex2 (i,file)** Scrivere una funzione che riceve in ingresso un insieme **i** contenente prefissi telefonici internazionali (senza il + iniziale o lo 00 iniziale) rappresentati come stringhe di cifre ed il nome di un file di testo **file** contenente al suo interno del testo che include anche dei numeri telefonici. I numeri telefonici hanno sempre una delle due forme (semplificate rispetto la realtà)

+39-344-12345  
0039-344-12345

In entrambe le forme i trattini separatori '-' sono opzionali (possono essere presenti o meno, ma solo nelle posizioni indicate) ed il numero di cifre segue sempre questo schema: **uno, due o tre** cifre per il prefisso internazionale, **tre cifre** per il prefisso locale, **cinque cifre** per il numero specifico.

La funzione deve leggere il file di testo e restituire un dizionario che ha come chiavi i prefissi telefonici presenti in **i** e come valore il numero (un intero) di numeri di telefono che usano tale prefisso. Ad esempio, se **i**={ '39', '351', '34' } e **file** contiene il seguente testo:

Ieri ho telefonato a marco al numero +39-34701234. Altre volte ho usato il numero 0039-213-44534  
quando era in Portogallo lo chiamavo al numero +35177711111  
mentre il numero 351-12-77445561 che mi era stato comunicato inizialmente era sbagliato  
Il numero di Anna in Francia è +33-123-12345

allora la funzione deve restituire { '39':2, '351':1, '34':0 }.

**Nota:** potete assumere che in **file** i numeri telefonici non siano mai preceduti o seguiti da una lettera o cifra, ma siano sempre seguiti e preceduti da un carattere NON alfanumerico (uno spazio, un punto, ecc.), e che due numeri telefonici in formato corretto siano sempre separati da almeno due caratteri. Inoltre assumete che un numero di telefono in **file** non compaia più di una volta.

- **Ex3(file1,file2)** Scrivere una funzione che prende in ingresso un file di testo **file1** in formato csv che contiene le informazioni sul menù offerto da un ristorante, un secondo file **file2** che contiene, in ordine di arrivo, le ordinazioni e calcola, per ogni cliente, il costo da pagare o segnala che non può completare l'ordine. Il file del menù **file1** ha il seguente formato:

NomePiatto,Costo,Disponibilità

dove NomePiatto identifica il piatto, Costo è il costo in euro e Disponibilità denota quanti piatti di quel tipo si possono preparare. Il file **file2** delle ordinazioni ha il seguente formato:

NomeCliente, NomePiatto\_1:quantità\_1,..., NomePiatto\_n:quantità\_n

Cioè per ogni riga contiene il nome (univoco) del cliente, seguito da n coppie (n non è fisso, ma può variare in ogni riga) con il NomePiatto seguito dal ':' seguito dalla quantità. La funzione deve calcolare un dizionario in cui le chiavi sono i nomi dei clienti (potete assumere che ogni cliente effettui uno ed un solo ordine) e valore il costo complessivo (un intero) da pagare, se l'ordine può essere completato, oppure la stringa 'respinto' se l'ordine contiene un piatto per cui non c'è abbastanza disponibilità. Se un ordine viene respinto, nessuno dei piatti ordinati viene preparato. Ad esempio, se il file **file1** contiene:

Carbonara,10,10  
PastaVongole,12,6  
Pollo,12,8

Ed il **file2** contiene:

Giorgio,Carbonara:2,PastaVongole:4  
Paola,Pollo:3,PastaVongole:3  
Francesca,Pollo:3,Carbonara:3

Allora la funzione deve restituire { 'Giorgio':68, 'Paola':'respinto', 'Francesca':66 } poiché a seguito dell'ordine di Giorgio sono rimaste solo 2 porzioni di PastaVongole e quindi l'ordine di Paola non può essere completato.