

## Esercitazione Python n. 8 -- 23 Novembre 2021

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 A\_Ex1.py, per l'esercizio 2, nel file A\_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**. In fase di correzione verranno eseguiti dei test diversi da quelli che trovate attualmente.

Per **consegnare l'esercitazione svolta**, comprimate la cartella LabPython08 (**FATE ATTENZIONE**, la cartella che comprimate deve contenere gli esercizi e **NON** un'ulteriore cartella con dentro gli esercizi) **in un file .zip** e caricatela sulla pagina del corso di <https://classroom.google.com/u/0/w/MzkwNTM3Njc3/t/all> (dalla sezione 'Esercitazioni su Python', selezionate 'Esercitazione 8' e successivamente 'Visualizza Compito'; poi cliccate su 'Aggiungi o crea' e scegliete il file da caricare). **NON** è necessario rinominare il file .zip. E' **NECESSARIO NON** rinominare i singoli file. Al termine dell'operazione cliccate su 'Contrassegna come completato'. La consegna deve avvenire in maniera inderogabile entro le **23:59 di Mercoledì 24 novembre**.

A valle dell'esercitazione provvederemo ad effettuare una correzione automatica delle soluzioni consegnate, il cui esito verrà comunicato ad ogni studente tramite un messaggio di posta privato.

Si rammenta che **le esercitazioni consegnate in ritardo o che non rispettano le indicazioni per la consegna saranno automaticamente valutate con punteggio 0**. In particolare, si ricorda di **comprimere e riconsegnare l'intera cartella dell'esercitazione**, e non singolarmente i file degli esercizi, di **NON usare formati di compressione diversi da .zip**, di **NON** rinominare i file o **metterli in sottocartelle**.

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 un intero positivo, potete assumere che l'input sia sempre un numero intero maggiore di zero, e non è necessario nel codice effettuare controlli per gestire casi diversi da questo).

### Esercizi

- **A\_Ex1(l)** Scrivere una funzione che prende in ingresso una lista **l** *non vuota* di stringhe e calcola quale carattere alfabetico minuscolo ('a'-'z') compare in più stringhe. Se ci sono più caratteri che compaiono lo stesso numero di volte si scelga quello alfabeticamente più grande. Ad esempio, se **l** = ['casa', 'senape', 'ketchup', 'pasta'], allora il carattere da restituire è 's' che compare in 3 stringhe ed è più grande di 'a' e 'p', che anche compaiono in 3 stringhe.
- **A\_Ex2(l)** Scrivere una funzione che prende in ingresso una lista **l** di stringhe e restituisce un insieme contenente tutti e soli i caratteri che appaiono almeno due volte in una delle stringhe. Ad esempio, se **l** = ['casa', 'albero', 'bello'], allora l'insieme da restituire sarà {'a', 'l'}. Ovviamente, se la lista **l** in ingresso è vuota, oppure non ci sono caratteri che appaiono almeno due volte in una stringa di **l**, la funzione deve restituire l'insieme vuoto.
- **A\_Ex3(l)** Scrivere una funzione che prende in ingresso una lista **l** di stringhe e restituisce un insieme di coppie (x1, x2) tali che x1 e x2 sono stringhe di **l** della stessa lunghezza e x1 è diverso da x2. Notare che una coppia è semplicemente una tupla di lunghezza due. In altri termini, l'insieme deve contenere tutte e sole le coppie formate combinando a due a due, in tutti i possibili modi, le stringhe di **l** che hanno la stessa lunghezza, escludendo i casi in cui una stringa si combina con se stessa. Ad esempio, se **l** = ['jkl', 'h', 'plqa', 'a', 'xkj'], allora l'insieme di coppie da restituire sarà {'(jkl','xkj)', ('h','a'), ('a','h'), ('xkj','jkl')}. Ovviamente se la lista **l** in ingresso è vuota, la funzione deve restituire l'insieme vuoto.

- **A\_Ex4(l)** Scrivere una funzione che prende in ingresso una lista **l** di stringhe e restituisce un insieme di tutte e sole le coppie (x,n) tali che x è una stringa di **l** ed n è il numero di volte che la stringa x appare in **l**. Ad esempio, se **l**=['jkl', 'h', 'plqa', 'jkl', 'h', 'xkj'], allora la funzione deve restituire l'insieme {( 'jkl',2), ('h',2), ('plqa',1), ('xkj',1)}. Ovviamente se la lista **l** in ingresso è vuota, la funzione deve restituire l'insieme vuoto.
- **A\_Ex5(a, b)** Scrivere una funzione che prende in ingresso due insiemi **a** e **b** di coppie (cioè tuple di dimensione due) tali che l'insieme **a** contiene le coppie (nome, cittanascita) (ogni coppia indica che quella persona è nata in quella città) e l'insieme **b** contiene le coppie (città, regione) (ogni coppia indica che quella città appartiene a quella regione). La funzione deve restituire un altro insieme contenente tutte e sole le coppie (nome, regione) che indicano che quella persona è nata in quella regione. Ad esempio, se **a**=({'Giovanni', 'Napoli'}, ('Marco', 'Roma'), ('Giuseppe', 'Rieti'), ('Aldo', 'Torino')) e **b**=({'Napoli', 'Campania'}, ('Benevento', 'Campania'), ('Roma', 'Lazio'), ('Rieti', 'Lazio'), ('Genova', 'Liguria'}), allora l'insieme da restituire sarà {( 'Giovanni', 'Campania'), ('Marco', 'Lazio'), ('Giuseppe', 'Lazio')}. Si assuma che **a** e **b** siano insiemi che contengono solo coppie i cui elementi sono stringhe, oppure che siano vuoti. Se una città è presente in **a** ma non in **b** allora la persona nata in quella città NON deve essere nell'insieme risultato. Ovviamente, se uno dei due insiemi in ingresso è vuoto, la funzione deve restituire un insieme vuoto.
- **A\_Ex6(l,c,n)** Scrivere una funzione che prende in ingresso una lista **l** di stringhe, un carattere **c** ed un numero intero **n** e restituisce una lista ottenuta da **l** eliminando tutte le stringhe che contengono almeno **n** volte il carattere **c**. Ad esempio, se **l** = ['palla','casse','palo'], **c** = 'a' ed **n** = 2 allora la funzione deve restituire la lista ['casse','palo']. Si noti che le stringhe non eliminate compaiono nel risultato nello stesso ordine in cui compaiono in **l**. Ovviamente se la lista **l** in ingresso è vuota, la funzione deve restituire una lista vuota.
- **A\_Ex7(l)** Scrivere una funzione che prende in ingresso una lista **l** di numeri interi positivi e restituisce una lista ottenuta modificando **l** nel seguente modo: ogni volta che un numero all'interno di **l** è più piccolo del successivo, alla lista viene aggiunta in fondo la differenza tra il secondo ed il primo dei due numeri. Ad esempio, se **l**=[10,1,11,31,251], allora la lista da restituire sarà [10,1,11,31,251,10,20,220,10,200,190]. Si noti che il confronto fra un numero ed il successivo deve essere fatto anche per i nuovi elementi inseriti, come mostrato dall'esempio. Ovviamente, se la lista in ingresso è vuota, la funzione deve restituire una lista vuota.
- **A\_Ex8(l)** Scrivere una funzione che prende in ingresso una lista **l** di insiemi di numeri interi e restituisce un altro insieme contenente tutti e soli gli elementi che appaiono in uno ed uno solo degli insiemi in **l**. Ad esempio, se **x**=[{3,2,90},{2,87,23},{2,23,3}], allora l'insieme da restituire sarà {90,87}. Se la lista in ingresso è vuota, la funzione deve restituire un insieme vuoto.