

Esercitazione Python n. 7 -- 14 Novembre 2023

Obiettivo dell'esercitazione è esercitarsi con liste, tuple e insiemi.

In laboratorio, per avviare la macchina virtuale, dovete selezionarla all'interno della schermata di VirtualBox (schermata visualizzata all'accensione del PC), cliccando sull'opzione **LXLE-BIAR-4.5**. Una volta avviata la macchina virtuale, svolgete gli esercizi così come indicato nel testo. Usate l'ambiente Spyder per svolgere gli esercizi. Ovviamente è possibile consultare il materiale didattico disponibile sulla pagina web del corso (<https://classroom.google.com/u/1/c/NjIwOTY0ODk3MDAx>). Si ricorda che le note relative alle lezioni Python possono essere lette con l'applicazione JupyterLab.

La consegna deve essere effettuata entro l'orario di fine dell'esercitazione.

LE ESERCITAZIONI SVOLTE CONSEGNATE OLTRE QUESTO TERMINE, O CHE NON RISPETTANO IL FORMATO INDICATO PER LA CONSEGNA, NON VERRANNO CONSIDERATE. In particolare, vi chiediamo di NON caricare un esercizio svolto per volta, di NON usare formati di compressione diversi da .zip, di NON rinominare i file o metterli in sottocartelle.

Fate attenzione che gli input siano richiesti all'utente UNO PER VOLTA e NELL'ORDINE RIPORTATO nell'esercizio e che le vostre stampe riportino a video i messaggi ESATTAMENTE nel formato atteso.

Ogni esercizio richiede che sia completata una funzione all'interno del file predisposto con lo stesso nome e, come visto a lezione, può essere testata eseguendo il file stesso, con i casi di test forniti. Notate infatti 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, come in sede di esame, per la correzione verranno usati insiemi di dati di test diversi.**

Esercizi

- **A_Ex1(I)** Scrivere una funzione che prende in ingresso una lista **I** 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 **I**. 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.
- **A_Ex2(I)** Scrivere una funzione che prende in ingresso una lista **I** di stringhe e restituisce un insieme di tutte e sole le coppie (x,n) (cioè tuple di dimensione due) tali che x è una stringa di **I** ed n è il numero di volte che la stringa x appare in **I**. Ad esempio, se $I = ['jkl', 'h', 'plqa', 'jkl', 'h', 'xkj']$, allora la funzione deve restituire l'insieme di coppie $\{('jkl', 2), ('h', 2), ('plqa', 1), ('xkj', 1)\}$. Se la lista **I** in ingresso è vuota, la funzione deve restituire l'insieme vuoto.
- **A_Ex3(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, citta_nascita) (ogni coppia indica che quella persona è nata in quella città) e l'insieme **b** contiene le coppie (citta, 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 $\{('Giovanni', 'Campania'), ('Marco', 'Lazio'), ('Giuseppe', 'Lazio'), ('Aldo', 'Liguria')\}$.

'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. Se uno dei due insiemi in ingresso è vuoto, la funzione deve restituire un insieme vuoto.

- **A_Ex4(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**. Se la lista **l** in ingresso è vuota, la funzione deve restituire una lista vuota.
- **A_Ex5(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. Se la lista in ingresso è vuota, la funzione deve restituire una lista vuota.
- **A_Ex6(v1,v2)** Scrivere una funzione che prende in ingresso due insiemi **v1** e **v2** di interi, e restituisce l'insieme di tutte e sole le coppie (**n1,n2**) del prodotto cartesiano di **v1** e **v2** tali per cui **n1** è un divisore di **n2** oppure **n2** è un divisore di **n1**. Ad esempio, se **v1**= {3,10,28} e **v2**= {2,20}, allora la funzione deve restituire l'insieme {(10,2),(28,2),(10,20)}. Se almeno uno tra i due insiemi **v1** e **v2** è vuoto, la funzione restituisce l'insieme vuoto.
- **A_Ex7(l)** Scrivere una funzione che prende in ingresso una lista non vuota **l** di coppie (**s1,n**) dove **s1** è una stringa e **n** un intero positivo, e restituisce la tupla (**s2,m**), tale che **s2** è la stringa che compare più volte come prima componente di **l** e **m** è il valore "massimo associato a **s2**", cioè il massimo tra tutti gli interi **n** tali che (**s2,n**) appartiene a **l**. Se ci sono più stringhe che compaiono lo stesso numero di volte come prima componente in **l**, la funzione restituisce (**s2,m**) tali che **m** è maggiore. Se ci sono più stringhe che compaiono lo stesso numero di volte come prima componente in **l** e il cui massimo associato è lo stesso, restituisce (**s2,m**) tali che **s2** è alfabeticamente più piccola. Ad esempio, se **l**=[['italiano',6], ['geografia', 10], ['matematica',6],['storia',8], ['matematica', 10], ['storia', 8], ['geografia', 7]], allora la funzione deve restituire ('geografia',10).
- **A_Ex8(l)** Scrivere una funzione che prende in ingresso una lista **l** di coppie (**s,l2**) dove **s** è una stringa e **l2** una lista di stringhe, e restituisce una lista **l3** di coppie (**s,v**), ordinata rispetto ad **s**, tale che **s** è una stringa che compare come prima componente in una coppia di **l** e che compare al più una volta come prima componente di **l3**, e **v** è l'insieme che contiene tutti e soli gli elementi che compaiono in almeno una lista **l2**, per ogni **l2** tale che (**s,l2**) appartiene a **l**. Ad esempio, se **l**=[('pippo',['blu','giallo']), ('minnie', ['rosa']), ('pippo', ['rosso', 'blu'])], allora la funzione deve restituire la lista [(('minnie',{'rosa'}), ('pippo', {'blu', 'rosso', 'giallo'}))]. Se la lista **l** in ingresso è vuota, la funzione deve restituire la lista vuota.