

# Fondamenti di Informatica per Ing. Gestionale

---

## Esercitazione del 26 Aprile 2018

---

### Istruzioni

Collegarsi al sito del corso <https://piazza.com/uniroma1.it/spring2018/1017401/home> e scaricare i file necessari all'esercitazione dalla sezione risorse. I file sono compressi in un file `.zip`, contenente una cartella con i file dell'esercitazione.

### Svolgimento degli esercizi - Leggere attentamente

Leggere attentamente il testo e risolvere gli esercizi proposti.

Per ogni esercizio avete unacartella EsercN che contiene un file dal nome A\_ExN.py (dove N è il numero dell'esercizio) con lo scheletro della soluzione. Il file A\_S\_ExN.py contiene la soluzione dell'esercizio proposta dal docente. *E' nel vostro diretto interesse non aprire quest'ultimo file prima di aver dedicato tutto il tempo disponibile a risolvere autonomamente l'esercizio.* Non create nuovi file.

Per verificare la correttezza di un esercizio **DOVETE** usare il programma TestEx.py (che non va assolutamente modificato), che proverà la vostra soluzione su un certo numero di casi di test, mostrandovi il confronto tra i risultati ottenuti con la vostra soluzione e usando la soluzione proposta dal docente. Ciò ha lo scopo di facilitarvi nella ricerca e correzione di eventuale errori.

### Esercizi

---

1. `A_Ex1(s)` Completare la funzione Python `A_Ex1(s)` che prende in ingresso una stringa `s`. La funzione deve restituire una stringa ottenuta da `s` scambiando ogni carattere in posizione pari con quello che lo segue. Qualora la stringa abbia lunghezza dispari l'ultimo carattere non va spostato (in particolare, una stringa di un solo carattere non va modificata). Inoltre, se `s` è la stringa vuota la funzione deve restituire la stessa stringa. Ad esempio, se la stringa fosse `"vicendevo1e"` la funzione dovrebbe restituire la stringa `"ivecdnve1oe"`.
2. `A_Ex2(l)` Scrivere una funzione che prende in ingresso una lista formata da stringhe rappresentanti nomi di esami e voti e restituisce una lista contenente i valori numerici nello stesso ordine in cui appaiono nella lista originaria, seguiti dalla media dei valori numerici medesimi. Ad esempio, se la lista fosse `["Fisica", "30", "Chimica", "30", "Logica", "30"]` la funzione dovrebbe restituire come risultato la lista `[30, 30, 30, 30.0]`.
3. `A_Ex3(file, c)` Completare la funzione Python `A_Ex3(file, c)` che prende in ingresso il nome di un file testo e un carattere. Il file contiene parole separate da spazi ed è privo di punteggiatura. La funzione deve restituire una lista contenente tutte le parole che iniziano con il carattere ricevuto in input. Ogni parola deve comparire una sola volta e la lista deve essere ordinata alfabeticamente. Ad esempio, se il testo fosse:

*this book is designed*

*to be a text book*

e il carattere fosse `'b'`, la funzione dovrebbe restituire la lista `["be", "book"]`. Nel caso non vi siano parole che inizino con il carattere considerato, la funzione deve restituire la lista vuota.

4. `A_Ex1(s)` Scrivere una funzione che prende in ingresso una stringa e restituisca una lista di liste ove ciascuna sottolista è composta da due elementi, il primo elemento è il carattere della stringa mentre il secondo elemento è il numero di volte, in formato stringa, che il carattere compare nella stringa. Ad esempio se la stringa data è `"abaco"` la funzione deve restituire `[['a', '2'], ['b', '1'], ['c', '1'], ['o', '1']]`. Se la stringa in ingresso è vuota allora la funzione deve restituire la lista vuota. Si noti, come mostrato nell'esempio, che il carattere `'a'` compare due volte ma nella lista di uscita vi è una sola sottolista associata al carattere il cui numero di occorrenze però è `'2'`.