

Compito d'esame -- 8 febbraio 2023 -- Compito C

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 il programma `REGISTRAs studente.py` 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 `REGISTRAs studente.py`.

Tempo a disposizione: 1 ora e 45 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 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(l)** Scrivere una funzione che riceve in ingresso una lista di interi **l** e calcola la lunghezza della più lunga sequenza di **l** la cui somma è strettamente minore di 100. Per esempio, se `l=[335,88,21,2,3,9,150]` la funzione deve restituire 4.
- **Ex2(file)** Scrivere una funzione che riceve in ingresso il nome di un file di testo **file** che contiene solo parole (solo caratteri alfabetici, spazi e `\n`) e restituisce il numero di parole che cominciano con una consonante e contengono la stessa consonante in due locazioni consecutive nel resto della parola, e.g., `lallismo`. La funzione non deve distinguere tra lettere maiuscole e minuscole, quindi `Lallismo` viene contata. N.B. La soluzione **deve** usare una espressione regolare (sono vietati `for` e `while`). Per esempio, se il file contiene:

Ci sono molti **pappici** in questa assurda ma **sussurrata** minestra

La funzione deve restituire 2.

- **Ex3(file,n)** Scrivere una funzione che riceve in ingresso il nome di un **file** che contiene solo parole (solo caratteri alfabetici, spazi e \n) e che:
 1. calcola un dizionario che ha per chiave le lettere minuscole dell'alfabeto e come valore una lista delle parole del file che cominciano per la chiave e che soddisfano la seguente proprietà: il primo elemento nella lista è la prima parola del file che comincia con la chiave e ogni successiva parola che comincia con la chiave viene aggiunta **solo** se ha esattamente **n** lettere distinte (quindi senza contare i duplicati) in comune con l'ultima parola inserita. N.B., la funzione non distingue tra lettere maiuscole e minuscole e
 2. **restituisce** la lunghezza massima delle liste nel dizionario.

Per esempio, chiamando la funzione con `Ex3(file,3)`, se il file contiene:

Nel catino esiste un unico attrezzo più utile di quello nel campo

La funzione:

- 1) calcola il dizionario

```
{ 'n': ['nel', 'nel'], 'c': ['catino', 'campo'], 'e': ['esiste'], 'u': ['un'],  
  'a': ['attrezzo'], 'p': ['più'], 'd': ['di'], 'q': ['quello']}
```

- 2) **restituisce 2.**

N.B.: le parole `unico` e `utile` **non** vengono inserite nella lista della chiave **u** perchè, quando vengono lette, l'ultima parola della lista `un` non ha esattamente 3 lettere distinte in comune con loro.

Domande veroFalso

Ex4 Il file `Ex4.py` contiene la funzione `veroFalso()` che stampa 8 domande sullo schermo. La funzione deve essere modificata cambiando il valore del `return`, elencando le lettere delle domande che ritenete essere vere. Ad esempio, se ritenete che le domande B e C sono vere il `return` deve essere modificato in

```
return 'BC'
```