

Compito d'esame -- 24 gennaio 2024 -- Compito D
Nota: gli studenti DSA devono sostenere gli esercizi 1, 2 e 4

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`.

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**.

È 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.

Per risolvere l'esercizio si possono importare solo le seguenti librerie: numpy, math, pandas e re.

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(s,l)** Scrivere una funzione che riceve in ingresso una stringa `s` ed una lista `l` di stringhe e calcola quante stringhe di `l` contengono tutti e soli i caratteri di `s` (anche in numero diverso dalle occorrenze in `s`). Ad esempio, se la stringa `s` vale `'canto'` e la lista `l` vale `['canto', 'conto', 'cotanto', 'taccone']` la funzione deve restituire 2 poiché `'canto'` e `'cotanto'` contengono tutti e soli i caratteri di `'canto'`, mentre `'conto'` non contiene la `'a'` e `'taccone'` contiene anche la `'e'`.
- **Ex2(file)** Scrivere una funzione che riceve in ingresso il nome di un file di testo `file` e calcola quante ore (corrette e scorrette) sono presenti nel file. Le ore sono scritte in questo formato:
 - 1 o 2 cifre
 - Il carattere separatore `':'` oppure `'.'`
 - 1 o 2 cifre

- **Lo stesso carattere separatore di cui sopra**
- **1 o 2 cifre**

Un'ora è corretta se, oltre a rispettare il formato di sopra corrisponde anche ad un'ora corretta. Ad esempio, 1:1:01 è corretta, mentre 25.11.19 non è corretta (assumete che le ore possano andare da 0 a 24 inclusi, mentre minuti e secondi da 0 a 59). La funzione deve restituire due numeri interi, cioè il numero di ore che rispettano il formato di sopra sono corrette ed il numero di ore che rispettano il formato di sopra ma NON sono corrette. Ad esempio, se il file **file** contiene:

Fin dalle 12:00:00 mi sono attivato, poi alle 25.3.5 ho lavorato, fino alle 11:13:19.
Alle 12:12.01 ho deciso che dalle 1:1:11 alle 31:9:13 mi riposerò

Allora la funzione deve restituire (3,2) poiché 12:00:00, 11:13:19 e 1:1:11 sono ore corrette, 25.3.5 e 31:9:13 sono ore scorrette, mentre 12:12.01 non è un'ora (usa 2 separatori diversi, non ammesso). NB: per la soluzione si consiglia (ma non è obbligatorio) l'uso delle espressioni regolari.

- **Ex3(file,n)** Un gruppo di amici sta giocando a un gioco tipo gioco dell'oca, dove i giocatori si spostano a turno tirando un dado su un tabellone fatto di **n** caselle (numerate da 1 ad **n**) con le seguenti regole:
 - *Tutti i giocatori partono dalla casella speciale 0 e muovono nell'ordine in cui sono nel file.*
 - *Se un giocatore, che si trova nella posizione i , tira il dado e fa il numero k (compreso tra 1 e 6) allora si sposta nella posizione $i+k$. Se in quella posizione c'è un altro giocatore allora il giocatore che ha mosso rimane fermo. Se $i+k$ è maggiore di **n** allora la mossa non si può effettuare ed il giocatore rimane fermo. La casella speciale 0 è la sola dove possono essere presenti contemporaneamente più giocatori.*
 - *Un giocatore vince se arriva per primo nella casella **n**. Se alla fine della partita nessuno arriva nella casella **n** allora non c'è alcun vincitore. Poiché i giocatori muovono in un ordine fissato, non ci può essere un pari merito.*

Le informazioni sulle mosse sono nel file di testo in formato csv, **file** che ha il seguente formato:

```
NomeGiocatore_1,...,NomeGiocatore_m  
Tiro_1,...,Tiro_m
```

dove NomeGiocatore_*i* identifica il giocatore *i*, mentre le righe successive contengono i tiri che hanno fatto i giocatori nei vari turni. Il numero dei giocatori non è fissato e nemmeno il numero di tiri, ma potete assumere che nel file tutti i giocatori fanno sempre lo stesso numero di tiri, cioè tutte le righe contengono lo stesso numero di valori, e che tutti i tiri siano corretti (cioè contengano valori tra 1 e 6).

La funzione deve restituire il nome del vincitore (se ce n'è uno) altrimenti deve restituire "Nessuno". Ad esempio, se **n** vale 10 ed il file **file** contiene:

```
Paolo,Miriam,Carla,Giulia,Flavio  
2,1,2,3,5  
5,6,3,4,4  
3,2,2,4,5  
5,6,3,1,2
```

allora la funzione deve restituire "Paolo", poiché "Carla" al primo tiro trova "Paolo" nella posizione 2 e quindi rimane ferma nella casella 0.

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'
```