

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

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(s,c1,c2)** Scrivere una funzione che riceve in ingresso una stringa `s` e due caratteri `c1` e `c2` e calcola la lunghezza della più lunga sottostringa (caratteri consecutivi) di `s` che contiene lo stesso numero (maggiore di zero) di `c1` e `c2`. Ad esempio, se `s` vale 'matanto va la gatta al lardo', `c1` = 't' e `c2` = 'a', la funzione deve restituire 16 perché esiste una sottostringa di lunghezza 16 avente la caratteristica richiesta (quella in grassetto) e non c'è in `s` una sequenza di questo tipo che abbia lunghezza superiore.
- **Ex2(file)** Scrivere una funzione che riceve in ingresso il nome di un file di testo `file`, e conta in quante righe è presente una sequenza di 3 parole consecutive che hanno la proprietà di avere tutte e 3 una doppia. Se ad esempio il file è il seguente:

Sono andato a casa sotto ferro butta
Metto leso
Pelle rotta
Cielo rosso netto panna verde

Allora la funzione deve restituire 2 poiché nella prima e nell'ultima riga ci sono 3 parole consecutive con le doppie. N.B.: l'esercizio va risolto usando le espressioni regolari.

- **Ex3(file)** Scrivere una funzione che prende in ingresso un file di testo in formato csv **file** che contiene le informazioni sul prezzo (in un dato mese) di alcuni oggetti in vendita e calcola un dizionario con chiave il nome dell'oggetto e valore il mese in cui l'oggetto ha avuto l'aumento maggiore. Più precisamente, il file ha il seguente formato:

```
Mese,oggetto1,prezzo1,...,oggetton,prezzon
```

dove *Mese* è il mese (potete assumere che i mesi siano presenti nel file in ordine), *oggetto1* è il nome di un oggetto e *prezzo1* è il suo prezzo per quel mese, il numero *n* di oggetti non è fissato e può cambiare di mese in mese. Se un oggetto non è ancora mai apparso nel file allora non è ancora in vendita, se era già apparso in precedenza e per un mese non compare vuol dire che il suo prezzo è rimasto invariato per quel mese. Se ci sono più mesi in cui si verifica l'aumento massimo, selezionate il primo presente nel file. Se un oggetto non subisce mai un rincaro allora dovete restituire 'nessun rincaro'. Ad esempio, se il file **file** contiene:

```
gennaio,giacca,200,pantalone,100  
febbraio,camicia,50,pantalone,110  
marzo,pantalone,120,giacca,230
```

allora la funzione deve restituire {'giacca': 'marzo', 'pantalone': 'febbraio', 'camicia': 'nessun rincaro'}, poiché la giacca ha il prezzo di 200 a gennaio e febbraio (prezzo stabile perché non compare) e poi aumenta di 30, il pantalone ha il prezzo massimo a marzo ma l'aumento è lo stesso a febbraio e marzo e quindi il valore corretto è febbraio, mentre la camicia viene messa in vendita a febbraio e non subisce mai un aumento..

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'
```