

Compito d'esame -- 14 febbraio 2022 -- Compito D

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 (con un doppio click sull'icona) il programma `RegistraStudente` 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 `RegistraStudente`.

Tempo a disposizione: 1 ora e 30 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 di dati di test diversi**.

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(I)** Scrivere una funzione che riceve in ingresso una lista `I` di numeri interi e calcola la lunghezza della più lunga sequenza di numeri tale che, a partire dal terzo numero della sequenza, il numero in posizione `i` è uguale alla somma dei numeri in posizione `i-1` e `i-2`. Si assuma che la lista `I` in input abbia sempre lunghezza maggiore o uguale a 3. Ad esempio, se la lista `I` è `[7, 3, 10, 2, 5, 7, 12, 3]` allora la funzione deve restituire 4 poiché la sequenza `2, 5, 7, 12` rispetta la proprietà ed è quella di lunghezza massima. Se non esiste alcuna sequenza di questo tipo, la funzione restituire 0.
- **Ex2 (I,s,t)** Scrivere una funzione che seleziona delle parole per il gioco wordle (simile al mastermind) in cui bisogna indovinare una parola di 5 lettere minuscole (che per semplicità assumiamo siano tutte diverse tra di loro) in base agli indizi che ci vengono forniti da un precedente tentativo. La funzione riceve in ingresso una lista di stringhe `I`, che sono le possibili parole da provare, la stringa `s` provata in precedenza ed una tupla `t` di 5 numeri interi, tale che ciascun numero in `t` rappresenta quanto la lettera di `s` nella stessa posizione sia corretta. Se il valore è 0 allora la lettera del tentativo NON compare mai nella soluzione, se è 1, la lettera compare ma non in quella posizione, se è 2 allora compare proprio in quella posizione. Ad

esempio, se la stringa **s** è `pinza` e la tupla **t** è `(2, 0, 0, 0, 1)`, allora vuol dire che la soluzione NON contiene le lettere `'i'`, `'n'` e `'z'`, contiene la `'p'` in posizione 1 e contiene la `'a'` ma NON in posizione 4. La funzione deve restituire la lista delle parole di **l** (nell'ordine in cui compaiono in **l**) che sono tentativi corretti, cioè che non violano gli indizi che abbiamo già ottenuto dal tentativo precedente. Ad esempio, se **l** vale `['caste', 'pesta', 'paste', 'pance']`, **s** vale `'pinza'` e **t** vale `(2, 0, 0, 0, 1)`, allora la funzione deve restituire `['paste']`, poiché `'caste'` non ha la `'p'` in prima posizione, `'pesta'` ha la `'a'` in posizione 4 e `'pance'` contiene la `'n'`.

- **Ex3(maxSpesa,file)** Scrivere una funzione che decide gli acquisti a rate di un individuo in un periodo di 5 anni. Questa funzione prende in ingresso un numero intero **maxSpesa** che rappresenta la massima spesa mensile che la persona può permettersi ed un file csv **file** che contiene (in ordine crescente di mese) le informazioni sulle offerte di oggetti da comprare a rate che la persona riceve (e che deve decidere se acquistare) nel seguente formato:

`MeseOfferta, NomeOggetto, NumRate, ImportoRata`

dove **MeseOfferta** è un numero intero > 0 e ≤ 60 (5 anni) che denota il mese in cui è stata fatta l'offerta, **NomeOggetto** identifica l'oggetto in offerta (potete assumere che nessun oggetto venga offerto più di una volta), **NumRate** è il numero di rate mensili che il cliente deve pagare (ogni mese a partire da quello successivo a quello dell'offerta ed in numero tale che termineranno entro il mese 60) e **ImportoRata** è l'ammontare (in euro) che il cliente deve pagare ogni mese (fino a completamento del numero di rate) per comprare quell'oggetto.

L'obiettivo è calcolare, per ogni oggetto, se verrà comprato o meno. Le offerte vengono accettate se in nessun mese la persona si troverà a pagare più del suo **maxSpesa**, cioè il totale delle rate che deve pagare ogni mese non deve superare mai questo valore. Ad esempio, se il limite di spesa mensile **maxSpesa** è 200 ed il file **file** contiene:

```
1,Moto,20,105
1,Casco,20,25
2,Cappotto,30,100
10,Guanti,5,50
30,Sci,10,200
```

allora la funzione deve restituire `{'Moto': 'SI', 'Casco': 'SI', 'Cappotto': 'NO', 'Guanti': 'SI', 'Sci': 'SI'}`

Nota: Potete assumere che il numero di rate e l'importo di una rata nel file **file** siano sempre convertibili in numeri interi.