

Esercitazione di Fondamenti Informatica del 15.04.2016

Istruzioni (leggere attentamente)

Nota importante: la mancata osservanza delle seguenti regole può comportare la perdita di informazioni necessarie alla valutazione dell'esame.

Registrazione dei dati dello studente (SOLO PER ESAME, NON FARE OGGI)

Prima di iniziare il compito, cliccare su `Risorse del Computer`, aprire prima la cartella (di rete) che trovate nella finestra e poi la cartella `Esame`. Eseguire il programma `registrazione.pyc`, inserire i dati personali fornendo (separatamente) Numero di Matricola, Cognome e Nome. Il programma genera un file `studente.txt` che non deve essere modificato manualmente. Verificare che i dati nel file `studente.txt` siano corretti, in caso di errore potete rieseguire il programma `registrazione.pyc`.

Svolgimento degli esercizi

Leggere attentamente il testo e risolvere gli esercizi proposti. Per ogni esercizio avete una cartella `EsercN` che contiene un file dal nome `A_ExN.py` (dove `N` è il numero dell'esercizio) con lo scheletro della soluzione.

Per testare un esercizio **DOVETE** usare il programma:

- `TestEx.pyc` (Se non funziona, **in aula 17 usate `TestEx.bat`**)

Non create nuovi file. Il programma `TestEx.pyc` (oppure se non funziona, usate `TestEx.bat`) proverà la vostra soluzione con un insieme di casi di test.

E' possibile consultare la documentazione ufficiale del linguaggio Python (disponibile sul desktop cliccando su `Doc_Python3`), ma **non è possibile usare libri o appunti**. In caso di problemi tecnici chiedere ai docenti o ai tecnici del laboratorio.

Esercizi

- **A_Ex1(file, c)** scrivere una funzione che riceve in ingresso il nome di un file **file** e un carattere **c** restituisce una nuova lista che contiene tutte e solo le parole del file che contengono il carattere **c** (sia minuscolo che maiuscolo). Ad esempio se il **file** vale `"NEL mezzo del cammin\n"` e **c** vale `'e'` la funzione deve restituire `['NEL', 'mezzo', 'del']`;
- **A_Ex2(l, file)** scrivere una funzione che riceve in ingresso una lista **l** di almeno 2 stringhe tutte di lunghezza differente e il nome di un file **file** e crea un nuovo **file** che contiene, su due righe, la stringa più corta e la più lunga della lista. Ad esempio, se **l** vale `['sellino', 'trota', 'il', 'stringa molto lunga']`, il file **f** deve contenere `"il\nstringa molto lunga\n"`.
- **A_Ex3(s)** scrivere una funzione che prende in ingresso una stringa **s** composta di caratteri alfabetici e spazi bianchi e restituisce una lista con tutte le parole diverse della stringa. Ad esempio, se **s** vale `"tanto va la gatta al lardo che ci lascia la zampina"` la funzione deve restituire `["tanto", "va", "la", "gatta", "al", "lardo", "che", "ci", "lascia", "zampina"]`.
- **A_Ex4(file)** Scrivere una funzione che prende in ingresso un **file** che contiene **n** interi per ogni riga separati da spazio e restituisce l'intero più piccolo e l'intero più grande. Ad esempio, se il file vale `"3 7\n55 17\n8 23\n99 12\n6 21\n"` la funzione deve restituire `(3,99)`.
- **A_Ex5(file_in, file_out)** Scrivere una funzione che prende in ingresso un **file_in** che contiene **n** interi per ogni riga separati da `','` e crea un nuovo file che contiene la somma delle righe del file di ingresso. Ad esempio, se le righe del file sono `"3;4;5"` e `"9;2;6"`, la funzione deve creare un file con due righe: `12` e `17`.
- **A_Ex6(file)** Scrivere una funzione che prende in ingresso un **file** csv che contiene l'altezza di **n** persone misurata ogni anno e restituisce il nome della persona che è la più alta di tutti in tutte le misurazioni. Se tale persona non esiste la funzione deve restituire `None`. Ad esempio, se le righe del file sono `"Marco, Lucio, Giovanni", "1.50, 1.30, 1.45", "1.55, 1.50, 1.49"` la funzione deve restituire `"Marco"`.