

16.02.2015

## Fondamenti Informatica I (modulo Python) Compito D

### 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

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`

Si consiglia, in ogni caso, prima di usare il programma `TestEx.pyc` di verificare la correttezza sintattica della vostra soluzione premendo `F5`.

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

- **D\_Ex1(l)** (6 punti) scrivere una funzione che riceve in ingresso una lista `l` di interi e calcoli la lunghezza della più lunga sequenza di interi diversi (ovvero in cui ciascun intero compare solo una volta) **a partire dall'intero in posizione 0**. Ad esempio se la lista `l` vale `[3,7,9,5,3,2]` allora la funzione deve restituire 4 poiché la sequenza `3,7,9,5` è lunga 4 e non contiene nessun intero ripetuto.
- **D\_Ex2(s1,s2)** (6 punti) scrivere una funzione che riceve in ingresso due stringhe `s1` e `s2` contenenti esclusivamente caratteri alfabetici senza accenti e verifichi se alle vocali della stringa `s1` siano associate vocali della stringa `s2` nelle stesse posizioni. Ad esempio, se `s1` vale `'Lucca'` e `s2` vale `'cassa'` la funzione deve restituire `True` in quanto sia nell'indice 1 ('u' e 'a') che nell'indice 4 ('a' e 'a') l'elemento di `s2` è una vocale come in `s1`. Si faccia attenzione che le 2 stringhe possono essere di lunghezza diversa e che non si possono confrontare elementi che siano fuori dal range di una delle stringhe.
- **D\_Ex3(f)** (9 punti) scrivere una funzione che prende in ingresso un file contenente il resoconto dell'occupazione di disco di vari utenti. La prima riga del file ha il formato:

`assegnazioneIniziale1;nome1;assegnazioneIniziale2;nome2;.....assegnazioneInizialeN;nomen`

in cui si specifica lo spazio in `MegaByte` assegnato inizialmente a ogni utente. Le righe successive contengono il resoconto delle varie operazioni di lettura e scrittura, specificando in ognuna quanto spazio su disco ogni utente ha guadagnato o perso nel seguente formato:

`variazione1;variazione2;.....;variazionen`

dove la variazione può essere positiva o negativa. Leggere il file e costruire la lista ordinata (in ordine temporale) degli utenti che a ogni ciclo hanno meno spazio su disco (assumere che ne esista sempre solo uno). Ad esempio se il **file** contiene:

```
1000;Paolo;500;Marco;1000;Anna;300;Giorgio
-100;+-400;-200;-100
+50;+-50;0;+200
```

la funzione deve restituire: [ 'Marco', 'Marco'] poiché dopo il primo ciclo Marco ha solo 100 MB e al secondo ciclo solo 50MB.

- **D\_Ex4(file1, file2) (9 punti)** scrivere una funzione che prende in ingresso due file **file1** e **file2**, dove **file1** contiene un insieme di dati su componenti di un medicinale nel formato:

- 

Componente;PesoInMilligrammiPerDose

- Il file **file2** contiene un insieme di allergie a componenti nel seguente formato:

- 

NomePaziente;Componente;PesoMassimoAmmissibile

Il programma deve restituire una lista, **ordinata alfabeticamente**, di tutti i pazienti per cui quel medicinale è potenzialmente pericoloso per lui (ovvero in cui il peso per singola dose è superiore a quella massima ammissibile). Ad esempio, se file1 è il seguente:

```
diosodio;30
tiamfenicolo;810
acetilsalicinico;100
```

E **file2** contiene:

```
Luca;E112;40
Mario;diosodio;15
Mario;E112;20
Sergio;acetilsalicinico;50
```

la funzione deve restituire: ['Mario', 'Sergio'], in quanto la dose massima di diosodio per Mario è 15 e la dose massima di acetilsalicinico per Sergio è 50 e il medicinale in questione le supera entrambe.