

Fondamenti di Informatica I (12 cfu) - A.A. 2013-2014

Corsi di Laurea in Ingegneria Gestionale

Sapienza Università di Roma

Prova al calcolatore Esercitazione 30 Maggio 2014

Esercizio 1

Si assuma una rappresentazione Python di città mediante dizionario: ogni città è rappresentata mediante un dizionario contenente le seguenti chiavi:

- **nome** il nome della città (una stringa)
- **regione** la regione in cui si trova la città (una stringa)
- **capoluogo** un valore booleano che se posto uguale a **True** indica che la città è capoluogo di regione
- **abitanti** il numero di abitanti della città (un intero)

Completare il file `Esercit12Prog1.py` con la definizione delle seguenti funzioni Python:

- `capoluogo(lista_citta, nome_regione)` che presi in ingresso una lista `lista_citta` contenente dizionari rappresentanti città e una stringa `nome_regione` contenente il nome di una regione restituisce il nome della città capoluogo della regione; se la regione non ha capoluogo, allora il metodo restituisce la stringa vuota;
- `altre_citta(lista_citta, nome_regione)` che presi in ingresso una lista `lista_citta` contenente dizionari rappresentanti città e una stringa `nome_regione` contenente il nome di una regione restituisce una lista contenente i nomi delle città della regione `lista_regione` che non sono capoluogo di provincia;
- `citta_popolose(lista_citta, limite)` che presi in ingresso una lista `lista_citta` contenente dizionari rappresentanti città e un intero `limite` contenente un numero di abitanti restituisce una lista contenente i nomi delle città che hanno un numero di abitanti maggiore di `limite`;

Scrivere la soluzione in modo da poter eseguire il programma di prova `ProvaEx1.py`, senza doverlo modificare.

Esercizio 2

Scrivere una funzione Python `somma_righe(matrice, estremi)` (contenuta nel file `Esercit12Prog2.py`) che prese in ingresso una matrice $n \times m$ di interi `matrice` e una matrice $n \times 2$ di interi `estremi`,

restituisca una lista di interi in cui l'elemento i -esimo contiene la somma degli elementi della riga i -esima di `matrice` compresi fra gli elementi contenuti rispettivamente nella posizione 0 e 1 della riga i -esima della matrice `estremi`. Si considerino a titolo di esempio le matrici che seguono:

$$m = \begin{pmatrix} 1 & 2 & 0 & 3 \\ 1 & -1 & 2 & 1 \\ 0 & 2 & 4 & 0 \end{pmatrix} e = \begin{pmatrix} 0 & 3 \\ 1 & 2 \\ 1 & 1 \end{pmatrix} somma_righe = \begin{pmatrix} 6 \\ 1 \\ 2 \end{pmatrix}$$

Nell'esempio riportato sopra, il valore 6 è ottenuto sommando gli elementi della prima riga di m a partire dall'indice di colonna 0 fino all'indice di colonna 3. Analogamente il valore 1 (seconda riga) è ottenuto sommando gli elementi di m a partire dall'indice di colonna 1 fino all'indice di colonna 2 incluso. Infine il valore 2 (terza riga) è ottenuto sommando gli elementi di m a partire dall'indice 1 fino all'indice 1. Scrivere la soluzione in modo da poter eseguire il programma di prova `ProvaEx2.py`, senza doverlo modificare.

Esercizio 3

In un grafo, un cappio è un arco che ha nodo uscente ed entrante coincidenti. Scrivere una funzione Python `cappi(g)` (contenuta nel file `Esercit21Prog3.py`) che presi in ingresso un grafo g rappresentato come un dizionario (lista delle adiacenze) restituisca una lista di nodi che presentano cappi.

Scrivere la soluzione in modo da poter eseguire il programma di prova `ProvaEx3.py`, senza doverlo modificare.