

Esercizio 1

Si consideri il seguente programma Python:

```
def f2(i,j,a):
    u = a - 1
    return i == 0 or i == u or j == 0 or j == u

def f1(m):
    a = len(m)
    x = 0
    for i in range(a):
        for j in range(a):
            if f2(i,j,a):
                x += m[i][j]
            else:
                x -= m[i][j]
    return x == 0

m = [ [3, 2, 1, 3],
       [2, 9, 8, 0],
       [3, 7, 0, 4],
       [1, 2, 1, 2]
     ]

print(f1(m))
```

Si descriva sinteticamente la funzione svolta dal programma e si mostri la traccia d'esecuzione, sapendo che l'output prodotto è *True*.

Esercizio 2

Si scriva una funzione *calcola_lista* che riceve una lista *L* di interi ed un intero *x* e restituisce una lista contenente tutti gli indici *i* che soddisfano le seguenti condizioni:

- l'elemento $L[i]$ ha almeno x elementi successivi;
- nessuno degli x elementi successivi ad $L[i]$ ha valore maggiore di $L[i]$.

Esempio: Se $L = [3, 1, 2, 1, 2, 2, -1, 4]$ e $x = 3$, allora la funzione restituisce la lista $[0, 2]$ perché:

- l'elemento $L[0] = 3$ ha 7 ($\geq x$) elementi successivi e i primi x di essi sono ≤ 3 (1, 2 ed 1);
- l'elemento $L[1] = 1$ ha 6 ($\geq x$) elementi successivi, ma almeno uno tra i primi x è > 1 ($L[2] = 2$);
- l'elemento $L[2] = 2$ ha 5 ($\geq x$) elementi successivi e i primi x di essi sono ≤ 2 (1, 2 e 2);
- l'elemento $L[3] = 1$ ha 4 ($\geq x$) elementi successivi, ma almeno uno tra i primi x è > 1 ($L[4] = 2$);
- l'elemento $L[4] = 2$ ha 3 ($\geq x$) elementi successivi, ma almeno uno tra i primi x è > 2 ($L[7] = 4$);
- gli elementi $L[5]$, $L[6]$ ed $L[7]$ non hanno 3 elementi successivi.

Esercizio 3

Si vuole realizzare un'applicazione per la gestione dei movimenti finanziari avvenuti lungo un anno solare tra n società commerciali, ognuna identificata mediante un codice intero compreso tra 0 e $n-1$. Le informazioni relative ai movimenti sono memorizzate in una matrice quadrata M di ordine n , in cui il generico elemento $M[i][j]$ rappresenta la cifra totale trasferita dalla società con codice i alla società con codice j . Gli elementi sulla diagonale principale di M contengono il valore 0. Le informazioni relative alle società sono memorizzate in un dizionario D che associa al codice di ogni società una lista $[prop, n_dip, anno]$ in cui $prop$ rappresenta il nome del proprietario, n_dip il numero di dipendenti e $anno$ l'anno in cui la società è stata creata. Il *flusso netto* di una società è dato dalla differenza tra la somma delle cifre ricevute dalla società e la somma delle cifre trasferite dalla società ad altre (quest'ultima somma è detta *spesa totale*).

Si scriva un modulo Python che metta a disposizione (almeno) le seguenti funzioni:

- $flussi_netti(M, D)$, che restituisce una lista contenente i flussi netti delle società.
- $societa_intermedia(M)$, che restituisce il codice x della società che soddisfa le seguenti condizioni:
 - esiste almeno un'altra società con una spesa totale strettamente maggiore della società x ;
 - esiste almeno un'altra società con una spesa totale strettamente minore della società x .Se più di una società soddisfa le condizioni, la funzione restituisce il codice di una qualsiasi di esse.
- $flussi_per_dipendente(M, D, p)$, che restituisce una lista contenente, per ognuna delle società di cui p è proprietario, il rapporto tra flusso netto e numero di dipendenti della società.
- $statistiche(M, D, a_min)$, che restituisce un dizionario che associa ad ogni proprietario di società create non prima dell'anno a_min la somma delle spese totali di tali società.

Esempio: Se $M =$

0	20	30	25
10	0	10	10
30	10	0	20
20	20	20	0

e $D =$

Chiave	Valore
0	['Bianchi', 5, 2013]
1	['Rossi', 10, 2012]
2	['Bianchi', 10, 2014]
3	['Neri', 5, 2010]

allora:

- $flussi_netti(M)$ restituisce la lista $[-15, 20, 0, -5]$.
- $societa_intermedia(M)$ restituisce 2, perché la società con codice 2 ha spesa totale pari a 60 e
 - la società con codice 0 ha spesa totale pari a 75;
 - la società con codice 1 ha spesa totale pari a 30.

Si noti che la funzione avrebbe potuto restituire in alternativa il valore 3, perché la società con codice 3 ha spesa totale pari a 60.

- $flussi_per_dipendente(M, D, 'Bianchi')$ restituisce la lista $[-3, 0]$ perché:
 - la società con codice 0 ha flusso netto pari a -15 e 5 dipendenti;
 - la società con codice 2 ha flusso netto pari a 0 e 10 dipendenti.
- $statistiche(M, D, 2011)$ restituisce il dizionario

Chiave	Valore
'Bianchi'	135
'Rossi'	30