

Cognome e nome:		Matricola:		Corso:	<input type="checkbox"/> A <input type="checkbox"/> B
------------------------	--	-------------------	--	---------------	--

Esercizio 1

Si consideri il seguente programma Python:

```
def f_2(m,t):
    if t:
        r = m[0][0]
    else:
        r = m[0][len(m)-1]
    for i in range(1,len(m)):
        if t:
            r = max(r, m[i][i])
        else:
            r = max(r, m[i][len(m)-1-i])
    return r

def f_1(m,x):
    a = f_2(m,True)
    b = f_2(m,False)
    return x == max(a,b)

m = [ [1,3,3,5],
       [0,6,7,8],
       [4,4,1,9],
       [6,8,0,7] ]

print(f_1(m,7))
```

Si descriva sinteticamente la funzione svolta dal programma e, in particolare, si mostrino la traccia d'esecuzione e l'output prodotto.

Esercizio 2

Si scriva una funzione *verifica_liste* che riceve una lista *L1* di interi e una lista *L2* di booleani. La funzione restituisce *True* se e solo se per tutti gli indici *i* tali che *L2[i]==True*, la lista *L1* non contiene elementi di valore uguale a *L1[i]* in posizione diversa da *i*.

Esempio: Se *L1* = [7, 4, 7, 3, 5, 6] ed *L2* = [False, True, False, True, True, False], allora la funzione restituisce *True* perché:

- gli indici tali che *L2[i]==True* sono 1, 3 e 4;
- la lista *L1* non contiene elementi di valore uguale a *L1[1]* (cioè pari a 4) in posizione diversa da 1;
- la lista *L1* non contiene elementi di valore uguale a *L1[3]* (cioè pari a 3) in posizione diversa da 3;
- la lista *L1* non contiene elementi di valore uguale a *L1[4]* (cioè pari a 5) in posizione diversa da 4.

Esercizio 3

Si vuole realizzare un'applicazione per la gestione di abbonamenti ad una rassegna di eventi, della durata di una settimana. I dati relativi agli abbonamenti ed al loro utilizzo sono memorizzati in una lista A di interi avente n elementi (uno per ognuno degli n clienti, identificati con codici compresi tra 0 e $n-1$) ed in una matrice S di interi di dimensioni $n \times 7$. Il generico elemento $A[i]$ rappresenta il numero di eventi compresi nell'abbonamento dell' i -esimo cliente, mentre il generico elemento $S[i][j]$ rappresenta il numero di eventi ai quali l' i -esimo cliente ha partecipato nel giorno j .

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

1. *statistiche*(S), che restituisce una lista L il cui i -esimo elemento rappresenta il fatto che il cliente con codice i ha partecipato ad $L[i]$ eventi nell'arco dell'intera settimana.
2. *controlla*(A, S), che restituisce *True* se e solo se ognuno dei clienti ha partecipato, nell'arco dell'intera settimana, ad un numero di eventi minore o uguale a quello previsto dal suo abbonamento.
3. *dominante*(S, c), che restituisce il codice di un cliente che in ognuno dei giorni ha partecipato ad un numero di eventi maggiore o uguale al cliente con codice c . Nel caso in cui più clienti soddisfino tale condizione, la funzione restituisce uno qualsiasi di essi.
4. *presenze_filtrate*(S, c), che restituisce il giorno in cui è stato registrato il maggior numero partecipazioni ad eventi, escludendo dal calcolo il cliente con codice c . Nel caso in cui più giorni soddisfino tale condizione, la funzione restituisce uno qualsiasi di essi.

Esempio: Se $A = \begin{bmatrix} 9 \\ 6 \\ 12 \\ 5 \\ 10 \end{bmatrix}$ ed $S = \begin{bmatrix} 2 & 2 & 0 & 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 3 \\ 3 & 3 & 1 & 1 & 0 & 2 & 2 \\ 0 & 0 & 1 & 2 & 1 & 0 & 1 \\ 0 & 2 & 1 & 2 & 2 & 0 & 2 \end{bmatrix}$ allora:

- *statistiche*(S) restituisce la lista $\begin{bmatrix} 8 \\ 6 \\ 12 \\ 5 \\ 9 \end{bmatrix}$.
- *controlla*(A, S) restituisce *True* perché il cliente con codice 0 ha partecipato ad 8 eventi (il suo abbonamento ne prevedeva 9), il cliente con codice 1 ha partecipato a 6 eventi (il suo abbonamento ne prevedeva 6) e così via.
- *dominante*($S, 3$) restituisce 4 perché il cliente con codice 4, in ognuno dei giorni, ha partecipato ad un numero di eventi maggiore o uguale al cliente con codice 3.
- *presenze_filtrate*($S, 4$) restituisce 6 perché escludendo il cliente con codice 4, sono state registrate 6 partecipazioni nel giorno 0, 5 partecipazioni nel giorno 1, 3 partecipazioni nel giorno 2, 4 partecipazioni nel giorno 3, 2 partecipazioni nel giorno 4, 4 partecipazioni nel giorno 5 e 7 partecipazioni nel giorno 6.