

### **Esercizio 1**

Si consideri il seguente programma Python:

```
def f2(x,c):  
    s = []  
    for i in range(len(x)):  
        for j in range(len(x[0])):  
            if c and i % 2 == 0 or not c and i % 2 != 0:  
                s.append(x[i][j])  
    return s  
  
def f1(x):  
    return sum(f2(x,True)) == sum(f2(x,False))  
  
m = [[1,2,4,0,3],  
      [2,0,3,5,4],  
      [4,1,0,3,6],  
      [3,4,2,1,0]]  
  
print(f1(m))
```

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\_coppie* che riceve in ingresso una lista  $L$  di interi positivi ed un intero  $k$ . La funzione verifica se esiste un valore  $n$  tale che  $L$  contiene almeno  $k$  coppie di elementi consecutivi uguali ad  $n$  (si noti che una sequenza di  $x$  valori uguali consecutivi equivale a  $x-1$  coppie). Se tale valore esiste, la funzione lo restituisce. Se esiste più di un valore che soddisfa la condizione, la funzione ne restituisce uno qualsiasi. Se nessun valore soddisfa la condizione, la funzione restituisce  $-1$ .

*Esempio:* Se  $L = [5, 5, 7, 1, 1, 7, 5, 7, 5, 5, 5, 1, 1, 3]$  e  $k = 3$ , allora la funzione restituisce 5.

### Esercizio 3

Si vuole realizzare un'applicazione per la gestione delle informazioni riguardanti il prestito e la restituzione di libri presso una biblioteca durante una giornata lavorativa.

L'applicazione memorizza le prenotazioni da evadere durante la giornata utilizzando una matrice  $R$  avente 3 colonne, in cui la generica riga  $[nome\_cl, oper, cod\_libro]$  rappresenta il fatto che il cliente con nome  $nome\_cl$  ha prenotato l'operazione  $oper$  ('prestito' oppure 'restituzione') sul libro con codice  $cod\_libro$ . I dati sui libri sono memorizzati utilizzando un dizionario  $L$ , in cui ad ogni chiave  $cod\_libro$  è associata una coppia ( $scaffale, n\_copie$ ) che indica il codice dello scaffale su cui sono conservate le  $n\_copie$  copie del libro presenti in biblioteca all'inizio della giornata.

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

1.  $clienti\_attivi(R)$ , che restituisce una lista contenente i nomi dei clienti che hanno prenotato il maggior numero di operazioni.
2.  $scaffale\_occupato(L)$ , che restituisce il codice dello scaffale su cui sono attualmente conservati il maggior numero di libri (considerando per ogni libro il numero di copie). Se più di uno scaffale soddisfa la condizione, la funzione ne restituisce uno qualsiasi.
3.  $gestibili(R, L)$ , che restituisce *True* se e solo se le prenotazioni presenti in  $R$  sono gestibili, cioè se e solo se, per ogni libro, il numero di copie disponibili (considerando sia quelle presenti in biblioteca all'inizio della giornata che quelle di cui è stata prenotata la restituzione) è maggiore o uguale al numero di copie di cui è stato prenotato il prestito.
4.  $clienti\_veloci(R, L)$ , che restituisce una lista contenente i nomi dei clienti "veloci"; un cliente è considerato "veloce" se soddisfa entrambe le seguenti condizioni:
  - ha effettuato prenotazioni per un'unica operazione (prestito o restituzione);
  - tutti i libri per i quali ha effettuato prenotazioni sono conservati sullo stesso scaffale.

Esempio: Se  $R = \begin{bmatrix} 'Verdi' & 'prestito' & 'L1' \\ 'Rossi' & 'restituzione' & 'L2' \\ 'Verdi' & 'prestito' & 'L3' \\ 'Bianchi' & 'prestito' & 'L2' \\ 'Rossi' & 'prestito' & 'L1' \end{bmatrix}$  e  $L =$

Chiave	Valore
'L1'	('S1', 2)
'L2'	('S2', 1)
'L3'	('S1', 2)

allora:

- La funzione  $clienti\_attivi(R)$  restituisce  $['Verdi', 'Rossi']$  perché i clienti Verdi e Rossi hanno prenotato il maggior numero di operazioni (2).
- La funzione  $scaffale\_occupato(L)$  restituisce 'S1' perché sullo scaffale S1 sono attualmente conservati il maggior numero di libri (4).
- La funzione  $gestibili(R, L)$  restituisce *True* perché
  - per il libro L1 sono stati prenotati prestiti per 2 copie e sono disponibili 2 copie;
  - per il libro L2 è stato prenotato il prestito di una copia e sono disponibili 2 copie (di cui una in restituzione);
  - per il libro L3 è stato prenotato un prestito e sono disponibili 2 copie.
- La funzione  $clienti\_veloci(R, L)$  restituisce  $['Verdi', 'Bianchi']$  perché il cliente Verdi ha prenotato solo prestiti di libri sullo scaffale S1 e il cliente Bianchi ha prenotato solo prestiti di libri sullo scaffale S2.