

Programmazione e Algoritmi 1

A.A. 2023/24 — Compito del 22 gennaio 2024

prof. Gianluca Amato

Gli esercizi di programmazione saranno valutati sulla base della correttezza, efficienza e comprensibilità della soluzione proposta. In generale, **non è possibile usare tipi che non fanno parte del programma del corso**.

Esercizio 1 (5 punti)

Scrivere una funzione `same_set` che prende in input due liste e determina se esse hanno gli stessi elementi, ignorando sia l'ordine che i duplicati. Per esempio, le liste `[1, 4, 9, 16, 9, 7, 4, 9, 11]` e `[11, 11, 7, 9, 16, 4, 1]` hanno gli stessi elementi, mentre `[1, 4, 9, 19]` e `[4, 4, 19, 1, 3]` no. La funzione deve restituire `True` se le due liste hanno gli stessi elementi, `False` altrimenti.

Esercizio 2 (5 punti)

Scrivere alcuni test nel framework `pytest` per verificare il corretto funzionamento di `same_set`. In particolare:

- controllare il corretto funzionamento della funzione per gli input di esempio dell'Esercizio 1;
- controllare il corretto funzionamento della funzione quando la prima lista è generata casualmente e la seconda è ottenuta dalla prima scambiando casualmente due elementi.

Esercizio 3 (5 punti)

Un quadrato magico è una matrice quadrata di numeri interi positivi tale che la somma dei numeri di ogni riga, di ogni colonna e delle due diagonali è uguale. Per esempio, la seguente matrice è un quadrato magico, in cui tutte le righe, le colonne e le diagonali hanno somma 15:

2	7	6
9	5	1
4	3	8

Scrivere una funzione `is_magic_square` che prende in input una matrice quadrata di numeri interi e restituisce `True` se la matrice è un quadrato magico, `False` altrimenti. Ad esempio `is_magic_square([[2, 7, 6], [9, 5, 1], [4, 3, 8]])` deve restituire `True`.

Esercizio 4 (5 punti)

Scrivere un programma che legge da un file una matrice quadrata di numeri interi e determina se si tratta o meno di un quadrato magico (è ovviamente possibile usare la funzione `is_magic_square`). Il file deve avere il seguente formato: ogni riga del file rappresenta una riga della matrice, mentre le colonne sono separate da spazi. Ad esempio, un file il cui contenuto è

```
1 2 3
4 5 6
7 8 9
```

rappresenta il quadrato (non magico)

1	2	3
4	5	6
7	8	9

Esercizio 5 (5 punti)

Illustrare il funzionamento dell'algoritmo di ordinamento per selezione (selection sort) prima in termini generali, e poi applicandolo alla lista `[3, 5, 2, 1, 4]`.

Esercizio 6 (8 punti)

Si consideri il seguente codice Python:

```
1 def f1(x, i, l):
2     for j in range(i, len(l)):
3         if l[j] == x:
4             return True
5     return False
6
7 def f2(l):
8     for i in range(len(l)):
9         if f1(l[i], i+1, l):
10            return True
11    return False
12
13 print(f2([1, 5, 3, 5, 2]))
```

Eseguire passo passo il programma (senza l'ausilio di un computer) e fornire la traccia di esecuzione. **Inoltre**, spiegare a parole cosa fanno le funzioni `f1` ed `f2`.