

Programmazione e Algoritmi 1

A.A. 2023/24 — Compito di prova

prof. Gianluca Amato

Gli esercizi di programmazione saranno valutati sulla base della correttezza, efficienza e comprensibilità della soluzione proposta.

Esercizio 1 (5 punti)

Scrivere una funzione `frequenza` in Python che prende come parametro una lista `dati` di numeri interi compresi tra 0 e 9 e restituisce una nuova lista che contiene, in posizione i , il numero di occorrenze del numero i in `dati`. Ad esempio, se la lista `dati` è `[1, 2, 3, 1, 9, 9, 7, 1]`, la funzione restituirà la lista `[0, 3, 1, 1, 0, 0, 0, 1, 0, 2]`.

Esercizio 2 (3 punti)

Scrivere una funzione `frequenza2` che estende la funzione `frequenza` in modo che possa funzionare anche con liste di numeri non negativi arbitrari (non necessariamente compresi tra 0 e 9). Ad esempio, se la lista passata come argomento è `[0, 0, 15, 15, 15, 2]`, la funzione modificata restituirà la lista `[2, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3]`.

Esercizio 3 (7 punti)

Scrivere un test nel framework `pytest` per verificare il corretto funzionamento della funzione `frequenza`. In particolare:

- controllare il corretto funzionamento della funzione per gli input `[]` e `[1, 2, 3, 1, 9, 9, 7, 1]` (l'esempio dell'Esercizio 1).
- fornendo ripetutamente alla funzione `frequenza` varie liste di numeri casuali, verificare che:
 - la somma dei numeri nelle lista restituita sia uguale alla lunghezza della lista passata come argomento;
 - la lista restituita contenga solo numeri non negativi.

Esercizio 4 (5 punti)

Sia dato un file di testo chiamato `dati.txt`. Ogni linea del file contiene un numero intero compreso tra 0 e 9. Scrivere un programma Python che, dopo aver letto il file, visualizza sul terminale un grafico a barre orizzontali con le frequenze assolute dei numeri da 0 a 9. Ad esempio, se il file `dati.txt` contiene le seguenti righe:

```
1
2
3
1
9
9
7
1
```

il programma visualizzerà qualcosa di simile a

```
0
1 ***
2 *
3 *
4
5
6
7 *
8
9 **
```

in cui il numero di asterischi è uguale al numero di occorrenze del numero corrispondente nel file `dati.txt`.

Suggerimento: si utilizzi la funzione `frequenza` dell'esercizio 1.

Esercizio 5 (5 punti)

Spiegare il funzionamento dell'algoritmo di ricerca binaria ed illustrare i passi che compie quando è applicato alla lista `[2, 4, 10, 32, 55, 90, 100]` per cercare il valore 55.

Esercizio 6 (8 punti)

Si consideri il seguente codice Python (che, per curiosità, implementa la “funzione 91” di McCarthy):

```
1 def mc91(n):
2     if n > 100:
3         return n-10
4     else:
5         x = mc91(n+11)
6         return mc91(x)
```

Eeguire passo passo la funzione quando viene chiamata come `mc91(99)`.