

Report W7 D1

1. Introduzione.

L'obiettivo di questo esercizio è scrivere una funzione in Python che, data una lista di parole in ingresso (lista A), restituisca una lista di interi (lista B), dove ogni elemento rappresenta la lunghezza della parola corrispondente in A.

2. Spiegazione.

Il codice fornito ha l'obiettivo di calcolare la lunghezza dei nomi di alcune nazioni e memorizzarli in una lista. Tuttavia, presenta un piccolo errore nella stampa. Di seguito analizziamo il codice e proponiamo una versione corretta.

3. Analisi profonda.

- Il codice utilizza un ciclo for che scorre attraverso la lista nazioni.
- Per ogni elemento della lista, calcola la lunghezza con len(nazione).
- Attraverso lunghezza.append aggiungiamo alla lista lunghezza la lunghezza delle parole

4. Codice.

```
1 nazioni = ["Russia", "Canada", "Uzbekistan", "Zimbabwe"]
2 lunghezza = []
3
4 for nazione in nazioni:
5     print("La nazione è:", nazioni, "-La lunghezza è:", len(nazione))
6     lunghezza.append(len(nazione))
7
8 print(lunghezza)
```

```
La nazione è: ['Russia', 'Canada', 'Uzbekistan', 'Zimbabwe'] -La lunghezza è: 6
La nazione è: ['Russia', 'Canada', 'Uzbekistan', 'Zimbabwe'] -La lunghezza è: 6
La nazione è: ['Russia', 'Canada', 'Uzbekistan', 'Zimbabwe'] -La lunghezza è: 10
La nazione è: ['Russia', 'Canada', 'Uzbekistan', 'Zimbabwe'] -La lunghezza è: 8
[6, 6, 10, 8]
```

Facoltativo

1. Obiettivo.

Scrivi una funzione generatrice di password. La funzione deve generare una stringa alfanumerica di 8 caratteri qualora l'utente voglia una password semplice, o di 20 caratteri ascii qualora desideri una password più complicata.

2. Spiegazione.

Il codice è un generatore di password casuali che permette all'utente di scegliere tra due livelli di complessità.

- Password semplice (10 caratteri).
- Password complessa (20 caratteri, inclusi caratteri speciali).

3. Analisi.

- Importazione della libreria random: il modulo random è utilizzato per prendere casualmente i caratteri tramite random.choices.
- Definizione dei caratteri disponibili associandoli a una stringa.
- Limitazione dei caratteri usando la K.
- Loop while true in caso l'utente sbaglia a scegliere possa sempre riprovarci.

4. Codice.

```
1 import random
2 alfabeto = "abcdefghijklmnopqrstuvwxyz"
3 alfabeto_maiuscolo= alfabeto.upper()
4 numeri = "1234567890"
5 caratteri_speciali = "!()/=[${}"
6
7 scelta = input("Se vuoi una password semplice semplice digita 1 se ne vuoi una
    complicata digita 2\n")
8 if scelta == "1":
9     caratteri = alfabeto + alfabeto_maiuscolo + numeri
10    password_facile = "".join(random.choices(caratteri, k=10))
11    print(password_facile)
12
13 elif scelta == "2":
14     caratteri = alfabeto + alfabeto_maiuscolo + numeri + caratteri_speciali
15     password_difficile = "".join(random.choices(caratteri, k=20))
16     print(password_difficile)
17 else :
18     print("Scelta non valida\n")
19
```

```
Se vuoi una password semplice semplice digita 1 se ne vuoi una complicata digita 2
3
Scelta non valida

Se vuoi una password semplice semplice digita 1 se ne vuoi una complicata digita 2
2
3Y5CgE6uWafnXK=r=5!c
```