Răspunsuri exerciții seminar 3:

Exercițiu A.

```
a = [1, 2, 4, 10]
b = a
a[2] = 'hello!'
print(b)
```

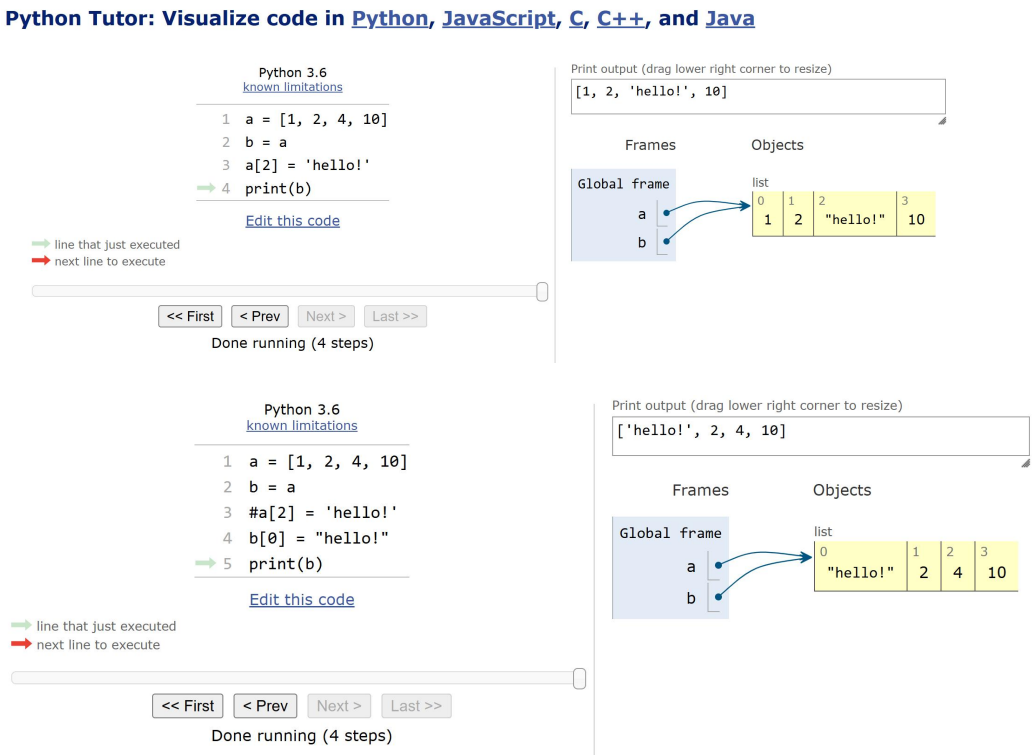
A.1. _____

Se afișează [1, 2, 'hello!', 10], prin atribuirea `b = a`, b ajunge să refere aceeași listă ca și a, astfel orice modificare făcută prin a asupra listei se va putea vedea și accesând b.

```
a = [1, 2, 4, 10]
b = a
b[0] = 'hello!'
print(a)
```

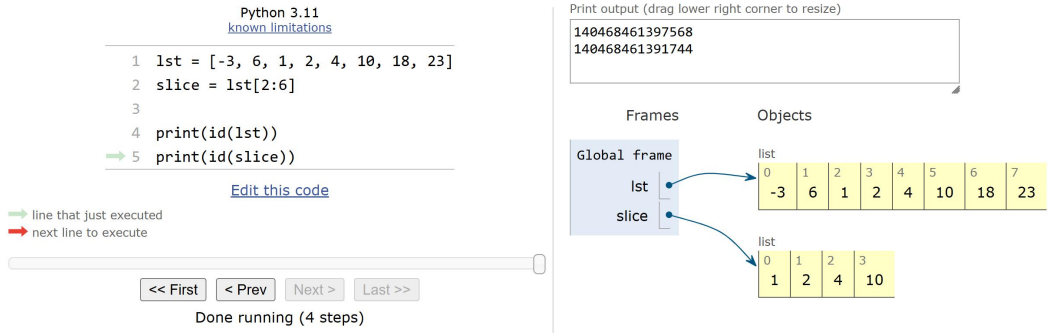
A.2. _____

Analog cu A.1. b și a referă aceeași listă din memorie.



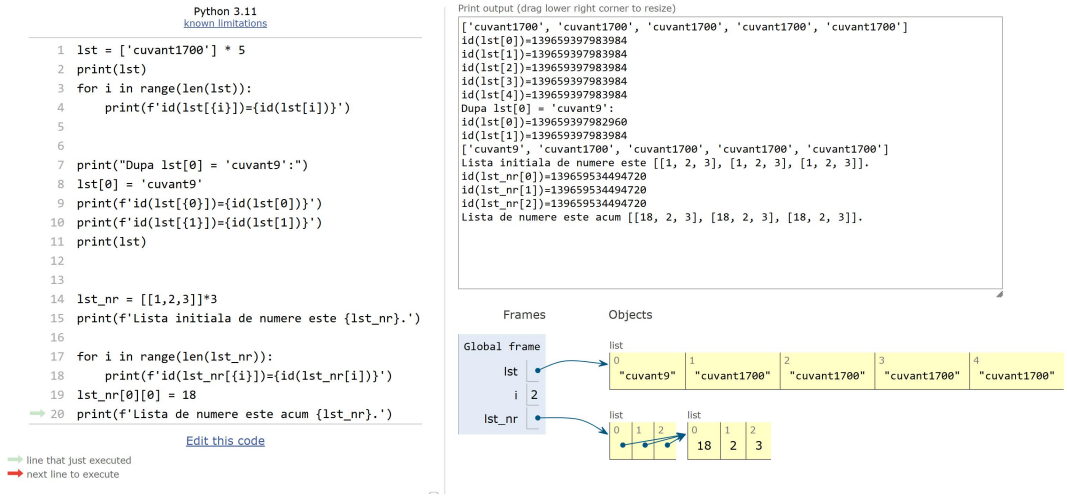
A.3. List slicing: https://www.pythonmorsels.com/slicing/

Se creează și afișează o nouă listă care conține elementele de la indexul 2 până la indexul 5 din lista originală: [1, 2, 4, 10]. Indexul de start se ia în considerare, dar nu și cel de final.



A. 4. Se afișează [‘a’, ‘a’, ‘a’, ‘a’, ‘a’]. Se creează o listă în care obiectul de tip str ‘a’ este multiplicat de 5 ori.

! Atenție: ce se întâmplă dacă obiectele din interiorul listei inițiale sunt **mutabile**?



A.5. Se afișează [5, 4, 3, 2, 1].

Read more about range here

A. 6. Se afișează [‘Ana’, ‘are’, ‘mere’].

`list.extend(iterable)`

Extend the list by appending all the items from the iterable. Equivalent to `a[len(a):] = iterable`.

A. 7. Se afișează 3.

Read more about tuple unpacking here.

A. 8. Se afișează numărul de la ultimul index al listei: -5. Dacă am încerca să afișăm lst[-2], s-ar afișa valoarea 20, întrucât acesta este penultimul element al listei.

Read more about negative indexing here.

A. 9. Se afișează True, se verifică dacă ‘E’ există în tuplu.

Extra: ce se afișează în urma executării următoarelor instrucțiuni?

```
participants = ('A', 'B', 'V', 'E')
print('E' in participants)

participants_l = ['A', 'B', 'V', 'E']
print('E' in participants_l)

participants_s1 = 'ABVE'
print('E' in participants_s1)

participants_s2 = 'ABVCDEFG'
print('E' in participants_s2)

participants_s3 = 'ABCDEFG'
print('CDE' in participants_s3)

participants_modified = ['A', 'B', 'V', 'Eva']
print('E' in participants_modified)
```

A.10. Se șterge ultimul element, 9.

`list.remove(x)`

Remove the first item from the list whose value is equal to *x*. It raises a `ValueError` if there is no such item.

Cum ar trebui să modificăm codul pentru a se șterge lista de la indexul 1 din listă?

B.

```
def count_character_frequencies(sentence: str) -> dict:
    """
    Functia calculeaza frecventa caracterelor dintr-o propozitie data
    :param sentence: propozitia data
    :return un dictionar care contine perechi de caracter-frecventa
    """
    frequency_dict = {}
    for character in sentence.lower():
        if character in frequency_dict:
            frequency_dict[character] += 1
        else:
            frequency_dict[character] = 1

    return frequency_dict


test_sentence1 = "Ana are mere."
freq_dict_test1 = count_character_frequencies(test_sentence1)
assert (freq_dict_test1['a'] == 3)
assert (freq_dict_test1[' '] == 2)

test_sentence2 = ""
freq_dict_test2 = count_character_frequencies(test_sentence2)
assert (len(freq_dict_test2) == 0)
```