

TEMA - Problemele care nu sunt rezolvate în timpul seminarului

PROBLEME CU COMPREHENSIIUNE

1. Se citește o propoziție cu cuvintele separate prin spații (unul sau mai multe). Să se creeze o listă cu cuvintele care încep cu o vocală (folosind și comprehensiune)
2. Se consideră o listă de liste (matrice) `ls` (de exemplu `[[4,7, 3], [3,1,20], [5,2,11]]`). Să se creeze o listă cu toate elementele din `ls` ordonate crescător (pentru exemplu `[1, 2, 3, 3, 4, 5, 7, 11, 20]`). (folosind și comprehensiune)
3. Cifrul lui Cezar (folosind comprehensiune)
 - a) Să se creeze în memorie o listă cu **literele mici** ale alfabetului folosind comprehensiune și un cuvânt având ca litere toate literele mici ale alfabetului (există `string.ascii_lowercase`).
 - b) Se citește un text ce conține numai **litere mici** ale alfabetului englez și semnele de punctuație uzuale și un număr natural k . Să se creeze în memorie și să se afișeze textul cifrat cu cifrul lui Cezar, prin care fiecare **literă** dintr-un text dat este înlocuită cu litera aflată peste k poziții la dreapta în alfabet în mod circular (valoarea k reprezintă cheia secretă comună pe care trebuie să o cunoască atât expeditorul, cât și destinatarul mesajului criptat).
4. Păsărească (folosind și comprehensiune) - Se citește de la tastatură un text. Se cere să se "traducă" în limba păsărească textul dat astfel: după fiecare vocală se adaugă litera `p` și încă o dată acea vocală (după `a`, `e`, `i`, `o`, `u` se adaugă respectiv `pa`, `pe`, `pi`, `po`, `pu`). Exemplu: "Ana are mere." devine "Apanapa aparepe meperepe.".
5. Se dă o propoziție care conține cuvinte separate prin separatorii uzuali (, ; :). O parte din cuvinte reprezintă numere naturale. Să se determine media aritmetică a cuvintelor din șir care sunt numere. Mai exact, rezultatul va fi un șir de forma: " $(nr_1+...nr_k)/n=media$ ", unde $nr_1,...,nr_k$ sunt numerele care apar în șir și media este afișată cu 2 zecimale.

OPERATII LISTE <https://docs.python.org/3/tutorial/datastructures.html>

6. Care atribuiri sunt incorecte + ce fac cele corecte pentru lista `ls=list(range(7))`

```
ls[4:5]=45
```

```
ls[4:5]="ab"
```

```
ls[4:5]=ls[5:4]
```

```
ls[4:5]=ls[3]
```

```
ls[4:5]=ls[3:]
```

```
ls [4:5]=[ls[3]]
```

7. Ce afișează următoarea secvență de cod?

```
ls=[0,0]
```

```
matr=[]
```

```
matr[:]=[ls]
```

```
matr[1:]=[[1,1]]
```

```
ls[0]=3
```

```
print(matr)
```

8. Dată o listă de numere naturale și un număr k , propuneți mai multe modalități de a șterge primele k elemente din listă
9. Se dau două liste l_1 și l_2 de lungime n . Să se înlocuiască elementele de pe poziții pare din l_1 cu cele de pe poziția corespunzătoare din l_2 folosind feliere (slice)
10. Se dă o listă de numere naturale. Să se șteargă din listă subsecvența delimitată de primele două zerouri din listă (inclusiv zerourile)
11. Se dă o listă de numere naturale. Să se șteargă din listă toate zerourile.
12. Se dă o listă de numere naturale și un număr natural k . Să se elimine din listă subsecvența de lungime k de sumă minimă (dacă sunt mai multe se va elimina prima = cea mai din stânga) – fără a folosi liste suplimentare
13. Se dă un vector de numere naturale ordonat crescător (toate elementele sale se vor da pe o linie separate prin spațiu). Să se elimine duplicatele din vector.
14. Se dă o listă de numere reale (toate elementele sale se vor da pe o linie separate prin spațiu). Să se insereze câte un 0 după fiecare element negativ (fără a folosi liste suplimentare)
15. Se citesc doi vectori (elementele fiecăruia fiind date pe o linie separate prin spațiu) de numere naturale de exact două cifre (elementele vectorilor nu sunt neapărat distincte). Să se determine dacă cei doi vectori sunt egali (folosind vector de frecvențe vs folosind sorted)