

Laboratorul 1

1. a) Instalați Python: <https://www.python.org/downloads/>.

Pentru Windows, se recomandă bifarea ☒ **Add python.exe to PATH**; se poate verifica instalarea în Command Prompt cu `>Python --version`.

b) Instalați JupyterLab https://jupyterlab.readthedocs.io/en/stable/getting_started/installation.html sau deschideți <https://colab.research.google.com/drive/>. Pentru Windows, JupyterLab se poate instala în Command Prompt și apoi porni cu comenzile

```
>pip install jupyterlab
>jupyter-lab
```

Observație: Instalarea JupyterLab este doar o recomandare. La laborator se poate folosi orice IDE. Versiuni online:

- <https://colab.research.google.com>
- <https://jupyter.org/try-jupyter/lab/>

c) Citiți:

- <https://docs.python.org/3/tutorial/introduction.html>
- <https://docs.python.org/3/tutorial/controlflow.html>

Pentru următoarele probleme, vom avea în vedere următoarele documentații:

<https://docs.python.org/3/library/random.html>

<https://docs.python.org/3/library/math.html>

<https://docs.python.org/3/library/itertools.html>

2. Pentru această problemă, vom utiliza următoarele funcții:

```
[ ]: from random import sample
     from math import factorial
     from itertools import permutations
```

a) Afișați o listă cu toate **permutările** cuvântului **word**.

b) Afișați numărul total al permutărilor cuvântului **word**.

c) Afișați o permutare aleatoare a cuvântului **word**.

3. Scrieți funcții care rezolvă problema anterioară pentru **aranjamente** și **combinări**, folosind:

```
[ ]: from random import sample
     from math import perm, comb
     from itertools import permutations, combinations
```

Exemple de apel:

```
[1]: aranjamente('word',2)
```

```
[1]: [('w', 'o'),  
      ('w', 'r'),  
      ('w', 'd'),  
      ('o', 'w'),  
      ('o', 'r'),  
      ('o', 'd'),  
      ('r', 'w'),  
      ('r', 'o'),  
      ('r', 'd'),  
      ('d', 'w'),  
      ('d', 'o'),  
      ('d', 'r')]
```

```
[2]: aranjamente('word',2,numar_total=True)
```

```
[2]: 12
```

```
[3]: aranjamente('word',2,aleator=True)
```

```
[3]: ['r', 'o']
```

```
[4]: combinari('word',2)
```

```
[4]: [('w', 'o'), ('w', 'r'), ('w', 'd'), ('o', 'r'), ('o', 'd'), ('r', 'd')]
```

```
[5]: combinari('word',2,numar_total=True)
```

```
[5]: 6
```

```
[6]: combinari('word',2,aleator=True)
```

```
[6]: ['o', 'd']
```

4. Afișați toate combinațiile cu repetiție de 4 elemente din grupul de litere ABCDE, folosind

```
[ ]: from itertools import combinations_with_replacement
```

De exemplu: ABBE, BBBC.

▷ Combinații cu repetiție de k elemente din mulțimea M : se aleg k elemente din M , cu repetiție (după fiecare alegere elementul este pus înapoi în mulțime), ordinea de extragere nu contează.

5. (Suplimentar) Scrieți un program care afișează în câte moduri se pot așeza 5 persoane pe 12 scaune astfel încât între ele să existe cel puțin un scaun liber.