

# Labor 1

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

Für Windows ☒ **Add python.exe to PATH**; man überprüft in Command Prompt mit `>Python --version`.

**b)** JupyterLab installieren [https://jupyterlab.readthedocs.io/en/stable/getting\\_started/installation.html](https://jupyterlab.readthedocs.io/en/stable/getting_started/installation.html).

Für Windows Windows in Command Prompt: `>pip install jupyterlab` und dann `>jupyter-lab`.

▷ Die Verwendung von JupyterLab ist nur eine Empfehlung. Im Labor können Sie auch eine andere Entwicklungsumgebung verwenden.

**c)** Python Tutorial (fakultativ):

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

► In diesem Labor werden benutzt:

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

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

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

```
import random
from random import sample
import math
from math import factorial, perm, comb
import itertools
from itertools import permutations, combinations
help("random.sample")
help("math.perm")
#help("itertools.permutations")
print(perm(4, 4))
print(perm(4, 2))
#help("math.comb")
```

**2.**

**a)** Man erstelle eine Liste mit den Permutationen von ABC.

**b)** Welches ist die gesamte Anzahl der Permutationen von ABC?

Hinweis: `math.perm`

**c)** Man generiere eine zufällige Permutation von MATHE.

Hinweis: `random.sample`

**d)** Man generiere eine zufällige Variation mit 3 Buchstaben aus dem String MATHE.

Hinweis: `random.sample`

**e)** Man generiere alle Variationen (d.h. Anordnungen ohne Wiederholung, mit Berücksichtigung der Reihenfolge) mit 3 Buchstaben aus dem String MATHE (d.h. alle Variationen je 3 Buchstaben aus dem String MATHE).

**f)** Welches ist die gesamte Anzahl der Variationen je 3 Buchstaben von MATHE?

**g)** Man generiere alle Kombinationen (d.h. alle Anordnungen ohne Wiederholung, ohne Berücksichtigung der Reihenfolge) mit 3 Buchstaben aus dem String MATHE (d.h. alle Kombinationen je 3 Buchstaben aus dem String MATHE).

Hinweis: `itertools.combinations`

**h)** Welches ist die gesamte Anzahl der Kombinationen je 3 Buchstaben von MATHE?

Hinweis: `math.comb`

- Eine **Permutation mit Wiederholung** ist eine Anordnung von Objekten, von denen manche nicht unterscheidbar sind.

Z.B. Alle Permutationen von

a) AABB sind: AABB, ABAB, ABBA, BAAB, BABA, BBAA ;

b) AABC sind: AABC, AACB, ABAC, ABCA, ACAB, ACBA, AAC, BACA, BCAA, CAAB, CABA, CBAA .

Für `more_itertools` wird `pip install more-itertools` benötigt!

► Man fülle die fehlenden Zeilen in Python aus:

```
import more_itertools
from more_itertools import distinct_permutations

M=list(distinct_permutations("AABB"))
print(M)
...
print("Anzahl Permutationen von AABB mit Wiederholung:",m)

for p in distinct_permutations("1112"):
    print("".join(p))
...
print("Anzahl Permutationen von 1112 mit Wiederholung:",n)
```

- Bei einer **Kombination mit Wiederholung** werden aus  $n$  verschiedenen Objekten  $k$  Objekte ohne Beachtung der Reihenfolge ausgewählt, wobei Objekte auch mehrfach ausgewählt werden können.

Z.B. Alle Kombinationen von ABC je 2 mit Wiederholung sind: AA, AB, AC, BB, BC, CC (hier ist  $n = 3$  und  $k = 2$ ).

► Man fülle die fehlenden Zeilen in Python aus:

```
import itertools
from itertools import combinations_with_replacement
print("Alle Kombinationen von ABC je 2, mit Wiederholung")
...

print("Anzahl Kombinationen von ABC je 2 mit Wiederholung:",k)
```

**Übung:** Wie viele Möglichkeiten gibt es 6 rote Kugeln in 4 Kartons aufzuteilen? Manche Kartons können leer bleiben. Man zähle alle möglichen Anordnungen auf. Hinweis: Wir bezeichnen die vier Kartons mit 1,2,3,4. Eine mögliche Anordnung ist: [1,1,2,3,3,3], d.h. 2 Kugeln im Karton "1", eine Kugel im Karton "2", 3 Kugeln im Karton "3" und keine Kugel im Karton "4".

# Kombinatorik Formeln

