

02b_Miniuebungen

0.1 Mini-Aufgaben zur Überprüfung des Verständnis: Boolesche Variablen

Wahrheitstabellen sind der Schlüssel zum Verständnis logischer Verknüpfungen. Tatsächlich ist eine vollständige Wahrheitstabelle sogar ein mathematischer Beweis einer logischen Aussage, und damit unanfechtbar.

Programmieren Sie die Ausgabe einer Wahrheitstabelle für das folgende **de-Morgansche Gesetz**:

$$\neg(a \vee b) = \neg a \wedge \neg b$$

Hierbei ist \neg die mathematische Notation für die Negation (not), \wedge ist das Symbol für and, und \vee das Symbol für or.

```
[1]: # Lösung
```

0.1.1 Abbruchbedingungen von Schleifen

Logische Ausdrücke sind eng verknüpft mit Abbruchbedingungen von while Schleifen. Wir betrachten die vollkommen künstliche Situation, in einer Schleife so lange um Eins hochzuzählen, bis entweder die Kontrollvariable den Wert 100 erreicht hat, oder das Quadrat der Kontrollvariable den Wert 81 überschreitet. Realisieren Sie dies, indem Sie die folgenden Codeschnipsel ergänzen.

```
[2]: i = 1
while insert_condition_here:
    print(i)
    i = i + 1
```

NameError

Traceback (most recent call last)

```
/tmp/ipykernel_19273/2505608879.py in <module>
1 i = 1
----> 2 while insert_condition_here:
3     print(i)
4     i = i + 1
```

NameError: name 'insert_condition_here' is not defined

0.2 Impressum

0.2.1 Programmierkurs Python, Dominik Göddeke <https://www.ians.uni-stuttgart.de>,
Universität Stuttgart

Version vom April 2023

Lizenziert unter der Creative Commons Namensnennung 4.0 International Lizenz



Veröffentlicht auf <https://zoerr.de>, (alle Rechte am Logo vorbehalten)



Gefördert durch die Stiftung Innovation in der Hochschullehre. (alle Rechte am Logo vorbehalten)



Gefördert mit Mitteln der Deutschen Forschungsgemeinschaft (EXC 2075 - 390740016) im Rahmen der Exzellenzstrategie.

[]: