

In der Ausbreitung einer Pandemie spielt die Verdoppelung in Tagen eine entscheidende Rolle. Eine Übersicht zu Corona erhältst du in Christian Endt und Benedict Witzemberger, Alle Daten zur Pandemie, in: SZ online v. 25. März 2020, 9:45 Uhr, <https://www.sueddeutsche.de/wissen/coronavirus-zahlen-daten-fakten-1.4844448>. Dort heißt es unter:

Woran lässt sich erkennen, ob die Trendwende erreicht ist?

Es gibt verschiedene Möglichkeiten, um die Ausbreitung einer Epidemie mathematisch abzubilden. Am eingängigsten ist die Verdopplungszeit: Das ist die Zeitspanne, nach der sich die Zahl der Infizierten jeweils verdoppelt hat. Dabei darf man nicht unterschätzen, wie stark sich exponentielles Wachstum beschleunigt: Beispielsweise sind nur zehn Verdopplungszyklen nötig, um von tausend Fällen auf eine Million Fälle zu kommen. Bei einer Verdopplungszeit von sieben Tagen wäre das nach zehn Wochen erreicht, bei einer Verdopplungszeit von drei Tagen bereits nach einem Monat.

Aufgabe:

Dieser Code gibt dir (in [Jupyter Notebook](#)) die Verdoppelung täglich, inkl. Diagramm:

```
from collections import defaultdict
result=defaultdict(int)
def corona():
    summe=0
    corona=1
    for i in range(1,30):
        corona=corona*2
        result[i]+=corona
    return result
print(corona())

import matplotlib.pyplot as plt
%matplotlib inline
result
plt.plot(list(result.keys()), list(result.values()))
plt.ylabel("Kranke")
plt.xlabel("Tage")
plt.title("Entwicklung Corona")
plt.show()
```

Schreibe eine Funktion `zyklen(z)`, die dir die Verdoppelung in beliebig angebbaren Zyklen (i.e. das Argument `z` für `zyklen(z)`) liefert, und die nach einer Million Fälle abbricht. Diese Funktion soll ein Dictionary mit `Tagen=keys` und `Infektionen = values` zurückgeben. Dieses Dictionary verwendet dann `pyplot` für ein Diagramm