



Lektion 3 : Programmbeispiele

Anwendung : Primzahltest

In dieser Anwendung von Einheit 3 verwenden Sie die in den vorherigen Lektionen erworbenen Konzepte, um Algorithmen zu programmieren, mit denen Sie Ihr Wissen über Zahlen und insbesondere Primzahlen verfeinern können.

Lernziele :

- Anwendung von Tests und Schleifen in einem Python-Programm über **Primzahlen**

Eine **Primzahl** hat als Teiler nur die 1 und sich selbst.

Zum Beispiel :

- 1 ist keine Primzahl, da sie nur einen Teiler (sich selbst) hat.
- 7 ist eine Primzahl mit den Teilern 1 und 7.
- 8 ist keine Primzahl, denn sie hat die 4 Teiler 2,4,8 und 1

Die ersten Primzahlen sind : 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, ...

Es gibt unendlich viele Primzahlen

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

Mit einem Programm soll die **2020. Primzahl** bestimmt werden. Es handelt sich dabei nicht um eine Funktion wie bisher, da zum Start kein Funktionsname aufgerufen werden muss.

Das Programm

- Legen Sie ein neues Skript « **PRIMZ** » an.
- Das Modul « **math** » muss vorab geladen werden.
- Das Programm enthält eine Funktion **prim(n)**, die den Test auf Primzahl durchführt. Die zu testende Zahl **n** wird dabei schrittweise euklidisch von **2** bis $\lfloor \sqrt{n} + 1 \rfloor$ durch **k** dividiert.
- Ist **r = 0 (keine Primzahl)**, so ist **prim(n) = 0**, andernfalls **1**.
- Im Hauptteil des Programmes werden die Zahlen **N** erzeugt, und die Anzahl der Primzahlen **np** gezählt, bis die **Grenze 2020** erreicht ist. Die Anweisungen des Hauptteiles befinden sich am Ende des Programmes und werden **nicht eingerückt** !
- Es dauert dann etwa 30s, bis das Ergebnis dargestellt wird.
- Die Rechenzeit lässt sich verkürzen, indem man die geraden Zahlen bei der Primzahlbestimmung überspringt.

```

EDITOR: PRIMZ
PROGRAM LINE 0006
from math import *
def prim(n):
    if n<=1:
        return 0
    for k in range(2,floor(sqrt(n)
)+1):
        if n%k==0:
            return 0
    return 1
N=2
np=1
while np<2020:
    N=N+1
    np=np+prim(N)
print(N)_

```

```

PYTHON SHELL
>>> # Shell Reinitialized
>>> # Running PRIMZ
>>> from PRIMZ import *
17573
>>> |

```