



Lektion 4 : Verwendung des Moduls tiplotlib

Übung 2 : Den Grafen einer Funktion erstellen

In dieser zweiten Lektion der Lektion 4 wird gezeigt, wie man eine Funktion mit Hilfe der Python-Bibliothek **tiplotlib** grafisch darstellen kann.

Lernziele :

- Den Grafen einer Funktion darstellen.
- Verwendung einer **for**-Schleife.
- Einrichten der grafischen Darstellung.

Der Graph der Funktion wird punktweise erstellt für **N** x-Werte in einem Intervall [a ; b]. Es werden Befehle aus dem Modul **tiplotlib** verwendet. Das Programm, das erstellt wird, wird sehr allgemein gehalten sein, so dass es für andere Beispiele ebenfalls verwendet werden kann.

Lehertipp : Wer mit dem Konzept der For-Schleife nicht vertraut ist, kann sich in den Lektionen 1 bis 3 informieren.

- Erstellen eines neuen Programmes U4SB2.
- Importieren des Moduls **tiplotlib**, das man erhält, indem man **f(x)** drückt und dann **5** : **tiplotlib** wählt.
- Definieren der Funktion $g: x \mapsto x^2 - 3x + 4$.

```

EDITOR: U4SB2
PROGRAM LINE 0004
import tiplotlib as plt
def g(x):
--return x**2-3*x+4

```

Lehertipp : Die Liste der x-Werte **lx** wird mit einer For-Schleife erzeugt. Man erhält den Befehl, indem man die Tasten **f(x)** drückt und dann **4** : **for i in range(size)** wählt.

Die Liste der y-Werte **ly** wird im Anschluss erzeugt. Dazu wird der Befehl **7** : **for i in list** verwendet.

Nun kann das Programm geschrieben werden. Dazu gehören auch die üblichen Anweisungen :

- Säubern des Bildschirms : **plt.cls()**.
- Einstellen der Window-Parameter : **plt.window(xmin, xmax, ymin, ymax)**.
- Einschalten der Koordinatenachsen : **plt.axes(« on »)**.
- Bezeichnung der Koordinatenachsen : **plt.labels(« x », « y »)**.
- Festlegen des Grafikstils : **plt.plot(lx,ly, « + »)**.
- Zeichnen der Grafik : **plt.show_plot()**.

Alle Befehle gehören zum Modul **tiplotlib**. Die Befehle zur Einstellung der Grafik befinden sich im Menü **SetUp**, die zum Zeichnen im Menü **Draw**.

```

EDITOR: U4SB2
PROGRAM LINE 0002
import tiplotlib as plt
def g(x):_
--return x**2-3*x+4
def graf(f,a,b,N):
--lx=[a+i*(b-a)/N for i in range(N+1)]
--ly=[f(a+i*(b-a)/N) for i in range(N+1)]
--plt.cls()
--plt.window(-0.5,3.5,-1,5)
--plt.color(0,0,0)

```

```

EDITOR: U4SB2
PROGRAM LINE 0015
--ly=[f(x) for i in lx]:
--plt.cls()
--plt.window(-0.5,3.5,-1,5)
--plt.color(0,0,0)
--plt.axes("on")
--plt.labels("x","y")
--plt.color(255,0,0)
--plt.plot(lx,ly,"+")
--plt.show_plot()

```

Bemerkung : Der Bereich für die Farben r, g, b geht von 0 bis 255. Die Einstellungen für die Farbe stehen direkt vor den Zeichenbefehlen.



10 Minuten Coding

TI-84 PLUS CE-T PYTHON EDITION

LEKTION 4 : ÜBUNG 2

LEHRERMATERIAL

Das Programm wird ausgeführt, indem man nach **Run** die Taste `[vars]` drückt und die Funktion **graf()** auswählt.

Die Zeichnung selbst soll dann erfolgen im Intervall $[0 ; 3]$ mit 25 Punkten.

Lehertipp : Sollen mehr Punkte berechnet werden, so sollte als Darstellung der Punkt anstelle des Plus verwendet werden..

Mögliche Erweiterungen :

- Zusätzlich ein Gitter anzeigen.
- Die Funktion ändern oder mehrere Funktionen darstellen.
- Veränderung der Anzahl der Punkte und die Auswirkung auf den Grafen.

```
PYTHON SHELL
>>> # Shell Reinitialized
>>> # Running U4SB2
>>> from U4SB2 import *
>>> graf(g,0,3,25)
```

