

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.

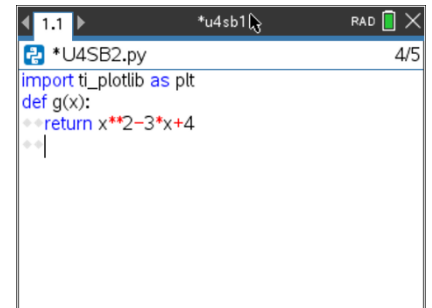
Lernziele :

- 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** aus dem Menü.
- Definieren der Funktion $g: x \mapsto x^2 - 3x + 4$.



```

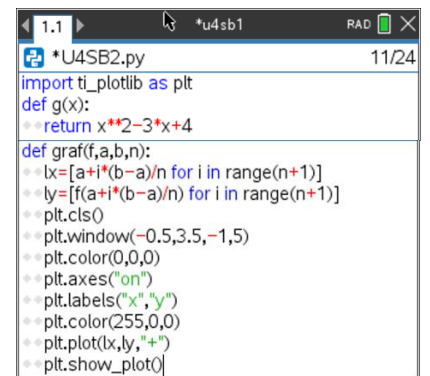
1.1 *U4sb1 RAD 4/5
*U4SB2.py
import tiplotlib as plt
def g(x):
    return x**2-3*x+4

```

Lehertipp : Die Liste der x-Werte **lx** wird mit einer For-Schleife erzeugt. Man muss den Befehl dazu ausschreiben, da er sonst falsch platziert wird. Die Liste der y-Werte **ly** wird auf gleiche Weise im Anschluss erzeugt

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 »)**.
- Farbe der Grafik : **plt.color(255,0,0)**
- Festlegen des Grafikstils : **plt.plot(lx,ly, « + »)**.
- Zeichnen der Grafik : **plt.show_plot()**.



```

1.1 *U4sb1 RAD 11/24
*U4SB2.py
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)
    plt.axes("on")
    plt.labels("x","y")
    plt.color(255,0,0)
    plt.plot(lx,ly,"+")
    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**.

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

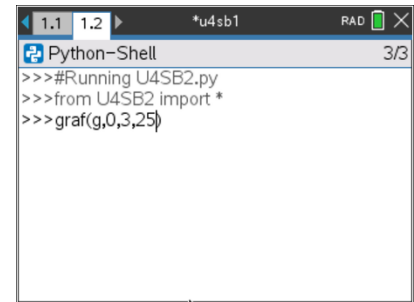
Bei der Programmausführung muss bei `var` die Funktion **graf()** ausgewählt werden.

Die Zeichnung selbst erfolgt dann im Intervall $[0 ; 3]$ mit 25 untereinander geradlinig verbundenen Punkten (Polygonzug).

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.



```
1.1 1.2 *u4sb1 RAD 3/3
Python-Shell
>>>#Running U4SB2.py
>>>from U4SB2 import *
>>>graf(g,0,3,25)
```

