

INFOFICHE : PYTHON - MATPLOTLIB

Algoritme

Python

CASIO

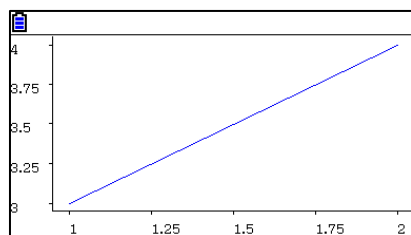


Menu Python

In Python is de bibliotheek Matplotlib beschikbaar om gegevens als grafieken te tekenen en te visualiseren.

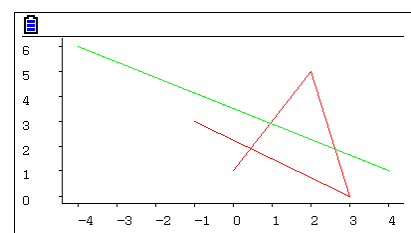
Typ manueel de tekst `from matplotlib.pyplot import *` in of zoek deze tekst in de catalogoog (SHIFT 4). Na het opladen is het mogelijk om alle functies van deze bibliotheek in Python of in de Shell te gebruiken.

```
MicroPython v1.9.4
|CASIO COMPUTER CO.,
|>>>from matplotlib.py
|>>>plot([1,2],[3,4])
|>>>show()
|>>>
|
| RUN | A↔a | CHAR
```



```
>>>plot([1,2],[3,4])
>>>show()
>>>show()
>>>plot([1,2],[3,4])
>>>show()
>>>plot([1,5,0,3], "red")
>>>
|
| RUN | A↔a | CHAR
```

```
>>>show()
>>>show()
>>>plot([1,2],[3,4])
>>>show()
>>>plot([1,2,3,4],[1,1,3,4])
>>>plot([-4,4],[6,1])
>>>show()
|
| RUN | A↔a | CHAR
```



PLOT :

Met de instructie `plot` kan een ononderbroken lijn tussen twee punten getekend worden. Voer na die instructie de lijst met de abscissen én de lijst met de ordinaatgetallen in. De getallen in de lijsten moeten gescheiden worden door een komma.

Een lijnstuk heeft als eindpunten de punten met coördinaten (1 ; 3) en (2 ; 4). Voer `plot([1,2],[3,4])` in om dit lijnstuk te tekenen.

Druk op EXE. Er gebeurt niets!

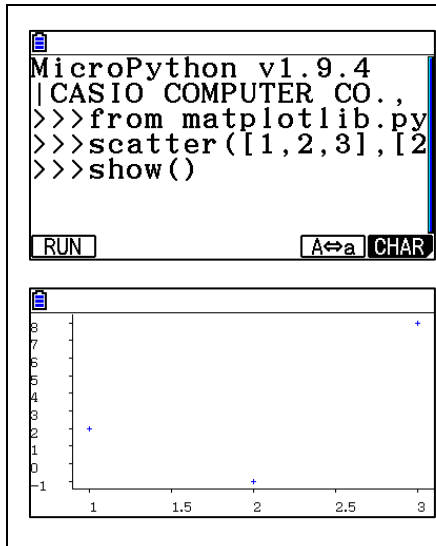
Dat komt omdat de instructie `show()` gebruikt moet worden om de grafiek te tekenen. Tenzij anders vermeld, wordt het kijkvenster automatisch aangepast aan de ingevoerde waarden. Druk op EXIT om terug te keren naar de Shell.

Met de instructie `plot` kunnen ook gebroken lijnen getekend worden. Bijvoorbeeld : een gebroken lijn die de vier punten met respectievelijke coördinaten (0 ; 1), (2 ; 5), (3 ; 0) en (-1 ; 3) verbindt.

Het is ook mogelijk om de tekenkleur van de gebroken lijn aan te passen door een derde argument toe te voegen (in het Engels en tussen aanhalingstekens). Hier is de kleur "red" gekozen voor de gebroken lijn.

Er kunnen verschillende `plot` instructies na elkaar ingevoerd worden, die dan daarna allemaal getekend worden. Het kijkvenster past zich automatisch aan.

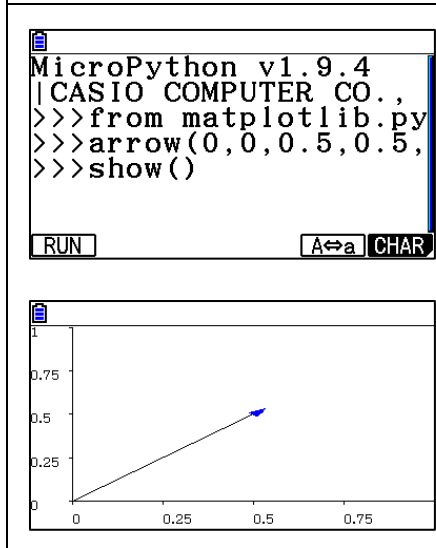
INFOFICHE



SCATTER :

Met de instructie *scatter* kan een puntwolkdiagram getekend worden. De argumenten zijn de lijst van de abscissen en de lijst van de ordinaatgetallen van de punten.

Voorbeeld : `scatter([1,2,3],[2,-1,8])`



ARROW :

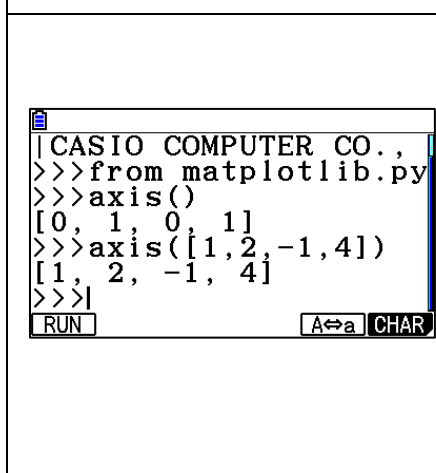
Met de instructie *arrow* kunnen vectoren getekend worden.

Deze instructie werkt met :

- 4 verplichte argumenten : de abscis van de oorsprong, het ordinaatgetal van de oorsprong, de abscis van het eindpunt en het ordinaatgetal van het eindpunt
- en eventueel nog twee andere argumenten: de breedte (*head_width*) en de lengte (*head_length*) van de pijl (Deze zijn optioneel maar wel nodig om de pijl van de vector te zien).

Standaard is de breedte van de pijl gelijk aan 0,003 en de lengte is 1,5 keer de breedte.

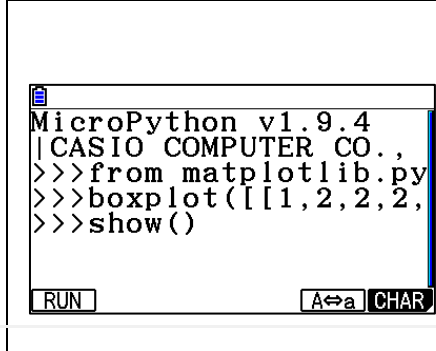
Let op : bij de instructie *arrow* past het kijkvenster zich niet automatisch aan.



AXIS :

Met instructie *axis* kan het grafische kijkvenster manueel aangepast worden. Wanneer deze instructie zonder argumenten gebruikt wordt, dan wordt het kijkvenster ingesteld met de minimale en maximale waarde van de abscissen en de ordinaatgetallen van het kijkvenster (standaard [0,1,0,1]).

Om het kijkvenster aan te passen, volstaat het om als argument de lijst met de minima en de maxima van de abscissen en de ordinaatgetallen van het gewenste venster op te geven. In het voorbeeld hiernaast wordt het kijkvenster aangepast zo dat $x_{min}=1$, $x_{max}=2$, $y_{min}=-1$ en $y_{max}=4$.



BOXPLOT :

Met de instructie *boxplot* kunnen boxplots van 1 of meerdere reeksen waarden weergegeven worden. Deze reeksen moeten in éénzelfde lijst opgenomen worden.

Voer bijvoorbeeld `boxplot([[1,2,2,2,2,3,-1,5,2],[2,3,6,7,0,0,0]])` in om de boxplot van deze twee reeksen waarden te bekomen.