Recherche d'extremum d'une fonction.

Soit f la fonction définie sur $\mathbb{R}$ par $f(x) = x^3 - x^2 - 5x + 3$.

Soit a et b deux réels, $a < b$ et $d \in \mathbb{R}_+^*$. Soit $A = \{a + kd | k \in \mathbb{N} \ et \ a + kd \leq b\}$, donc $A = \{a, a + d, a + 2d, .., a + nd\}$ avec n le plus grand entier tel que $a + nd < b$.

On se propose dans cette activité de rechercher les réels $ymin$ et $ymax$ vérifiant :

$ymin = min\{f(x)|x \in A\}$ ainsi que $ymax = max\{f(x)|x \in A\}$.

Cela correspond à la recherche d'une valeur approchée des extrema par balayage de f sur $[a, b]$ (on ne prend en compte que les valeurs de $x \in A$).

Pour que ces valeurs $(ymin, ymax)$ correspondent à des valeurs approchées convenables des extrema de f sur $[a, b]$ il faudra prendre un pas assez petit et une fonction assez régulière (ce qui est le cas dans la plupart des fonctions étudiées au lycée).

Dans un script MINMAX

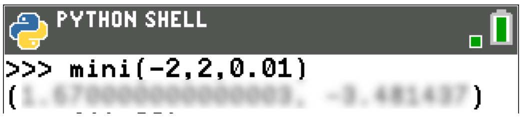
1°) Ecrire une fonction f qui prend en argument le réel x et qui renvoie $x^3 - x^2 - 5x + 3$.

On testera cette fonction en calculant les images ci-contre.

```
>>> f(-2)
1
>>> f(1)
-2
>>> f(3/2)
-3.375
```

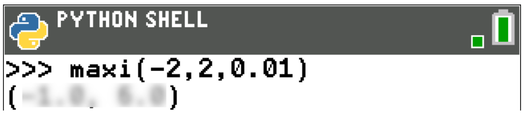
2°) Ecrire une fonction `mini` qui prend comme arguments les réels a, b et d et qui, en incrémentant x avec un pas de d, renvoie le couple $(xmin, ymin)$ que nous avons défini ci-dessus.

Quel est alors le minimum de f sur $[-2; 2]$ obtenu par cette méthode ?



3°) Ecrire une fonction maxi qui prend comme arguments les réels a, b et d et qui, en incrémentant x avec un pas de d, renvoie le couple $(xmax, ymax)$.

Quel est alors le maximum de f sur $[-2; 2]$ obtenu par cette méthode ?



Recherche d'extremum d'une fonction.

Fonction `f`

1°) Le fait de définir la fonction f dans le script permettra de la modifier facilement sans avoir à modifier les fonctions qui vont suivre et qui font appel à cette fonction f.

```
ÉDITEUR : MINMAX
LIGNE DU SCRIPT 0002

def f(x):
  y=x**3-x**2-5*x+3
  return y
```

Fonction `mini`

2°) On va utiliser 4 variables : `x`, `y=f(x)`, `xmin` et `ymin`.

`x` va être initialisé à `a` et incrémenté de `d` à chaque tour de boucle. La boucle se poursuit tant que `x` est inférieur ou égal à `b`.

On initialise `xmin` à a et `ymin` à `f(a)`.

Dès que `y` (qui vaut `f(x)`) sera inférieur à `ymin` alors on trouve une valeur plus petite que `ymin`, il faut remplacer `ymin` par cette valeur `y`.

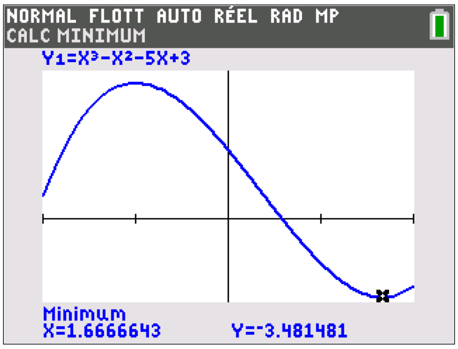
A la fin on renvoie le couple `xmin,ymin`.

A l'exécution de la fonction `mini` avec une précision (sur `x`) de 0,01 on trouve que le minimum de f sur $[-2; 2]$ vaut $-3{,}48$ et il est atteint en $1{,}67$.

Les résultats correspondent bien à ce qu'on peut trouver à l'aide du module recherche de minimum et maximum de la TI-83 Premium CE.

```
ÉDITEUR : MINMAX
LIGNE DU SCRIPT 0016
def mini(a,b,d):
  x=a
  xmin=a
  ymin=f(xmin)
  while x<=b:
    x=x+d
    y=f(x)
    if y<ymin:
      xmin=x
      ymin=y
  return xmin,ymin
```

```
PYTHON SHELL
>>> mini(-2,2,0.01)
(1.670000000000003, -3.481437)
>>> f(1.66)
-3.481304
>>> f(1.67)
-3.481437000000001
>>> f(1.68)
-3.480768000000001
```



Fonction `maxi`

3°) Il suffit de changer la condition du `if`. On a changé le nom des variables, mais cela n'était pas nécessaire en soit.

```
ÉDITEUR : MINMAX
LIGNE DU SCRIPT 0028
def maxi(a,b,d):
  x=a
  xmax=a
  ymax=f(xmax)
  while x<=b:
    x=x+d
    y=f(x)
    if y>ymax:
      xmax=x
      ymax=y
  return xmax,ymax
```

```
PYTHON SHELL
>>> maxi(-2,2,0.01)
(-1.0, 6.0)
>>> f(-0.99)
5.999601
>>> f(-1)
6
>>> f(-1.01)
5.999599
```