

Pour aller plus loin :

Indiquer les réponses obtenues en entrant les différentes lignes suivantes dans une console Python.

```
>>> 1+2==3
.....
>>> 1+2
.....
>>> 0.1+0.2==0.3
.....
>>> 0.1+0.2
.....
>>> from decimal import Decimal
>>> Decimal(str(0.1))+Decimal(str(0.2))==Decimal(str(0.3))
.....
>>> Decimal(str(0.1))+Decimal(str(0.2))
.....
>>> float(Decimal(str(0.1))+float(Decimal(str(0.2))))==float(Decimal(str(0.3)))
.....
>>> float(Decimal(str(0.1))+Decimal(str(0.2)))==float(Decimal(str(0.3)))
.....
>>> float(Decimal(str(0.1))+Decimal(str(0.2)))
.....
```

La fonction *Decimal()* du module *decimal* permet de manipuler les nombres flottants avec une plus grande précision.

Les affichages avec cette fonction comportent par défaut 30 chiffres alors qu'habituellement ils ne proposent que 17 chiffres.

Le programme suivant permet donc de définir des fonctions « plus performantes ».

```
from decimal import Decimal
from math import sqrt

def nbre(x):
    return Decimal(str(x))

def milieu(xA,yA,xB,yB):
    xM=(nbre(xA)+nbre(xB))/2
    yM=(nbre(yA)+nbre(yB))/2
    return(float(xM),float(yM))

def parallélogramme(xA,yA,xB,yB,xC,yC,xD,yD):
    if milieu(xA,yA,xC,yC)==milieu(xB,yB,xD,yD):
        return(True)
    else:
        return(False)

def longueur(xA,yA,xB,yB):
    return sqrt((nbre(xB)-nbre(xA))**2+(nbre(yB)-nbre(yA))**2)

def rectangle(xA,yA,xB,yB,xC,yC,xD,yD):
    if parallélogramme(xA,yA,xB,yB,xC,yC,xD,yD)==True and longueur(xA,yA,xC,yC)==longueur(xB,yB,xD,yD):
        return(True)
    else:
        return(False)
```