

Equation réduite de droite.



Soit (O, \vec{i}, \vec{j}) un repère du plan. Soient $A \begin{pmatrix} x_A \\ y_A \end{pmatrix}$, $B \begin{pmatrix} x_B \\ y_B \end{pmatrix}$ et $C \begin{pmatrix} x_C \\ y_C \end{pmatrix}$ où x_A, y_A, x_B, y_B, x_C et y_C sont des réels. On suppose que $A \neq B$.

Le but de cette activité est de donner l'équation réduite de la droite (AB)

Dans un script DROITE

1°) Ecrire une fonction `oy` qui prend comme arguments x_A, x_B et qui renvoie : `True` si (AB) est parallèle à l'axe des ordonnées et `False` sinon.

2°) Ecrire une fonction `reduite` qui prend comme arguments x_A, y_A, x_B, y_B et qui renvoie :

`c` si l'équation réduite de (AB) est de la forme $x = c$

`m,p` si l'équation réduite de (AB) est de la forme $y = mx + p$

Application : Prendre $A \begin{pmatrix} 4 \\ 5 \end{pmatrix}$ et $B \begin{pmatrix} 8 \\ -1 \end{pmatrix}$ puis $A \begin{pmatrix} 4 \\ 5 \end{pmatrix}$ et $B \begin{pmatrix} 4 \\ 9 \end{pmatrix}$.

3°) On souhaite avoir un affichage de ce type (voir ci-contre) :

On va supposer que x_A, y_A, x_B, y_B sont des entiers.

Ecrire une fonction `reduite2` qui prend comme argument x_A, y_A, x_B, y_B et qui renvoie l'affichage (au format `string`) de l'équation réduite de (AB) .

Application : Reprendre les coordonnées des points A et B de la question 3.

4°) Comment faire pour afficher une fraction irréductible pour m et p ?

```
PYTHON SHELL
>>> oy(8,10)
False
>>> oy(7,7)
True
```

```
PYTHON SHELL
>>> reduite(4,5,8,-1)
(-1.5, 11.0)
>>> reduite(4,5,4,9)
4
```

```
PYTHON SHELL
>>> reduite2(4,5,8,-1)
'y = -1.5 * x +11.0'
>>> reduite2(4,5,4,9)
'x = 4'
```

```
PYTHON SHELL
>>> reduite3(4,5,8,15)
'y = 5/2 * x -5'
```

Equation réduite de droite.

Fonction `oy`

1°) On renvoie `True` lorsque `xa` et `xb` sont égaux. On rappelle que la comparaison est symbolisée par `==`.

```
ÉDITEUR : DROITE
LIGNE DU SCRIPT 0009
def oy(xa,xb):
    if xa==xb:
        return True
    else:
        return False
```

Fonction `reduite`

2°) Si la droite est parallèle à l'axe des ordonnées, c'est-à-dire si la fonction `oy` renvoie `True` alors on retourne `c` qui vaut `xa` (ou `xb` au choix).

Sinon on calcule le coefficient directeur et l'ordonnée à l'origine et on renvoie le couple `m, p`.

```
ÉDITEUR : DROITE
LIGNE DU SCRIPT 0016
def reduite(xa,ya,xb,yb):
    if oy(xa,xb):
        return xa
    else:
        m=(yb-ya)/(xb-xa)
        p=yb-m*xb
        return m,p
```

Fonction `reduite2`

3°) On reprend la même structure que la fonction précédente.

Cette fois on renvoie du texte qu'on formate pour faire apparaître l'écriture habituelle d'une équation réduite.

On utilise la fonction Python `str` qui permet de convertir un nombre en chaîne de caractères.

Dans certains cas l'affichage est agréable :

```
ÉDITEUR : DROITE
LIGNE DU SCRIPT 0023
def reduite2(xa,ya,xb,yb):
    if oy(xa,xb):
        return "x = "+str(xa)
    else:
        m=(yb-ya)/(xb-xa)
        p=yb-m*xb
        return "y = "+str(m)+" * x + "+str(p)
```

```
PYTHON SHELL
>>> reduite2(4,5,8,-1)
'y = -1.5 * x +11.0'
>>> reduite2(4,5,4,9)
'x = 4'
```

Mais dans d'autres cas, il est difficilement lisible :

```
PYTHON SHELL
>>> reduite2(3,4,14,12)
'y = 0.7272727272727273 * x +1.818181818181818'
```

On va résoudre ce problème dans la question suivante...

Equation réduite de droite.

Fonction `reduite3`

4°) La particularité de ce script est qu'on va utiliser la fonction `pgcd` que nous avons construit précédemment. Pour cela on indique qu'on souhaite utiliser les fonctions du script ARITHM :

```
from ARITHM import *
```

Il suffit après de simplifier numérateur et dénominateur de chaque fraction par leur `pgcd`.

On a aussi envisagé le cas où le dénominateur vaut 1 pour ne pas afficher $\frac{7}{1}$ mais 7...

Et, pour le coefficient `p`, on a évité l'affichage `+5` lorsque que `p` était négatif. Cela fait beaucoup de cas à considérer ce qui rend le script un peu plus complexe.

Mais nous n'avons pas envisager tous les cas particuliers...

En reprenant l'exemple précédent on obtient :

```
PYTHON SHELL
>>> reduite3(3,4,14,12)
'y = 8/11 * x + 20/11'
```

```
ÉDITEUR : DROITE
LIGNE DU SCRIPT 0033
from ARITHM import *
def reduite3(xa,ya,xb,yb):
    if oy(xa,xb):
        return "x = "+str(xa)
    else:
        d=pgcd(abs(yb-ya),abs(xb-xa))
        mnum=(yb-ya)//d
        mdenom=(xb-xa)//d
        pnum=yb*mdenom-mnum*xb
        pdenom=mdenom
        d2=pgcd(abs(pnum),abs(pdenom))
        pnum=pnum//d2
        pdenom=pdenom//d2
        if mdenom!=1:
            maff=str(mnum)+"/"+str(mdenom)
        else:
            maff=str(mnum)

        if pdenom!=1:
            paff=str(pnum)+"/"+str(pdenom)
        else:
            paff=str(pnum)
        if pnum/pdenom>0:
            paff="+ "+paff
        return "y = "+maff+" * x "+paff
```