

Des formules dans un repère

Compétences visées

- **chercher**, expérimenter – en particulier à l'aide d'outils logiciels ;
- **représenter**, choisir un cadre (numérique, algébrique, géométrique...), changer de registre ;
- **raisonner**, démontrer, trouver des résultats partiels et les mettre en perspective ;
- **calculer**, appliquer des techniques et mettre en œuvre des algorithmes.

Le programme de la classe de 2^{nde} GT propose explicitement une approche algorithmique pour déterminer les coordonnées du milieu d'un segment, la distance entre deux points, la caractérisation de la colinéarité de deux vecteurs et l'alignement de trois points.

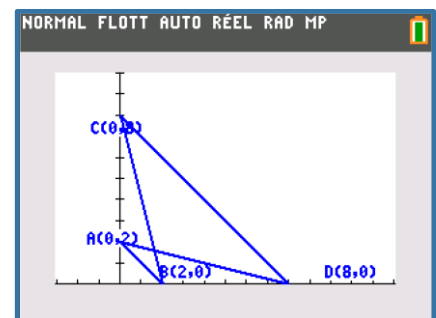
Situation déclenchante

Géométrie dans un repère

Sur la figure ci-contre : A (0,2), B (2,0), C (0,8) et D (8,0) sont quatre points du plan dont on donne les coordonnées.

On note E le milieu de [AB], F le milieu de [DC] et G l'intersection des segments [AD] et [BC].

Les points E, F et G sont-ils alignés ?



Problématique

Soient A (x_A, y_A), B (x_B, y_B), C (x_C, y_C) et D (x_D, y_D) ; créer des programmes permettant de :

1. Déterminer les coordonnées de I, milieu de [AB] ;
2. Déterminer la longueur du segment [AB] ;
3. Déterminer les coordonnées du vecteur \overrightarrow{AB} ;
4. Déterminer si les vecteur \overrightarrow{AB} et \overrightarrow{CD} sont colinéaires ;
5. Déterminer si les points A, B et C sont alignés.

Fiche méthode

Proposition de résolution

On va créer plusieurs fonctions correspondantes à chaque situation.

- Écrire la fonction **milieu** dont les quatre paramètres sont les coordonnées des extrémités du segment et renvoie les coordonnées du milieu de ce segment. Le résultat est rendu sous forme d'un tuple de deux valeurs. Son affichage est très proche de celui de coordonnées de points, ce qui en facilite la lecture.
- Écrire la fonction **vec** dont les quatre paramètres sont les coordonnées des points A et B et qui renvoie les coordonnées du vecteur \overline{AB} .
- Écrire la fonction **long** dont les quatre paramètres sont les coordonnées des extrémités du segment dont on veut connaître la longueur et qui renvoie une valeur approchée de la longueur de ce segment (la fonction **long2** en est une variante et donne évidemment le même résultat).
- Écrire la fonction **col** qui a pour paramètres les coordonnées des deux vecteurs dont on teste la colinéarité. Elle retourne un booléen : True si les deux vecteurs sont colinéaires, False dans le cas contraire.
- Écrire la fonction **ali** (on proposera une variante en terme de syntaxe nommée **ali2**) qui a pour paramètres les coordonnées des trois points dont on teste l'alignement. Elle retourne un booléen : True si les trois points sont alignés, False dans le cas contraire.

```
PYTHON SHELL
>>> milieu(-1,2,5,9)
(2.0, 5.5)
>>> vec(2,6,9,-11)
(7, -17)
>>> long(1,6,3,-2)
8.246211251235321
>>> long(0,0,3,4)
5.0
>>> long2(0,0,3,4)
5.0
>>> |
Fns... a A # Outils Éditer Script
```

```
PYTHON SHELL
>>> col(1,1,3,3)
True
>>> col(1,1,3,4)
False
>>> ali(0,0,1,1,5,5)
True
>>> ali(0,0,1,1,5,6)
False
>>> |
Fns... a A # Outils Éditer Script
```

Remarque

Importation en préambule du code de la bibliothèque « math » par « **from math import *** » pour pouvoir utiliser la fonction **sqrt** (racine carrée)

```
ÉDITEUR : LONGCRBE
LIGNE DU SCRIPT 0002
from math import *
Fns... a A # Outils Exéc Script
```

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus



Fiche méthode

Etapes de résolution

- La fonction **milieu** retourne un tuple de deux valeurs qui correspondent aux coordonnées du milieu du segment dont on a saisi en paramètres les coordonnées des extrémités.
- La fonction **vec** est similaire.
- La fonction **long** donne la longueur d'un segment ; les paramètres sont les coordonnées des extrémités du segment considéré.
- On en propose une seconde version, **long2** qui a une syntaxe plus concise.
- La fonction **col** ainsi rédigée est efficace car elle retourne un booléen (True dans le cas où les vecteurs considérés sont colinéaires, False dans le cas contraire). « $xu*yv-yu*xv==0$ » est en effet un booléen.
== signifie que l'on compare les deux quantités $xu*yv-yu*xv$ et 0.
- On utilise la propriété suivante : A, B, C sont alignés si et seulement si \vec{AB} et \vec{AC} sont colinéaires. On propose deux syntaxes pour tester l'alignement de trois points.
 - La première, **ali** utilise la fonction **col** et illustre bien la démarche consistant à étudier la colinéarité de deux vecteurs construits à partir de ces trois points.
 - La seconde syntaxe, **ali2** reprend le principe utilisé pour la fonction **col** en retournant directement un booléen.

```
ÉDITEUR : GEOMETRI
LIGNE DU SCRIPT 0011
from math import *

def milieu(xA,yA,xB,yB):
    return (xA+xB)/2,(yA+yB)/2

def vec(xA,yA,xB,yB):
    return xB-xA,yB-yA

Fns... a A # Outils Exéc Script
```

```
ÉDITEUR : GEOMETRI
LIGNE DU SCRIPT 0017

def long(xA,yA,xB,yB):
    x=xB-xA
    y=yB-yA
    return sqrt(x**2+y**2)

def long2(xA,yA,xB,yB):
    return sqrt((xB-xA)**2+(yB-yA)**2)

Fns... a A # Outils Exéc Script
```

```
ÉDITEUR : GEOM
LIGNE DU SCRIPT 0023

def col(xu,yu,xv,yv):
    return xu*yv-yu*xv==0

Fns... a A # Outils Exéc Script
```

```
ÉDITEUR : GEOMETRI
LIGNE DU SCRIPT 0029

def ali(xA,yA,xB,yB,xC,yC):
    x1=xB-xA
    y1=yB-yA
    x2=xC-xA
    y2=yC-yA
    return col(x1,y1,x2,y2)

def ali2(xA,yA,xB,yB,xC,yC):
    return (xB-xA)*(yC-yA)-(yB-yA)*(xC-xA)==0

Fns... a A # Outils Exéc Script
```

Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus



Fiche méthode

Retour à la situation déclenchante

On peut utiliser la fonction **milieu** pour déterminer les coordonnées des points E et F.

Pour déterminer les coordonnées de G, on peut chercher l'équation des droites (AD) et (BC) ; cela donne :

- (AD) : $y = -0,25x + 2$
- (BC) : $y = -4x + 8$

On peut utiliser le solveur de systèmes d'équations de la calculatrice pour déterminer les coordonnées de leur point d'intersection.

Pour cela, taper sur **rsol** puis valider la touche 2 : **PlySmlt2** ; on choisit à nouveau la touche 2 : **SOLVEUR SYST D'ÉQUATIONS**

On choisit de résoudre un système de deux équations à deux inconnues en validant les bons paramètres et on saisit les coefficients comme ci-contre.

Une fois **RÉSOL** validé, la solution est proposée (résultat donné ici sous forme fractionnaire).

On trouve ainsi : G (1.6,1.6)

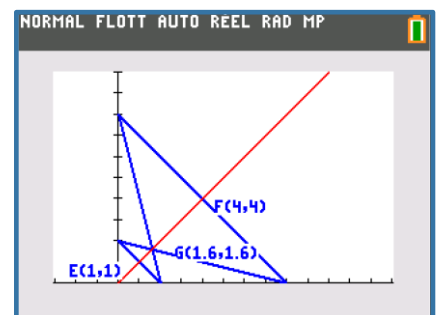
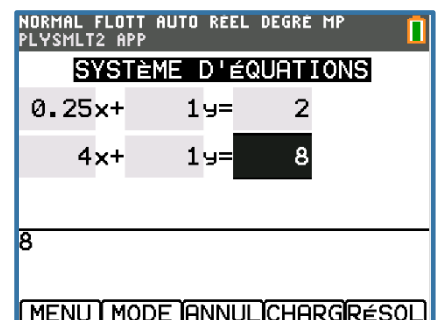
On peut à présent tester l'alignement des points E, G et F par **ali(1,1,1.6,1.6,4,4)**.

On peut également évaluer les longueurs des segments [AF] et [BF] par **long(0,2,1.6,1.6)** et **long(2,0,1.6,1.6)**.

Ces programmes une fois exécutés permettent de lancer deux pistes prouvant l'alignement de ces points :

- En montrant que les vecteurs \vec{EG} et \vec{EF} sont colinéaires.
- En montrant que $AG=GB$ et $AF=FB$, prouvant ainsi que les points E, F et G appartiennent tous trois à la médiatrice de [AB].

```
PYTHON SHELL
>>> milieu(0,2,2,0)
(1.0, 1.0)
>>> ali(1,1,1.6,1.6,4,4)
True
>>> long(0,2,1.6,1.6)
1.649242250247065
>>> long(2,0,1.6,1.6)
1.649242250247065
>>> |
```



Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus

