

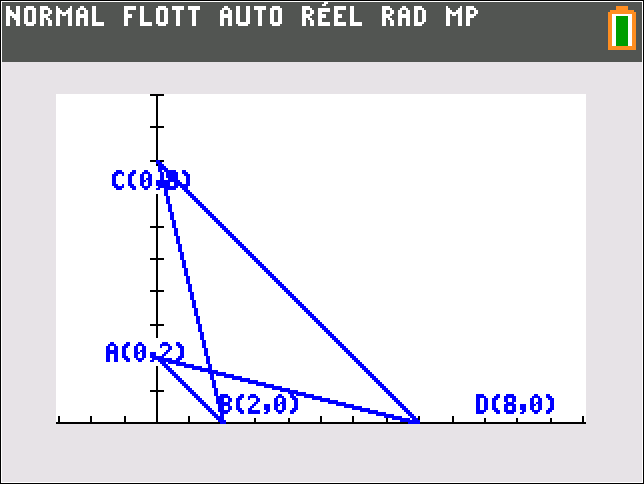
Ce document est mis à disposition sous licence Creative Commons <http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>

**© Texas Instruments 2019 / Photocopie autorisée**

Compétences visées

* **chercher**, expérimenter – en particulier à l’aide d’outils logiciels ;
* **modéliser**, faire une simulation, valider ou invalider un modèle ;
* **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 2nde 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 (xA , yA), B (xB , yB), C (xC , yC) et D (xD , yD) ; créer des programmes permettant de :

1. déterminer les coordonnées de I, milieu de [AB] ?
2. déterminer la longueur du segment [AB] ;
3. déterminer les coordonnés du vecteur  ;
4. déterminer si les vecteur  et  sont colinéaires ;
5. déterminer si les points A, B et C sont alignés.

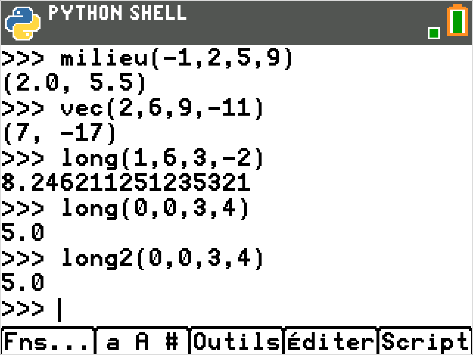


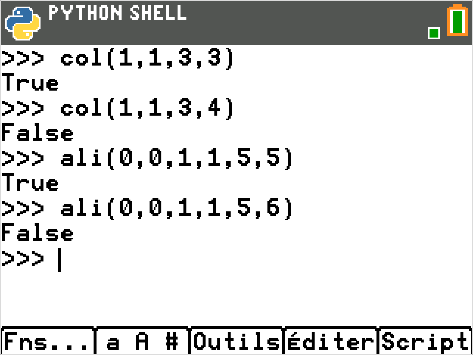
Pour profiter de tutoriels vidéos, Flasher le QRCode ou cliquer dessus !

Ce document est mis à disposition sous licence Creative Commons <http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>

**© Texas Instruments 2019 / Photocopie autorisée**

# Proposition de résolution

**On va créer plusieurs fonctions correspondantes à chaque situation.**

* On va tout d’abord créer la fonction **milieu**() rendant les coordonnées du milieu d’un segment. Les paramètres d’entrée sont les coordonnées des extrémités du 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 facilité la lecture.
* Il en est de même pour la fonction **vec**() qui retourne les coordonnées d’un vecteur  ; les paramètres d’entrée sont les coordonnées des points A et B.
* La fonction **long**() donne une valeur approchée de la longueur d’un segment (la fonction long2() en est une variante et donne évidemment le même résultat).
* La fonction **col**() 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.
* Pour l’alignements de trois points, on utilisera deux fonctions précédentes, à savoir **vec**() et **col**(). Les paramètres d’entrées sont les coordonnées des trois points considérés. On propose deux syntaxes pour le même résultat par les fonctions **ali**() et **ali2**().

# Remarque



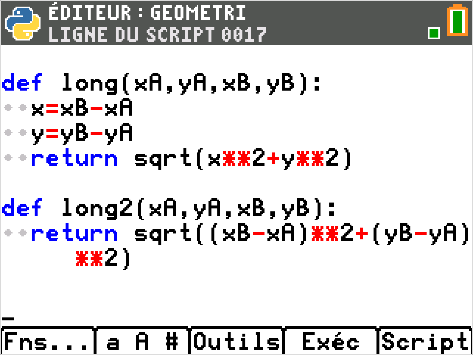
importation en préambule du code de la bibliothèque « math » par « **from math import \*** » pour pouvoir utiliser la fonction **sqrt()** (racine carrée)

**[](https://www.youtube.com/playlist?list=PL4V-Xo0EMx4iEoEJb1zsnft-xngiwTeNn)**

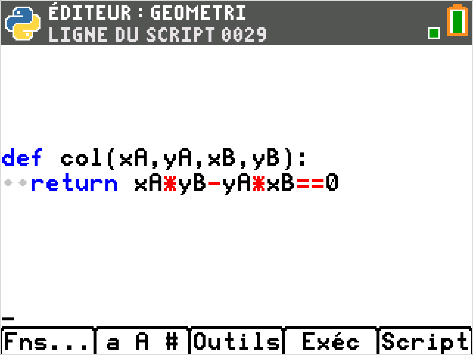
# C:\Users\Marion\AppData\Local\Temp\Texas Instruments\TI Connect CE\Capturer3-1556604866467.png

# 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.

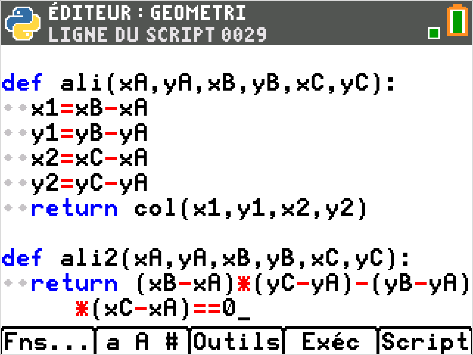


* La fonction **long**() donne la longueur d’un segment ; les paramètres sont les coordonnées des extrémités du segment considéré.
* On en propose un seconde version, **long2(**) qui a une syntaxe plus efficace.



* 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). « xA\*yB-yA\*xB==0 » est en effet un booléen.

== signifie que l’on compare les deux quantités xA\*yB-yA\*xB et 0.

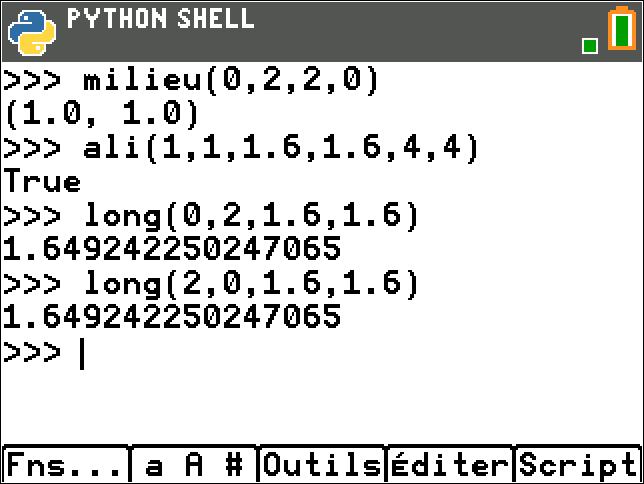


* On propose enfin deux syntaxes pour une fonction testant l’alignement de trois points : la première, **ali**() utilise la fonction **col**() et illustre bien la démarche consistant à étudier la colinéarité de deux vecteurs construits à partir de ces trois points.

La seconde syntaxe, **ali2**() reprend le principe utilisé pour la fonction **col**() en retournant directement un booléen.

**[](https://www.youtube.com/playlist?list=PL4V-Xo0EMx4iEoEJb1zsnft-xngiwTeNn)**

# Retour à la situation déclenchante

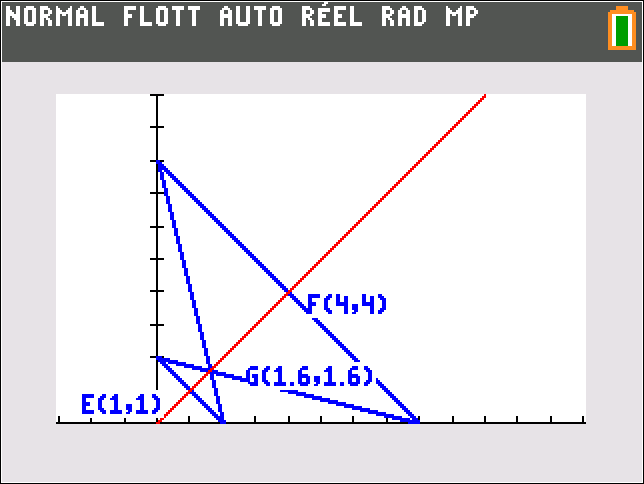
On peut utiliser la fonction **milieu**() pour déterminer les coordonnées des points E et F.

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

* (AD) :
* (BC) :

On peut utiliser le solveur de systèmes d’équations de la calculatrice pour déterminer les coordonnées de leur point d’intersection ; on trouve : 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).



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

Ce document est mis à disposition sous licence Creative Commons <http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>

**© Texas Instruments 2019 / Photocopie autorisée**

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

* en montrant que les vecteurs et sont colinéaires.
* **[](https://www.youtube.com/playlist?list=PL4V-Xo0EMx4iEoEJb1zsnft-xngiwTeNn)**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].