

Géométrie repérée et Algorithmique

LOIC CHAPELLIER

Ip2i, Jaunay-Marigny

Voici un programme construit avec le logiciel Scratch:



On se place dans un repère orthonormé (O, I, J).

On pourra construire une figure que l'on complétera au fur et à mesure.

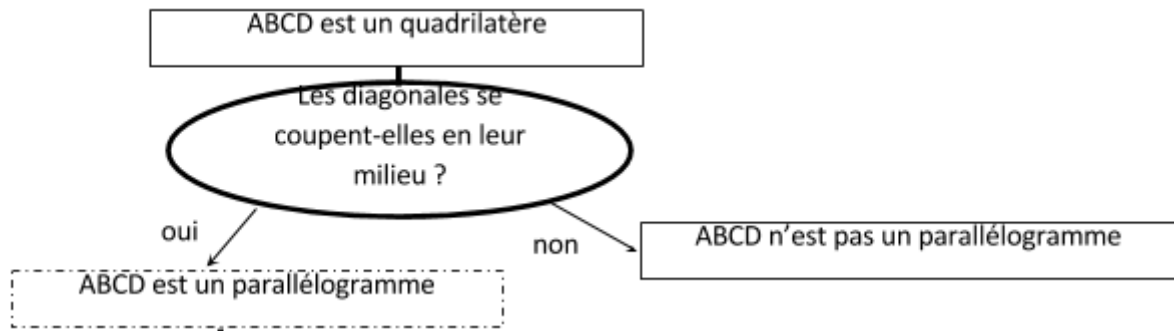
1. Que va-t-il afficher avec les points $A(-1;-1)$ et $C(5;1)$? Placer les points dans le repère construit.
2. A quoi sert ce programme ?
3. On considère les points $E(4,2)$ et $D(2;0)$.
 - a) Placer les points dans le repère.
 - b) Déterminer les coordonnées du point F tel que D est le milieu de [EF].
 - c) Comment vérifier le résultat à l'aide du programme ?
4. Voici le même algorithme programmé avec Python

```
def milieu(xA,yA,xB,yB):  
    return((xA+xB)/2,(yA+yB)/2)  
  
a=float(input('Quelle est l abscisse du point A'))  
b=float(input('Quelle est l ordonnée du point A'))  
  
c=float(input('Quelle est l abscisse du point B'))  
d=float(input('Quelle est l ordonnée du point B'))  
print(milieu(a,b,c,d))
```

- a. Quel est l'avantage de la programmation Python ?
- b. Que se passe-t-il quand on tape la commande « milieu(a,b,c,d) » dans le programme.
- c. Utilisez cet algorithme afin d'en construire un qui permet de vérifier si un quadrilatère est un parallélogramme.
- d. On considère les points $B(1;3)$ et $G(3;-3)$. Placer les points dans le repère dans le repère (O, I, J)

Utilisez l'algorithme pour vérifier que ABCG est un parallélogramme.

5. Cet algorithme peut être schématisé ainsi :



a. Cet algorithme peut être vu comme le point de départ d'un algorithme plus complet qui permettrait de vérifier qu'un quadrilatère est quelconque, parallélogramme ou rectangle.

Recopier le schéma et le compléter afin de pouvoir vérifier si le quadrilatère est rectangle.

b. Voici le programme suivant réalisé sous Python :

```
#Calcul les coordonnées du milieu d'un segment
from math import *

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

a=float(input('Quelle est l abscisse du point A'))
b=float(input('Quelle est l ordonnée du point A'))

c=float(input('Quelle est l abscisse du point B'))
d=float(input('Quelle est l ordonnée du point B'))
print(longueur(a,b,c,d))
```

A l'aide de cet algorithme, construire un algorithme qui permet de vérifier si un quadrilatère est un rectangle.

c. Utilisez cet algorithme pour vérifier si le quadrilatère ABCG est un rectangle.

d. On considère les points $M(5;-4)$; $N(7;1)$; $K(2;3)$ et $L(0;-2)$.

Déterminer la nature du quadrilatère MNKL et vérifier le avec l'algorithme construit.