

Ma Vache est partie

1 le background

Un paysan facétieux mais néanmoins amateur de mathématiques va chaque nuit modifier la clôture du champ d'un autre paysan. Étant un peu fantasque, il donne à cette clôture une forme parfois simple et parfois très tarabiscotée. Il se permet de changer le nombre de poteaux (avec un minimum de 3) ainsi que leur emplacement. La seule contrainte qu'il respecte, c'est que la clôture est fermée sur elle-même et que les fils ne se croisent jamais.

Une fois les modifications faites, il conduit l'unique vache de l'autre paysan au centre exact de la clôture. Généralement, ce dernier laisse faire et se contente de bougonner en attendant le prochain changement. Malheureusement, dans certain cas, sa vache se retrouve à l'extérieur de la clôture et elle peut s'enfuir. Bien entendu, il ne veut pas surveiller en permanence sa vache, au cas où elle serait à l'extérieur de la clôture, sans compter le temps perdu si, en fait, elle est à l'intérieur. A la troisième évasion, il décide donc d'aller voir un de ses amis qui se dit informaticien, pour trouver une solution.

Ce dernier lui dit : « je vais te faire un programme pour ton smart-phone. Tu n'auras qu'à suivre la clôture, et chaque fois que tu te trouveras devant un poteau, tu cliqueras. Comme ça, tu obtiendras les coordonnées des poteaux. Le programme calculera ensuite le centre physique du champ délimité par la clôture et te dira s'il se trouve à l'intérieur ou non. Tu sauras ainsi immédiatement si tu dois déplacer ta vache ou la laisser paître ».

2 l'énoncé

L'ami en question est peut être bon en développement pour smart-phone mais c'est une bille en algorithmique et mathématiques. Il vous demande donc de l'aide pour la partie calcul.

Votre programme doit lire sur l'entrée standard :

1. un entier N représentant le nombre de formes de clôture à traiter.
2. N lignes donnant la forme d'une des clôtures.

Chaque ligne de clôture commence par un entier donnant le nombre de poteaux, puis par une liste de couples de nombres à virgule, représentant les coordonnées en (x, y) de chaque poteau. Elles sont données « dans l'ordre », comme si l'on suivait le fil de la clôture, et cela, dans le **sens inverse** des aiguilles d'une montre. Toutes les valeurs sont

séparées par des espaces.

Pour chaque clôture, votre programme doit déterminer le barycentre (ou centre de masse/gravité) du champ délimité par celle-ci, puis vérifier que celui-ci se trouve à l'intérieur de la clôture.

Votre programme doit écrire sur la sortie standard N lignes. La i^{eme} ligne concerne la i^{eme} clôture traitée en entrée. Chaque ligne commence par le nombre de poteaux de la clôture, suivie de la chaîne `yes` ou `no`, selon le fait que le barycentre de la clôture se trouve ou non à l'intérieur de celle-ci.

Exemple d'entrée/sortie :

entrée	sortie
3	
4 -16.6 -20.1 -12.6 -18.6 -11.6 -16.6 -15.1 -15.1	4 yes
4 -1.1 -1.5 2.1 3.012 5.6 -1.21 1.97 4.07	4 no
5 0.1 -0.09 2 1.06 1.996 3.05 -2.1 2.95 0.08 2.1	5 yes

TABLE 1 – Exemple d'entrée et la sortie attendue

3 les ressources

Pour vous aider dans la réalisation du programme, vous trouverez sur

<http://cours-info.iut-bm.univ-fcomte.fr>

un article dans la section `hackaton` de l'année courante, portant le même titre que l'exercice. Il contient un lien permettant de télécharger un canevas de code, ainsi que le fichier d'entrée donné ci-dessus.

Bien entendu, vous êtes libres d'utiliser ou non ce canevas, mais c'est un gain de temps que de s'en servir comme base.