

- TP Polygones
 - Exercice 1 : Intersection de segments
 - 1. Produit vectoriel et orientation
 - Objectif
 - Conditions particulières
 - Solution technique
 - 1. Méthode coupe()
 - 2. Méthode intersection()

TP Polygones

Exercice 1 : Intersection de segments

1. Produit vectoriel et orientation

Pour deux vecteurs $v1(x1,y1)$ et $v2(x2,y2)$, le produit vectoriel 2D est défini par :

$$v1 \times v2 = x1*y2 - y1*x2$$

Le signe du produit vectoriel indique l'orientation relative des deux vecteurs :

- Si le produit est positif, $v2$ est à gauche de $v1$
- Si le produit est négatif, $v2$ est à droite de $v1$
- Si le produit est nul, les vecteurs sont colinéaires

Objectif

Implémenter deux méthodes dans la classe `Segment.java` :

- `coupe(Segment seg)` : teste si deux segments s'intersectent
- `intersection(Segment seg)` : calcule le point d'intersection s'il existe

Conditions particulières

- Les extrémités du segment paramètre sont incluses $[0,1]$ (intervalle fermé)
- Les extrémités du segment this sont exclues $]0,1[$ (intervalle ouvert)
- Cas spéciaux à gérer :
 - Segments qui se frôlent
 - Segments bout à bout
 - Segments alignés

Solution technique

1. Méthode coupe()

La détection d'intersection utilise les produits vectoriels pour déterminer si :

1. Les points du segment paramètre sont de part et d'autre du segment this
2. Les points du segment this sont strictement de part et d'autre du segment paramètre

Algorithme :

1. Calculer les produits vectoriels (pv1, pv2) des points du segment paramètre
2. Vérifier qu'ils ne sont pas du même côté (signes opposés)
3. Calculer les produits vectoriels pour les points du segment this
4. Vérifier qu'ils sont strictement de part et d'autre (produit négatif)

2. Méthode intersection()

Pour calculer le point d'intersection :

1. Vérifier d'abord s'il y a intersection avec coupe()
2. Résoudre le système d'équations linéaires des deux droites
3. Utiliser le déterminant pour trouver le point d'intersection