

TP1- Diviser pour régner

Question 1. Une première approche

Un rectangle de surface maximale respectant les contraintes a nécessairement deux sommets de la forme $(x_i, 0), (x_j, 0)$ avec $0 \leq i < j \leq n-1$ car une des contraintes est que le rectangle doit avoir comme base l'axe des x. C'est pour cela qu'il a forcément deux sommets qui ont pour ordonnée 0 (car ils sont sur l'axe des x)

Pour exprimer cela, il suffit de multiplier $(x_j - x_i)$ par la hauteur du point dont l'abscisse est supérieur à x_i et inférieur à x_j et qui a l'ordonnée la plus faible.

La fonction est codée en java, se trouve dans la classe *Rectangle* et s'appelle *getBiggestAreaRectangle()*. Une version améliorée de l'algorithme est *getBiggestAreaRectangle2()*.

Nous ne pensons pas que ce soit compatible avec notre algorithme.

Question 2. Diviser pour régner

La fonction est codée en java, se trouve dans la classe *Rectangle* et s'appelle *getBiggestAreaRectangleRec()*.

L'algorithme est plus efficace que les précédents algorithmes.