|  |
| --- |
| Projet de complexité |
| Pour le 18 Avril 2017 |
| Lieffroy Mervine, Lemaitre Martin, Calley Nicolas, Keller Rowane |

# Question 1

Pour notre tri, nous avons choisi d’implanter l’algorithme du tri par tas. Nous avons vu en cours que la complexité de cet algorithme est n log n.

La complexité au pire cas  de ces deux algorithmes est n² car

* Pour ToutesLesPaires, on compare à chaque fois un cercle avec tous les autres cercles. On a une double boucle (« pour » et « tant que ») qui dépend de n, la complexité est donc n² au pire cas (celui où on vérifie chaque disque)
* Pour Balayage, on observe également une double boucle qui dépend de n. La complexité au pire cas est donc, également, n² dans le cas où chaque disque est vérifié.

# Question 2

* 1. K= O(n²), car au pire cas on compare tous les cercles entre eux (n²/2). Par conséquent, k peut être majoré par n².
  2. Le calcul de la liste triée dépend de la coordonnée x de chaque disque. Or, k représente le nombre de croisements entre nos disques et ce facteur n’est pas pris en compte du tout cet algorithme. Par conséquent, la complexité du calcul de la liste triée ne dépend pas de k.
  3. On a la complexité du tri de notre liste L qui est de n log n. D’après l’énoncé, la complexité de l’algorithme Balayage est de

=

Sachant que la somme des ki est égale à k, on obtient la formule suivante pour la complexité de l’algorithme Balayage : n (log(n)) + k + (n-1)

* 1. La complexité obtenue en 1) ne dépend pas de k. Par conséquent, elle ne varie pas si k est grand ou petit.

La complexité obtenue en 2)d dépend de k. Si k est petit, il y aura peu de « croisements » de cercles et k sera inférieur à n log n. Au contraire, si k est grand beaucoup de croisements auront lieu et k sera supérieur à n log n, il pourra même aller jusqu’à n².

# Question 3

# Question 4

# Question 5

# Question 6