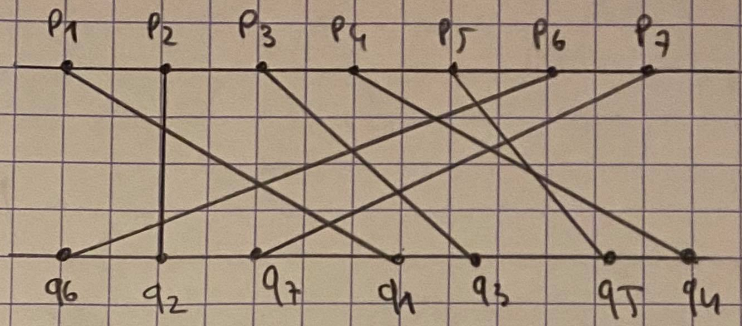
Exercice 10



(point noté dans l'ordre croissant)

(point q noté en fin de leur liaison au pt p)

	1	2	3	4	5	6	7
A	6	2	7	1	3	5	4

le tableau A donne l'ordre inverse des indices des pt q

Q. 1 Il y a une inversion si $i < j$ et $A[i] > A[j]$

i	j	inversion	
1	2	O	$1 < 2$ et $A[1] = 6 > A[2] = 2$
1	3	N	$1 < 3$ et $A[1] = 6 < A[3] = 7$
1	4	O	$1 < 4$ et $A[1] = 6 < A[4] = 1$
1	5	O	$1 < 5$ et $A[1] = 6 < A[5] = 3$
1	6	O	$1 < 6$ et $A[1] = 6 < A[6] = 5$
1	7	O	$1 < 7$ et $A[1] = 6 < A[7] = 4$
2	3	N	
2	4	O	
2	5	N	
2	6	N	
2	7	N	
3	4	O	
3	5	O	
3	6	O	
3	7	O	
4	5	N	
4	6	N	
4	7	N	

5	6	N
5	7	N
6	7	0

Q.2

Il y a 11 intersections de segments sur la figure et on a trouvé 11 inversion donc il y a autant d'intersections que d'inversion.

- Vrai pour toutes les instances car si $p_i < p_j$ alors une intersection correspond à $q_i > q_j$

-> si intersectⓞ $[p_i, q_i]$ et $[p_j, q_j]$

alors q_i "à droite" de q_j

-> Inversⓞ dans le tableau et réciproquement si $p_i < p_j$ et $q_i > q_j$ alors il doit y avoir une intersection entre $[p_i, q_i]$ et $[p_j, q_j]$

Q.3

Principe d'un algorithme naïf :

-> on prend chq couple (i, j)

-> on teste s'il y a une inversion

or il y a $\frac{n(n-1)}{2}$ paires donc la complexité est en $\Theta(n^2)$

Trier-et-compter (A) :

si $|A| = 1$ renvoyer $(0, A)$

sinon

Diviser A_1 en A_2

$(m_1, A'_1) \leftarrow$ Trier-et-compter (A_1)

$(m_2, A'_2) \leftarrow$ Trier-et-compter (A_2)

$(m_3, A) \leftarrow$ Fusionner et compter (A'_1, A'_2)

renvoyer $(m_1 + m_2 + m_3, A')$

Observation : L'algorythme de fusion est en $O(m)$ car pour faire la fusion, i parcourt un tableau de longueur $|A1'|$, j parcourt un tableau de longueur $|A2'|$, la la complexité est donc en $O(|A1'| + |A2'|) = O(m)$.

i

A_1' : | 2 | 6 | 7 |

↳ sous-tableau gauche

A'_2 | 1 | 3 | 4 | 5 |

j

↳ sous-tableau droit

| 2 | 6 | 7 | | 1 | 3 | 4 | 5 | + 3 inversions

| 1 | | 2 | 6 | 7 | | 3 | 4 | 5 | + 0 inversion

| 1 | 2 | | 6 | 7 | | 3 | 4 | 5 | + 2 inversions

Q.4

$m_3 \leftarrow m_3 + (|A_1'| - i + 1)$

(nombre de sauts jusqu'à la case avant i dépend de la longueur du tableau et de l'indice i)

Q.5

Fusionner et compter à la même complexité qu'une fusion de deux listes triées.
Donc la complexité est en $O(m)$

Q.6

$$T(m) = 2T\left(\frac{m}{2}\right) + O(m)$$

On utilise le théorème maître pour en déduire la complexité de l'algorithme

$$\begin{cases} a = 2 \\ b = 2 \\ d = 1 \end{cases} \quad \begin{array}{l} \Rightarrow \quad a = b^d \\ \Rightarrow \quad \Theta(m \log_2 m) \end{array}$$