Question III.8) - *version longue complexité*

Pour une liste `l` de taille $n \in \mathbb{N}^*$, l'appel `tri_fusion l` effectue :

- un appel à la fonction `separe` en $\mathcal{O}(n)$
- deux appels récursifs sur des listes de tailles $\lfloor \frac{n}{2} \rfloor$ et $\lceil \frac{n}{2} \rceil$
- un appel à `fusionne` en $\mathcal{O}(n)$.

En notant, pour tout $n \in \mathbb{N}$, $C(n)$ la complexité dans le pire cas d'un appel `tri_fusion l` où `|l|` $= n$,

$$C(n) = C\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + C\left(\left\lceil \frac{n}{2} \right\rceil\right) + \mathcal{O}(n)$$

Pour tout $p \in \mathbb{N}$, on a donc :

$$\begin{aligned}
&C(2^{p+1}) = 2C(2^p) + \mathcal{O}(2^{p+1}) \\
&\text{donc} \qquad \frac{C(2^{p+1})}{2^{p+1}} = \frac{C(2^p)}{2^p} + \mathcal{O}(1) \\
&\text{d'où} \qquad \frac{C(2^p)}{2^p} = \mathcal{O}(p) \\
&\text{puis} \qquad C(2^p) = \mathcal{O}(p2^p) \\
&\\
&\text{De plus, si } n = 2^p, p = \log_2 n
\end{aligned}$$

En supposant de plus que la complexité est croissante, on a pour n quelconque

$$C(n) = \mathcal{O}(n \log(n))$$

.