Définition
Complexité spatiale : nombre d'emplacements mémoire occupés lors de l'exécution d'un programme.
Complexité temporelle : nombre d'opérations effectuées lors de l'exécution d'un programme (on compte en général le nombre d'occurrences de l'opération la plus longue à exécuter).

Définition

- Complexité dans le meilleur des cas : majoration du temps d'exécution pour toutes les entrées possibles d'une même taille.
- Complexité dans le pire des cas : minoration du temps d'exécution pour toutes les entrées possibles d'une même taille.
- Complexité en moyenne : temps moyen d'exécution pour toutes les entrées possibles d'une même taille.

Définition **Notation de Landau**
Soient C et f deux fonctions $C, f : \mathbb{N} \to \mathbb{R}_*^+$. On note $C(n) = \mathcal{O}(f(n))$ lorsqu'il existe $c \in \mathbb{R}^+$ et $n_0 \in \mathbb{N}$ tels que pour tout $n \geq n_0$, $C(n) \leq c \cdot f(n)$.

Exemple On recense différents types de complexité :
- $\mathcal{O}(1)$: complexité constante *(recherche du maximum dans une liste triée)* ;
- $\mathcal{O}(n)$: complexité linéaire *(recherche d'un nombre dans une liste)* ;
- $\mathcal{O}(n^2)$: complexité quadratique *(recherche d'un nombre dans un tableau à deux dimensions)* ;
- $\mathcal{O}(2^n)$: complexité exponentielle *(recherche du plus court chemin passant par n points)* ;
- $\mathcal{O}(\log n)$: complexité logarithmique *(recherche dichotomique dans une liste triée)* ;
- $\mathcal{O}(n \log n)$: complexité semi linéaire.

Méthode
Dans le cas général il est probable que l'on vous demande d'intuiter la complexité temporelle d'un algorithme.
Si on vous demande de la démontrer, il faudra raisonner par récurrence (vérifier que la complexité est vraie au rang initial, admettre qu'elle est vraie jusqu'au rang n puis vérifier qu'elle est vraie au rang $n+1$).

Méthode **Décomptage du temps de temps d'exécution :**
- Le coût de deux instructions consécutives est égal à la somme des coût des deux instructions.
- Le coût d'une instruction conditionnelle, dans le pire des cas, est égale au maximum des coûts de chacun des blocs d'instructions.
- Le coût d'une instruction itérative est proportionnel au nombre d'itérations du bloc d'instructions.

Exemple Complexité de l'algorithme de recherche dichotomique d'un nombre dans une liste triée. Pour cela appelons $C(n)$ le nombre de divisions euclidienne par 2 réalisées.
Soit n la taille de la liste. On peut donc la diviser k fois par 2, jusqu'à ce que n soit strictement inférieur à 2 (quand la zone de recherche sera de taille 1, on ne pourra plus diviser le problème).
On cherche donc n tel que $n/2^k < 2 \Leftrightarrow n < 2^{k+1} \Rightarrow \ln n < (k+1)\ln 2 \Leftrightarrow k > \dfrac{\ln n}{\ln 2} - 1$.
Le nombre de divisions est donc proportionnel à $\ln n$. On a donc $C(n) = \mathcal{O}(n)$.