

Fiche 05 :

Notion de complexité

Licence 3 informatique

Exercice 1 : Calcul de complexité d'algorithmes itératifs

Soient les deux algorithmes suivants :

Algorithme machin(T : tableau d'entier, n : entier)

begin

for i from 0 to $n - 1$ **do**

$x = T[i]$

$T[i] = T[n - 1 - i]$

$T[n - 1 - i] = x$

end for

end

Algorithme truc(M : matrice d'entier, n : entier)

begin

for i from 1 to $n - 1$ **do**

for j from 0 to $i - 1$ **do**

$x = M[i][j]$

$M[i][j] = M[j][i]$

$M[j][i] = x$

end for

end for

end

Questions :

1.a. Quelle est la fonction calculée par chacun des algorithmes ?

1.b. Pour chacun des algorithmes, calculer la complexité de la manière la plus précise possible, puis une borne supérieure à une constante près, et enfin donner la classe de complexité.

Exercice 2 : Comparaison de complexité d'algorithmes

Questions :

2.a. Calculer la complexité temporelle de l'algorithme de tri par insertion (en notation \mathcal{O})

2.b. Calculer la complexité temporelle de l'algorithme tri rapide (quick sort).

Exercice 3 : Hash table

Questions :

3.a. Qu'est-ce qu'une table de hachage?

3.b. Comparer les complexités spatiale et temporelle de la recherche d'un nombre utilisant les structures de données : tableau, liste et table de hachage.

Exercice 4 : Base de données

Questions :

4.a. Quelle est la complexité temporelle pour retrouver une donnée dans une base de données relationnelle?