
TP 1 – Vitesse d'un algorithme

L'objectif de cette activité est de comparer les vitesses d'exécution d'algorithmes différents réalisant une même tâche.

1 Recherche dans un tableau trié

Soit une liste L de nombres triés dans l'ordre croissant et un nombre x .

On veut déterminer si le nombre x est présent ou non dans la liste L .

Pour cela on réalise une fonction qui prend en variables la liste L et le nombre x , et qui renvoie une variable booléenne : VRAI *présent* ou FAUX *non présent*.

1. Recherche séquentielle

Écrire et tester un algorithme qui réalise cette tâche en utilisant une boucle séquentielle.

2. Recherche dichotomique

Écrire et tester un algorithme qui réalise les opérations ci-dessous.

- a. définir les variables `début` et `fin` et les initialiser au rang du début et de la fin de la liste L
- b. lancer une boucle conditionnelle qui se termine quand `début` égale `fin`
- c. définir une variable `rang` qui prend la valeur de la moyenne entre `début` et `fin` arrondi à l'entier inférieur.
- d. définir la variable `valeur` qui prend la valeur de l'élément de la liste L de rang `rang`.
- e. Si la valeur x recherchée est égale à la variable `valeur`, retourner `True`
- f. Si la valeur x recherchée est inférieure à la variable `valeur`, adapter l'intervalle `[début, fin]` de sorte que la boucle conditionnelle s'applique à la partie de la liste qui peut contenir x .
- g. Si au final la valeur x n'est pas dans la liste L , retourner `False`.

3. Comparaison du nombre d'opérations

Comparer le nombre d'opérations réalisées dans chacune des deux recherches en fonction du nombre n d'éléments contenus dans la liste.

2 Exponentiation

Soit un réel a et un entier naturel n .

On cherche à réaliser une fonction pour calculer a^n . Il s'agit de programmer la fonction `a**n`.

1. Calcul à l'aide d'une boucle séquentielle

Écrire et tester un algorithme qui réalise cette tâche en utilisant une boucle séquentielle.

2. Calcul par carrés

Écrire et tester un algorithme qui réalise les opérations ci-dessous.

- a. initialiser la variable `nombre` à `a`, la variable `exposant` à `n`, et la variable `produit` à 1.
- b. lancer une boucle conditionnelle qui se termine quand le cas où `exposant` vaut 1 a été traité
- c. si n est pair, le diviser par 2, et affecter la variable `nombre` à son carré
- d. traiter judicieusement le cas où n est impair.
- e. retourner le résultat a^n .

3. Comparaison du nombre d'opérations

Comparer le nombre d'opérations réalisées dans chacun des deux calculs en fonction de la valeur de l'exposant n .

~