

Chapitre 11

Rechercher une valeur dans un tableau

1 Recherche générale dans un tableau

On suppose qu'on a un tableau `T` contenant des valeurs, on sait éventuellement que ce sont des entiers (naturels ou relatifs) ou encore des nombres réels (on dit nombre flottant en informatique) mais on ne sait rien d'autre sur ces valeurs. On verra plus loin que si on sait que le tableau est trié, alors il est possible d'en tirer parti. Comme ici, on n'a pas de connaissance particulière sur les valeurs contenues dans le tableau `T`, on parle de recherche générale.

On se pose la question de savoir si une `valeur` est dans le tableau `T` ou non.

Le principe de l'algorithme de recherche générale est de parcourir le tableau, élément par élément et de tester à chaque fois si l'élément est la valeur recherchée. Les différents cas à considérer sont les suivants :

- `valeur` n'est pas dans `T`
- `valeur` est présente une seule fois dans `T`
- `valeur` est présente plusieurs fois dans `T`

Exercice 28 (Recherche dichotomique)

1. Ecrire une fonction `recherche` qui prend en argument un tableau `T` et `valeur` la valeur recherchée. La fonction doit renvoyer `True` la première fois que `valeur` est rencontrée si elle est présente plusieurs fois dans le tableau, `False` sinon.
2. Dans le meilleur des cas, quel est le taux de croissance ? On répondra en utilisant la notation $\mathcal{O}()$.
3. Dans le pire des cas ?

```
def recherche(T, valeur) :  
    """  
    recherche valeur dans T sans aucune connaissance sur T  
    renvoie True/False  
    """  
    for x in T :  
        if x == valeur :  
            return True  
    return False
```

2 Recherche dichotomique dans un tableau trié

2.1 Hypothèse de base

On a vu que rechercher une valeur dans un tableau T de taille n se faisait en $\mathcal{O}(n)$. Peut-on faire mieux ? A priori non, car on ne sait rien sur les valeurs.

Hypothèse de base : on suppose maintenant que T est **trié** par ordre croissant. Ce qui va nous permettre **de ne pas parcourir tout le tableau** ! On ne peut pas faire de recherche dichotomique si le tableau n'est pas trié !

2.2 Principe de la recherche dichotomique

Le mot *dichotomie* signifie *couper en deux*.

Soit le tableau $T=[0,10,20,30,40,50,60,70,80,90]$ **trié** par ordre croissant.

Je recherche la valeur $x=10$ par dichotomie.

Je note $T[g..d]$ le sous-tableau de T dans lequel je recherche effectivement la valeur x . Les indices sont g pour gauche et d pour droite. Initialement, on recherche x dans tout le tableau, donc on a $g=0$ et $d=\text{len}(T)-1$.

Etape 1 : je recherche dans le tableau entier entre les indices $g=0$ et $d=9$:

je me place au milieu du tableau : $m = (0 + 9) // 2 = 4$

$T[4]$ correspond à 40 : la valeur recherchée 10 est inférieure à 40. Dans quelle partie du tableau peut se trouver la valeur recherchée **puisque qu'on sait que le tableau est trié par ordre croissant** ?

→ dans la partie de gauche

Etape 2 : je recherche dans le tableau entier entre les indices $g=0$ et $d=4-1=3$:

je me place au milieu du tableau : $m = (0 + 3) // 2 = 1$

$T[1]$ correspond à 10 : j'ai trouvé la valeur recherchée, je m'arrête. La valeur recherchée a été trouvée ici en 2 étapes, il aurait fallu 5 étapes pour la trouver avec une recherche générale qui parcourt tout le tableau.

Exercice 29 (et si la valeur n'est pas dans T)

Reprendre cet exercice en recherchant la valeur $x=72$.

Etape 1 : je recherche dans le tableau entier entre les indices $g=0$ et $d=9$:

je me place au milieu du tableau : $m = (0 + 9) // 2 = 4$

$T[4]$ correspond à 40 : la valeur recherchée 72 est supérieure à 40 → je dois chercher dans la partie de droite

Etape 2 : je recherche dans le tableau entier entre les indices $g=4+1=5$ et $d=9$:

je me place au milieu du tableau : $m = (5 + 9) // 2 = 7$

$T[7]$ correspond à 70 : la valeur recherchée 72 est supérieure à 70 → je dois chercher dans la partie de droite

Etape 3 : je recherche dans le tableau entier entre les indices $7+1=8$ et 9 :

je me place au milieu du tableau : $m = (8 + 9) // 2 = 8$

$T[8]$ correspond à 80 : la valeur recherchée 72 est inférieure à 80 → je dois chercher dans la partie de gauche

Etape 4 : je recherche dans le tableau entier entre les indices $g=7+1=8$ et $d=8-1=7$:

Le tableau entre l'indice de début 8 et l'indice de fin 7 n'existe pas : donc la valeur recherchée n'est pas dans T.

Condition d'arrêt

On a trouvé la **condition d'arrêt** de la recherche dichotomique :

- la suite des valeurs de l'indice gauche g est **croissante**,
- celle de l'indice droite d **décroissante**,

la recherche dichotomique a lieu tant que $g \leq d$ et finie quand $g > d$.

2.3 Implémentation python de la recherche dichotomique

Exercice 30 (Recherche dichotomique)

Ecrire une fonction `recherche_dichotomique(T, x)` qui applique le principe vu ci-dessus. À faire à la maison. La fonction doit renvoyer `True` si x est dans T, `False` sinon. On appellera g et d les indices de gauche et de droite entre lesquels on recherche la valeur x . L'indice m représentera le milieu (valeur entière) entre les indices g et d .

```
def recherche_dichotomique(T, x) :  
    g, d = 0, len(t)-1  
    while g <= d :  
        m = (g+d)//2  
        if T[m] == x :  
            return True  
        if T[m] > x :  
            d = m - 1  
        else :  
            g = m + 1  
    return False
```

2.4 Taux de croissance de la recherche dichotomique

Dans le pire des cas, la valeur recherchée n'est pas dans T.

On part d'un tableau de taille n qu'on divise en deux, puis de nouveau en 2, ..., jusqu'à ce que sa taille soit inférieure ou égale à 1

On fait l'inverse : on part d'un tableau de taille 1 et on double la taille du tableau à chaque étape. Combien faut-il d'étape pour que la taille du tableau soit égale à n ?

Soit x le nombre d'étape. $2^x = n$ soit $x = \lg n$!

Conclusion générale

La recherche dichotomique dans un tableau de taille n est en $\mathcal{O}(\lg n)$
car le tableau T est **trié**. On ne parcourt pas tout le tableau.

Si on n'a pas d'information sur les valeurs du tableau T,
on est obligé de parcourir tout le tableau et la recherche est en $\mathcal{O}(n)$.