i iobiematique

classique dans i

Recherche dans les données

tableau trié

Des données ordonnées Recherche dichotomique Efficacité

Recherche dichotomique

Christophe Viroulaud

Première - NSI

Sommaire

Recherche dichotomique

1. Problématique

- 2. Recherche classique dans un tableau
- 3. Recherche dans un tableau trié

Problématique

Recherche classique dans un cableau

Recherche dans les données

Recherche dans

Des données ordonnées Recherche dichotomique

Problématique

Rechercher un élément dans un tableau est une opération courante. Cette tâche a un coût qui dépend de la taille du tableau. Cependant, si le tableau est déjà trié est-il possible d'accélérer la recherche?

Problématique

Recherche :lassique dans un ableau

Génération des données Recherche dans les données

ableau trié Des données ordonnées

Recherche dichotomique

Comment implémenter une recherche efficace dans un tableau trié?

Sommaire

Recherche dichotomique

1 Dual-14----

- 2. Recherche classique dans un tableau
- 2.1 Génération des données
- 2.2 Recherche dans les données
- 3. Recherche dans un tableau trié

Problématique

Recherche classique dans un tableau

Génération des données Recherche dans les données

Recherche dan

Des données ordonnées Recherche dichotomique

Imaginons un supermarché qui référence chaque article par un entier. Les références, au nombre de cent mille, sont contenues dans un tableau.



Problématique

tableau Génération des données

Deskeration des données

Reciference dans les données

tableau trié Des données ordonnées

echerche dichotom fficacité

Activité 1 : Construire par compréhension un tableau de cent mille entiers compris entre 0 et 1000000.

Correction

1

Recherche dichotomique

Problématique

Recherche classique da

Génération des données

Recherche dans les données

tableau trié

Des données ordonnées Recherche dichotomique

 $\mathsf{entiers} = [\mathsf{randint}(0,\,1000000) \; \mathsf{for} \; _\mathsf{in} \; \mathsf{range}(100000)]$

Jeu de données

Sommaire

Recherche dichotomique

1 Problématique

- 2. Recherche classique dans un tableau
- 2.1 Génération des données
- 2.2 Recherche dans les données
- 3 Recherche dans un tableau trié

Problématique

classique dans u

Generation des données

Recherche dans les données

Recherche dans tableau trié

Des données ordonnées Recherche dichotomique

Recherche dans les données

Pour vérifier la présence d'une valeur dans les données, il faut parcourir le tableau élément par élément.

3 180 1007 56	217	'8 8
---------------	-----	------

FIGURE 1 – Parcours séquentiel

Recherche dichotomique

Problématique

Recherche classique da

Génération des données

Recherche dans les données

tableau trié

Des données ordonnées

echerche dichotomique fficacité

Dans le pire des cas

Recherche dichotomique

Problématique

Recherche classique da

Génération des données

Recherche dans les données

Recherche dans u tableau trié

> es données ordonnées echerche dichotomique

nombre d'éléments	nombre de comparaisons
100	100
10000	10000
1000000	1000000

À retenir

Dans le pire des cas le nombre d'opérations de la recherche dépend du nombre d'éléments.

La complexité est **linéaire**.

Recherche dichotomique

Problématique

classique dans u tableau

Génération des données

Recherche dans les données

Recherche dans u cableau trié

Des données ordonnées Recherche dichotomique Efficacité

Activité 2 :

- Écrire la fonction recherche_classique(tab: list, cherche: int) → bool qui renvoie True si l'entier cherche est présent dans le tableau.
- 2. Tester la fonction : vérifier si le nombre 575000 a été choisi par une personne.

Problématique

echerche assique dans ı

Recherche dans les données

cableau trié

Des données ordonnées

Recherche dichotomique Efficacité 1

3

5

6

8

9

return False

```
def recherche_classique(tab: list, cherche: int) -> bool:
    """
    Renvoie True si 'cherche' est dans 'tab'
    """
    for element in tab:
        if element == cherche:
            return True
```

à la fin de la boucle on n'a pas trouvé 'cherche'

Problématique

Recherche Hassique dan

tableau

Recherche dans les données

5

tableau trié

Des données ordonnées

Des données ordonnées Recherche dichotomique Efficacité

Activité 2 :

3. Dans le programme principal, créer une variable COMPTEUR initialisée à 0. Cette variable de test sera utilisée dans la fonction pour compter le nombre d'itérations de la boucle de recherche. On parle alors de variable globale car elle n'est pas propre à la fonction. Il faudra ajouter le code 1 au début de la fonction.

1 global COMPTEUR

Code 1 – Déclaration d'une variable globale

Problématique

classique dans un tableau

Recherche dans les données

ableau trié
Des données ordonnées

Recherche dichotor

```
COMPTFUR = 0
 1
 2
    def recherche_classique(tab: list, cherche: int) -> bool:
 3
 4
       Renvoie True si 'cherche' est dans 'tab'
 5
       77 77 77
 6
       global COMPTEUR
       for element in tab:
 8
          COMPTEUR += 1
 9
          if element == cherche:
10
              return True
11
12
       # à la fin de la boucle on n'a pas trouvé 'cherche'
       return False
13
```

classique dans

Génération des données

Recherche dans les données

ableau trié

Des données ordonnées Recherche dichotomique

Sommaire

Recherche dichotomique

1 Problématique

2. Recherche classique dans un tablea

- 3. Recherche dans un tableau trié
- 3.1 Des données ordonnées
- 3.2 Recherche dichotomique
- 3.3 Efficacité

Problématique

Recherche Hassique dans un

Recherche dans les données

Recherche dans un tableau trié

Des données ordonnées Recherche dichotomique Efficacité

tableau

Recherche dans les données

tableau trié

Des données ordonnées

Recherche dichotomique Efficacité

Considérons maintenant que les références sont triées par ordre croissant au fur et à mesure de leur ajout dans le tableau de données.

alimentaire			V	êtemer	nt	é	électro-ménage			
3	8	56	180		1007	2178	8000		11600	12130

FIGURE 2 - Références triées

Problématique

classique dans ur tableau

Recherche dans les données

Recherche da tableau trié

Des données ordonnées

Recherche dichotomique Efficacité

Activité 3 : Pour simplifier nous allons utiliser la méthode **sort** pour trier les données.

- 1. Construire par compréhension un tableau de cent mille entiers compris entre 0 et 1000000.
- 2. Trier le tableau.

Correction

1

```
Recherche dichotomique
```

Problématique

classique dans

Recherche dans les données

tableau trié

Des données ordonnées

Des données ordonnées

Recherche dichotomique Efficacité

```
\begin{array}{l} \mathsf{entiers} = [\mathsf{randint}(0,\,1000000) \; \mathsf{for} \; \_ \; \mathsf{in} \; \mathsf{range}(100000)] \\ \mathsf{entiers.sort}() \end{array}
```

Jeu de données

Sommaire

- Recherche dichotomique
- Problèmatique
- tableau
- Recherche dans les données
- Recherche dans tableau trié
- Des données ordonnées
- Recherche dichotomique Efficacité

- 1. Problématique
- 2. Recherche classique dans un tableau
- 3. Recherche dans un tableau trié
- 3.1 Des données ordonnées
- 3.2 Recherche dichotomique
- 3.3 Efficacité

Recherche dans les données

Des données ordonnées

Recherche dichotomique

Les données étant triées, le principe de la dichotomie, pour chercher la présence d'un élément, consiste à :

- couper le tableau en deux parties égales,
- ne garder que la partie contenant l'élément,
- répéter l'opération jusqu'à trouver l'élément ou avoir une partie vide.

Problématique

echerche

Génération des données

Recherche dans

Des données ordonnées

Recherche dichotomique

Efficacité

Cherchons 302 dans le tableau suivant :

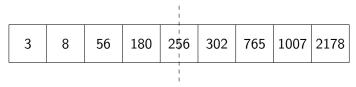


FIGURE 3 – Séparons les données en deux parties

3	8	56	180	256	302	765	1007	2178

FIGURE 4 – 256 n'est pas le nombre recherché et il est inférieur à 302

Recherche dichotomique

Recherche dichotomique

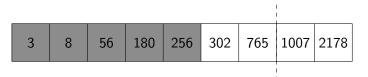


FIGURE 5 – Séparons les données restantes en deux parties

Problématique

lecherche lassique da

Génération des données

techerche da ableau trié

Des données ordonnées

Recherche dichotomique Efficacité

3 8 56 180 256	302 765	1007 2178
----------------	---------	-----------

FIGURE 6 – Nous pouvons éliminer la partie supérieure.

Problématique

Recherche lassique da

Génération des données

Recherche dar ableau trié

Des données ordonnées Recherche dichotomique

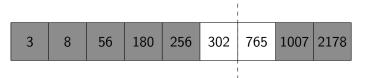


FIGURE 7 – Dernière séparation

Problématique

Recherche lassique da

Génération des données

Recherche dans ableau trié

Des données ordonnées Recherche dichotomique

cnercne dicnotom icacité

3 8	56	180	256	302	765	1007	2178
-----	----	-----	-----	-----	-----	------	------

FIGURE 8 – 302 a été trouvée en trois itérations

Problématique

echerche lassique dar

Génération des données

Recherche dans

Des données ordonnées

Recherche dichotomique
Efficacité

En pratique, on utilise les indices pour trouver le milieu.

0	1	2	3	4	5	6	7	8
3	8	56	180	256	302	765	1007	2178

FIGURE 9 –
$$\frac{8+0}{2}$$
 = 4 l'indice médian est 4

Recherche dichotomique

Problèmatique

Recherche classique dans

Génération des données Recherche dans les données

ableau trié Des données ordonnées

Des données ordonnées

Recherche dichotomique

Efficacité

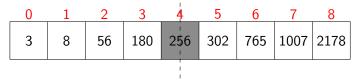


FIGURE 10 – 256 n'est pas le nombre recherché

Problématique

recnercne classique dans cableau

Recherche dans les données

ableau trié

Recherche dichotomique

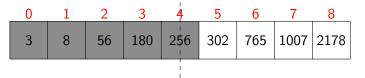


FIGURE 11 – 256 est inférieur au nombre recherché.

```
1 i_debut = 5 i_fin = 8
```

Problèmatique

classique dans un tableau

Recherche dans les données

ableau trié
Des données ordonnées

Recherche dichotomique

0 1 2 3 4 5 6 7 8 3 8 56 180 256 302 765 1007 2178

FIGURE $12 - \frac{8+5}{2} = 6$ l'indice médian est 6

Recherche dichotomique

Problématique

Recherche classique dans

Recherche dans les données

ableau trié

Des données ordonnées Recherche dichotomique Efficacité

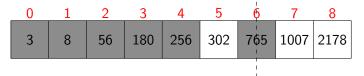


FIGURE 13 – 765 n'est pas le nombre recherché.

Problématique

Recherche lassique dan

Génération des données Recherche dans les données

> echerche dans Ibleau trié

Des données ordonnées

Recherche dichotomique Efficacité

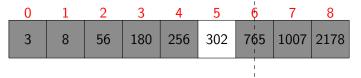


FIGURE 14 – 765 est supérieur au nombre recherché.

```
1 | i_debut = 5 | i_fin = 5
```

Problèmatique

classique dans un tableau

Recherche dans les données

ableau trié
Des données ordonnées

Des données ordonnées Recherche dichotomique

0 1 2 3 4 5 6 7 8 3 8 56 180 256 302 765 1007 2178

FIGURE 15 – $\frac{5+5}{2}$ = 5 l'indice médian est 5.

Recherche dichotomique

Problématique

classique dans un tableau

Recherche dans les données

ableau trié

Des données ordonnées Recherche dichotomique Efficacité

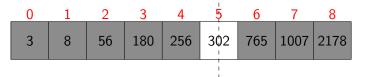


FIGURE 16 – On a trouvé l'élément.

Problématique

classique dans ui cableau

Recherche dans les données

ableau trié

Des données ordonnées Recherche dichotomique Efficacité

Problématique

classique dans un tableau

Recherche dans les données

ableau trié Des données ordonnées

Recherche dichotomique

Efficacité

Activité 4 : Écrire la fonction recherche_dicho(tab: list, cherche: int) → bool qui applique le principe de la dichotomie. Pour séparer les données en deux parties (à peu près) égales il faudra calculer l'indice médian de la partie encore valide.

0	1	2	3	4	5	6	7	8
3	8	56	180	256	302	765	1007	2178

```
def recherche_dicho(tab: list, cherche: int) -> bool:
    i_debut = 0
    i_fin = len(tab)-1
```

Problématique

classique dans u cableau

Recherche dans les données

ableau trié Des données ordonnées

Recherche dichotomique

6	7	8	

Recherche dans les données

Des données ordonnées

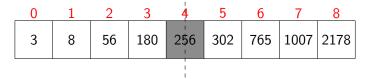
Recherche dichotomique

```
    0
    1
    2
    3
    4
    5
    6
    7
    8

    3
    8
    56
    180
    256
    302
    765
    1007
    2178
```

```
while i_fin >= i_debut:
    i_milieu = (i_debut+i_fin) // 2
```

Correction



```
if cherche == tab[i_milieu] :
return True
```

Recherche dichotomique

Problématique

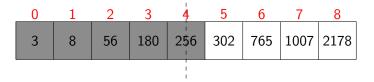
classique o

Génération des données

echerche dans ableau trié

Des données ordonnées Recherche dichotomique

Correction



```
elif cherche < tab[i_milieu] :
    i_fin = i_milieu-1
```

Recherche dichotomique

Problématique

Recherche classique d

Génération des données Recherche dans les données

ableau trié

Des données ordonnées Recherche dichotomique

```
def recherche_dicho(tab: list, cherche: int) -> bool:
 1
       i debut = 0
       i fin = len(tab)-1
 3
       while i fin >= i debut:
 4
 5
          i_milieu = (i_debut+i_fin) // 2
          if cherche == tab[i_milieu] :
 6
              return True
          elif cherche < tab[i milieu] :
 8
              i fin = i milieu-1
 9
          else: # cherche > tab[i milieu]
10
              i debut = i milieu+1
11
       # à la fin de la boucle on n'a pas trouvé 'cherche'
12
       return False
13
```

Problématique

classique dans ι tableau

Recherche dans les données

Recherche dans tableau trié

Des données ordonnées Recherche dichotomique

herche dichotomie cacité

Sommaire

Recherche dichotomique

1 Problématique

2. Recherche classique dans un tablea

- 3. Recherche dans un tableau trié
- 3.1 Des données ordonnées
- 3.2 Recherche dichotomique
- 3.3 Efficacité

Problématique

echerche

tableau Génération des données

Recherche dans les données

tableau trié

Des données ordonnées Recherche dichotomique Efficacité

Recherche dans les donnée

tableau trié

Des données ordonnées

Efficacité

Activité 5:

- En utilisant une variable COMPTEUR, compter le nombre d'itérations de la boucle de recherche dichotomique.
- 2. Tester pour différentes tailles de tableau.

```
COMPTFUR = 0
 1
 2
 3
    def recherche dicho(tab: list, cherche: int) -> bool:
       global COMPTEUR
 4
       i debut = 0
 5
       i fin = len(tab)-1
 6
       while i fin >= i debut:
 7
          COMPTEUR += 1
 8
          i_milieu = (i_debut+i_fin) // 2
 9
          if cherche == tab[i_milieu] :
10
              return True
11
          elif cherche < tab[i_milieu] :</pre>
12
13
              i fin = i milieu-1
          else: # cherche > tab[i milieu]
14
              i debut = i milieu+1
15
       # à la fin de la boucle on n'a pas trouvé 'cherche'
16
       return False
17
```

Problématique

ecnercne assique dans ui

Génération des données

ableau trié

es données ordonnées echerche dichotomique

Recherche dichotomique

Problématique

classique dans un tableau

Recherche dans les données

ableau trié

Des données ordonnées

Efficacité

$$\frac{n}{2^x} = 1$$

$$\Leftrightarrow n = 2^x$$

À chaque itération la quantité de données (notée n) à

égale à 1.

étudier est divisée par deux. Dans le pire des cas, on divise jusqu'à ce que la taille de la partie restante soit inférieure ou

Recherche dichotomique

Problématique

tableau

Recherche dans les données

ableau trié

Des données ordonnées Recherche dichotomique

Efficacité

Activité 6 :

- 1. Encadrer la valeur de x entre deux entiers, si le tableau contient n=10000 éléments.
- 2. Effectuer le même encadrement pour cent mille, un million d'éléments.

Correction

Recherche dichotomique

Problématique

decherche lassique dans un ableau

Génération des données Recherche dans les données

Recherche dar tableau trié

Des données ordonnées Recherche dichotomique

$$2^{13} = 8192 < x < 2^{14} = 16384$$

Dans le pire des cas

nombre d'éléments	nombre de comparaisons		
10	3-4		
100	6-7		
1000	9-10		
10000	13-14		
100000	16-17		
1000000	19-20		

Recherche dichotomique

Problématique

Recherche classique da

Génération des données

echerche dan Ibleau trié

les données ordonnées lecherche dichotomique

À retenir

La complexité temporelle de la recherche dichotomique est logarithmique:

$$\log_2 n = x$$

Recherche dichotomique

Code complet

Recherche dichotomique

Problématique

Recherche classique dans un

Génération des données Recherche dans les données

tableau trié

Des données ordonnées Recherche dichotomique Efficacité

Le code complet se trouve ici.