

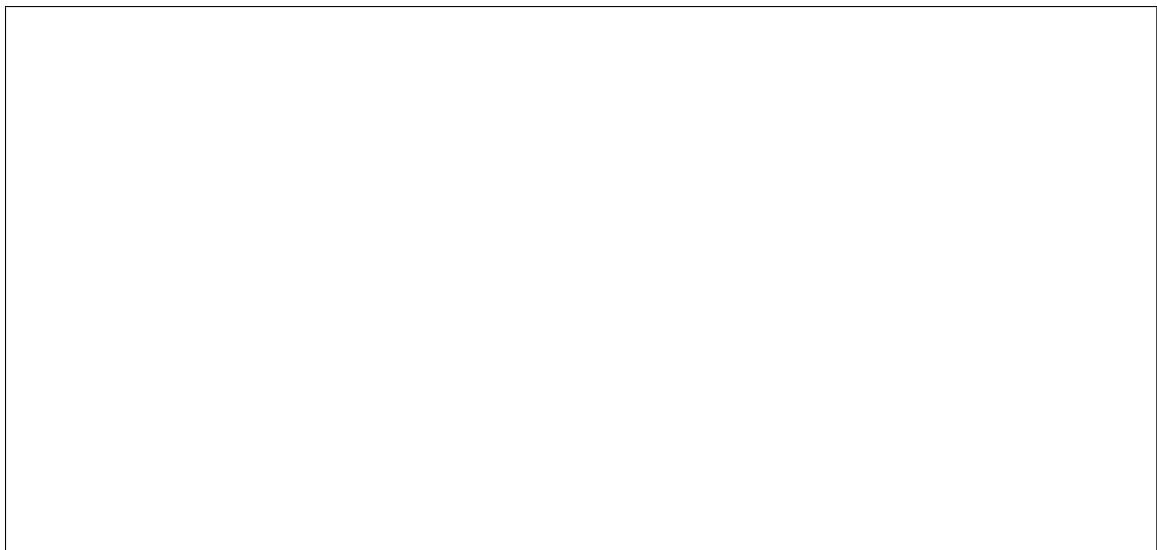
Recherche par dichotomie dans un tableau trié (synthèse):

Préambule: les algorithmes utilisant des tableaux de taille n seront indicés de 0 à $(n-1)$.

Algorithm 1: Recherche par dichotomie dans un tableau trie

```
1 Function recherche (tab,x):
2   debut  $\leftarrow$  0
3   fin  $\leftarrow$  len(tab) - 1
4   while debut  $\leq$  fin do
5     central  $\leftarrow$   $\lfloor$ (debut + fin)/2 $\rfloor$ 
6     val  $\leftarrow$  tab[central]
7     if val = x then
8        $\lfloor$  return True
9     else if val < x then
10       $\lfloor$  debut  $\leftarrow$  central + 1
11    else
12       $\lfloor$  fin  $\leftarrow$  central - 1
13  return False
```

1. Recherche de l'élément 12 dans la liste triée [2,5,9,12,13,24,34,35,46]



2. Recherche de l'élément 30 dans la liste triée [2,5,9,12,13,24,34,35,46]

A large, empty rectangular box with a thin black border, intended for the student to write their solution for finding the element 30 in the sorted list [2,5,9,12,13,24,34,35,46].

3. complexité:

A large, empty rectangular box with a thin black border, intended for the student to write their analysis of the complexity of the search algorithm.