

Représentation par tableaux ordonnés et recherche dichotomique

E est un ensemble d'éléments représenté par une structure avec un champ $E.table$ qui est un tableau ordonné en ordre croissant, et un champ $E.dernier$ position du dernier élément. E est vide si $E.dernier = 0$.

x est un élément.

Algorithme 1 Fonction Insérer(x, E)

```
 $i := E.dernier;$   
tant que  $i > 0$  et  $x < E.table[i]$  faire  
     $E.table[i + 1] := E.table[i];$   
     $i := i - 1;$   
fin tant que  
 $E.table[i + 1] := x;$   
 $E.dernier := E.dernier + 1;$ 
```

Complexité au pire en $\mathcal{O}(n)$ où $n = E.dernier$.

Recherche d'un élément x dans un ensemble E par l'algorithme de recherche dichotomique :

Algorithme 2 Fonction Dicho(x, E, g, d)

Entrée : $E.table$ tableau trié en ordre croissant. g et d indices du tableau, compris entre 1 et $E.dernier$.

Sortie : Position de x dans $E.table$ entre les indices g et d s'il y est, 0 sinon.

```
si  $g \leq d$  alors  
     $m := (g + d) \text{ div } 2;$   
    si  $x = E.table[m]$  alors  
        retourner ( $m$ )  
    sinon si  $x < E.table[m]$  alors  
        retourner (Dicho( $x, E, g, m - 1$ ))  
    sinon  
        retourner (Dicho( $x, E, m + 1, d$ ))  
    fin si  
sinon  
    retourner (0)  
fin si
```

L'appel à Dicho($x, E, 1, E.dernier$) renvoie la position de x dans E s'il y est, 0 sinon.

Complexité au pire en $\mathcal{O}(\log n)$ où $n = E.dernier$.

Pour supprimer un élément x , on peut rechercher sa position p par la recherche dichotomique, et ensuite, si $p \neq 0$, supprimer l'élément en position p dans $E.table$. Ou bien, rechercher x et le supprimer en même temps, ce qui est fait par l'algorithme ci-dessous.

Algorithme 3 Fonction Supprimer(x, E)

```
si  $E.dernier \neq 0$  alors
   $i := 1$ ;
  tant que  $i \leq E.dernier$  et  $x > E.table[i]$  faire
     $i := i + 1$ ;
  fin tant que
  si  $i > E.dernier$  ou  $x \neq E.table[i]$  alors
    retourner ( $x$  n'est pas dans  $E$ )
  sinon
    pour  $j$  de  $i$  à  $E.dernier - 1$  faire
       $E.table[j] := E.table[j + 1]$ 
    fin pour
     $E.dernier := E.dernier - 1$ ;
  fin si
fin si
```

Complexité au pire en $\mathcal{O}(n)$ où $n = E.dernier$.