

Fin des tables de hachages et arbres

Algorithmes et structures de données, 2025-2026

P. Albuquerque (B410) et O. Malaspinas (A401), ISC, HEPIA
2026-03-04

En partie inspiré des supports de cours de P. Albuquerque

Les tables de hachage

Préambule

- Ici, on ne considère pas le cas du chaînage en cas de collisions.
- L'insertion est construite avec une forme du type

```
index = h(key);  
while (table[index].state == occupied  
      && table[index].key != key) {  
    index = (index + k) % table_size; // attention à pas dépasser  
}  
table[index].key = key;  
table[index].state = occupied;
```

- Gestion de l'état d'une case *explicite*

```
enum state_t {empty, occupied, deleted};
```

L'insertion

Pseudocode ?

L'insertion

Pseudocode ?

```
rien insertion(table, clé, valeur) {  
    index = hash(clé)  
    tant que table[index].état == occupé  
        et table[index].clé != clé:  
        index = rehash(index, clé)  
  
    table[index].état = occupé  
    table[index].clé = clé  
    table[index].valeur = valeur  
}
```

Pseudocode ?

La suppression

Pseudocode ?

```
valeur suppression(table, clé):  
    index = hash(clé)  
    tant que table[index].état != vide:  
        si table[index].état == occupé  
            et table[index].clé == clé:  
                table[index].état = supprimé  
        sinon  
            index = rehash(index, clé)  
    }
```

Pseudocode ?

Pseudocode ?

```
booléen recherche(table, clé) {  
    index = hash(clé)  
    tant que table[index].état != vide:  
        si table[index].état == occupé  
            et table[index].clé == clé:  
                retourner vrai  
        sinon  
            index = rehash(index, clé)  
    retourner faux  
}
```

Écrivons le code !

- Mais avant :
 - Quelles sont les structures de données dont nous avons besoin ?
 - Y a-t-il des fonctions auxiliaires à écrire ?
 - Écrire les signatures des fonctions.

Écrivons le code !

- Mais avant :
 - Quelles sont les structures de données dont nous avons besoin ?
 - Y a-t-il des fonctions auxiliaires à écrire ?
 - Écrire les signatures des fonctions.

Structures de données

Écrivons le code !

- Mais avant :
 - Quelles sont les structures de données dont nous avons besoin ?
 - Y a-t-il des fonctions auxiliaires à écrire ?
 - Écrire les signatures des fonctions.

Structures de données

```
enum state_t {empty, deleted, occupied};
typedef char * key_t; // could be something different
typedef char * value_t;
struct cell_t {
    key_t key;
    value_t value;
    enum state_t state;
};
struct hm {
    struct cell_t *table;
    int capacity;
    int size;
};
```

Écrivons le code !

Fonctions auxiliaires

Écrivons le code !

Fonctions auxiliaires

```
static int hash(key_t key);  
static int rehash(int index, key_t key);  
static int find_index(hm h, key_t key);
```

Signature de l'API

Écrivons le code !

Fonctions auxiliaires

```
static int hash(key_t key);  
static int rehash(int index, key_t key);  
static int find_index(hm h, key_t key);
```

Signature de l'API

```
void hm_init(struct hm *h, int capacity);  
void hm_destroy(struct hm *h);  
bool hm_set(struct hm *h, key_t key, value_t *value);  
bool hm_get(struct hm h, key_t key, value_t *value);  
bool hm_remove(struct hm *h, key_t key, value_t *value);  
bool hm_search(struct hm h, key_t key);  
void hm_print(struct hm h);
```

Live code session !

0. Offered to you by HestiaVPN¹ !

1. The fastest way to connect to BBB !

Live code session !

0. Offered to you by HestiaVPN¹ !
1. Like the video.
2. Subscribe to the channel.
3. Use our one time voucher for ProtonVPN : PAULISAWESOME.
4. Consider donating on our patreon.

1. The fastest way to connect to BBB !

Les arbres

Les arbres : définition

“Un arbre est un graphe acyclique orienté possédant une unique racine, et tel que tous les nœuds sauf la racine ont un unique parent.”

Les arbres : définition

“Un arbre est un graphe acyclique orienté possédant une unique racine, et tel que tous les nœuds sauf la racine ont un unique parent.”

Santé !

Plus sérieusement

- Ensemble de **nœuds** et d'**arêtes** (graphe).
- Les arêtes relient les nœuds entre eux, mais pas n'importe comment : chaque nœud a au plus un **parent**.
- Le seul nœud sans parent est la **racine**.
- Chaque nœud a un nombre fini d'**enfants**.
- La hiérarchie des nœuds rend les arêtes **orientées** (parent -> enfants), et empêche les **cycles** (acyclique, orienté).
- La **feuille** ou **nœud terminal** est un nœud sans enfants.
- Le **niveau** est 1 à la racine et **niveau+1** pour les enfants.
- Le **degré** d'un nœud est le nombre d'enfants du nœud.

Les arbres : définition

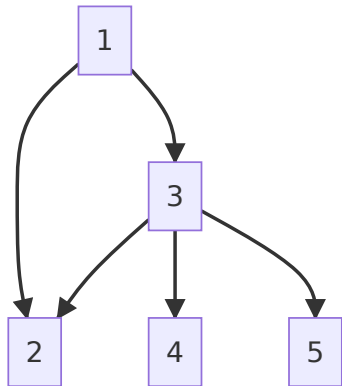
“Un arbre est un graphe acyclique orienté possédant une unique racine, et tel que tous les nœuds sauf la racine ont un unique parent.”

Santé !

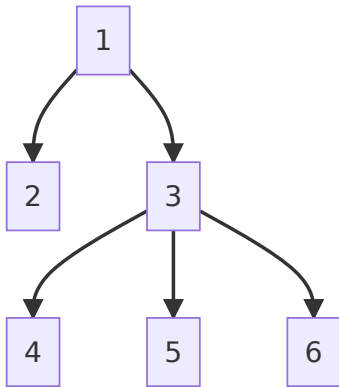
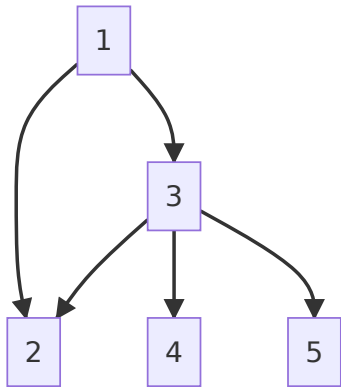
Plus sérieusement

- Ensemble de **nœuds** et d'**arêtes** (graphe).
- Les arêtes relient les nœuds entre eux, mais pas n'importe comment : chaque nœud a au plus un **parent**.
- Le seul nœud sans parent est la **racine**.
- Chaque nœud a un nombre fini d'**enfants**.
- La hiérarchie des nœuds rend les arêtes **orientées** (parent -> enfants), et empêche les **cycles** (acyclique, orienté).
- La **feuille** ou **nœud terminal** est un nœud sans enfants.
- Le **niveau** est 1 à la racine et **niveau+1** pour les enfants.
- Le **degré** d'un nœud est le nombre d'enfants du nœud.
- Chaque nœud est un arbre en lui-même.
- La **récurtivité** sera très utile !

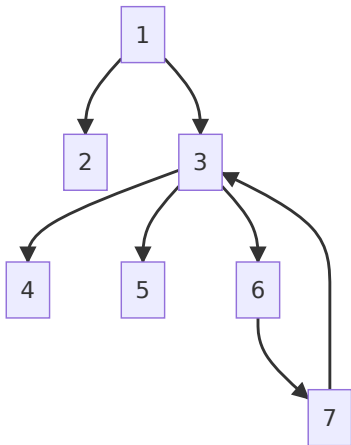
Arbre ou pas arbre ?



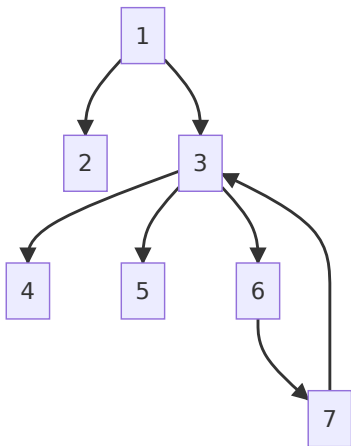
Arbre ou pas arbre ?



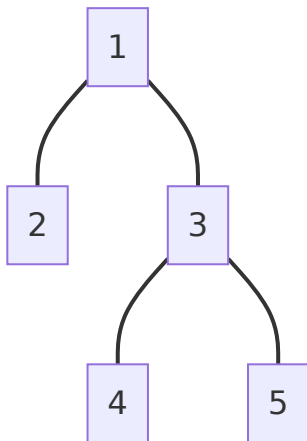
Arbre ou pas arbre ?



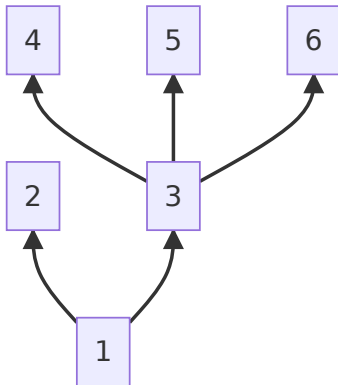
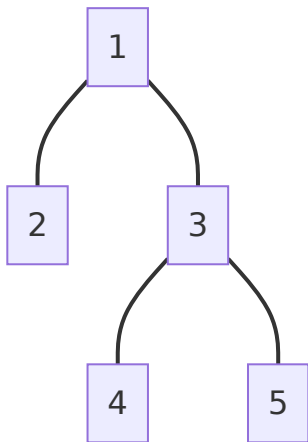
Arbre ou pas arbre ?



Arbre ou pas arbre ?

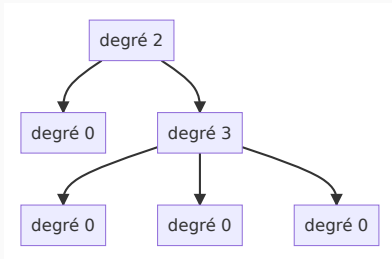


Arbre ou pas arbre ?



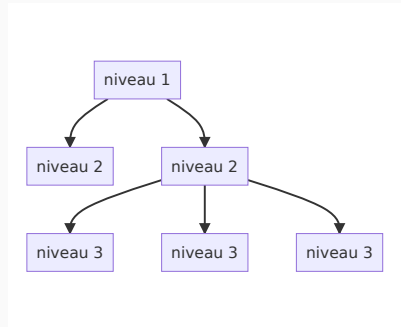
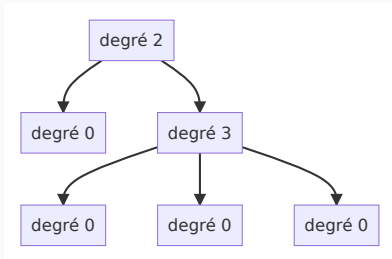
Degré et niveau

- Illustration du degré (nombre d'enfants) et du niveau (profondeur)



Degré et niveau

- Illustration du degré (nombre d'enfants) et du niveau (profondeur)



- Les nœuds de degré 0 sont des feuilles.

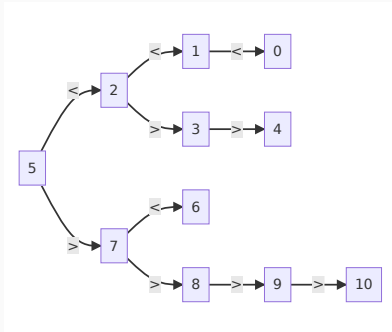
Application : recherche rapide

Pouvez-vous construire un arbre pour résoudre le nombre secret ?

Application : recherche rapide

Pouvez-vous construire un arbre pour résoudre le nombre secret ?

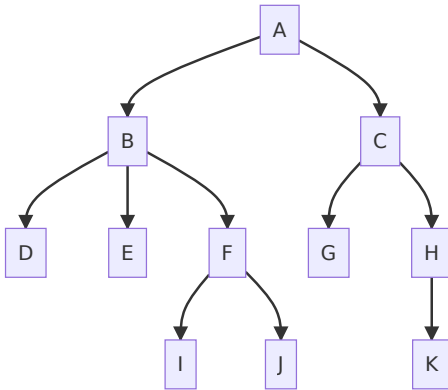
- Le nombre secret ou la recherche dichotomique (nombre entre 0 et 10).



Question : Quelle est la complexité pour trouver un nombre ?

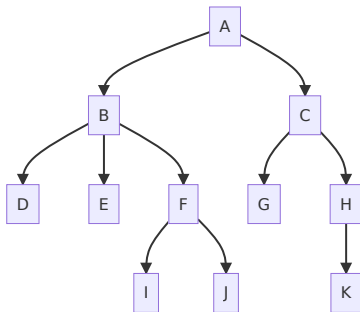
Autres représentations

- Botanique
- **Exercice** : Ajouter les degrés/niveaux et feuilles



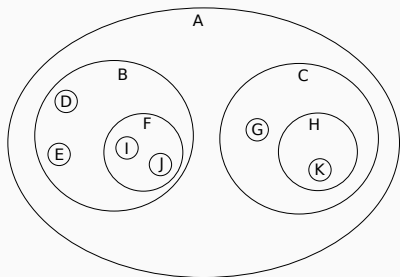
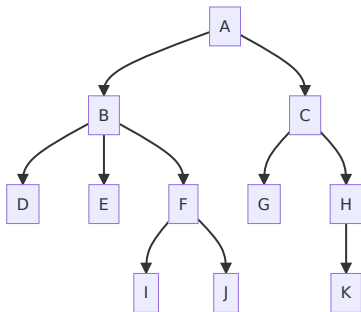
Autres représentations

- Ensembliste



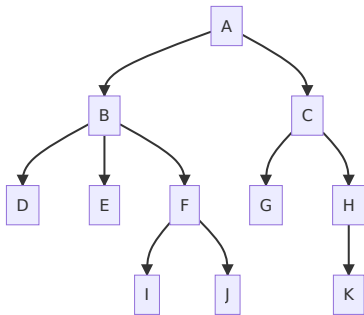
Autres représentations

- Ensembliste



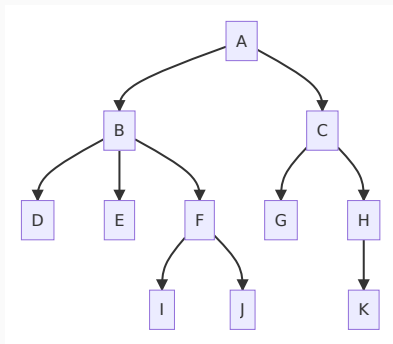
Autres représentations

- Liste



Autres représentations

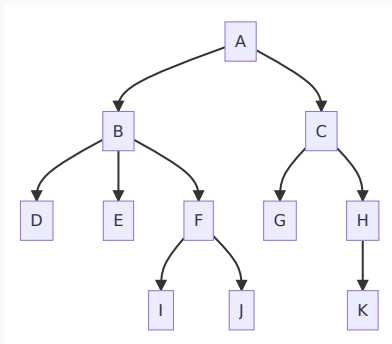
■ Liste



(A
 (B
 (D)
 (E)
 (F
 (I)
 (J)
)
)
 (C
 (G)
 (H
 (K)
)
)
)

Autres représentations

- Par niveau



Autres représentations

- Par niveau

