

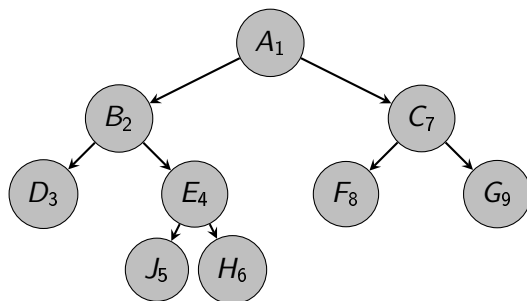
Quel ordre

Il s'agit de lire les nœuds en parcourant la descendance des nœuds avant d'aller voir les nœuds frères.

Par exemple, quand on en arrive à B, on parcourt les descendants de B avant d'envisager C et ses descendants.

Mais faut-il traiter B avant ses propres descendants ? après ?

Il y a plusieurs choix. Dans un premier temps, voyons cet ordre :

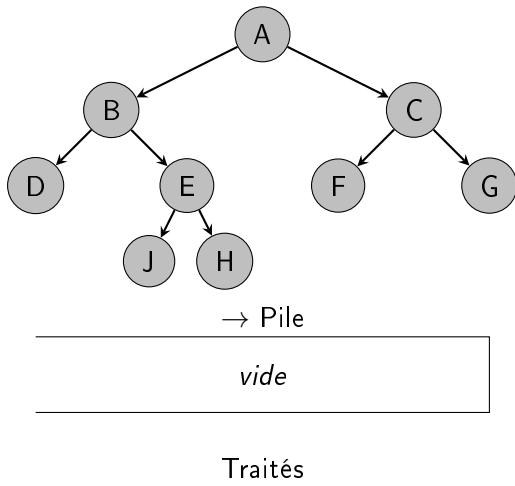


Algorithme utilisant une pile - 1

Entrées : arbre

```
1 Créer pile vide
2 Empiler racine de l'arbre
3 tant que Pile non vide
  faire
4   | Dépiler pile → nœud
5   | Empiler les enfants
6   | TRAITEMENT(nœud)
7 fin
```

Remarque : L'empilement des enfants se fait en ordre inverse.



Algorithme utilisant une pile - 2

Entrées : arbre

1 Créer pile vide

2 Empiler racine de l'arbre

3 **tant que** *Pile non vide*
 faire

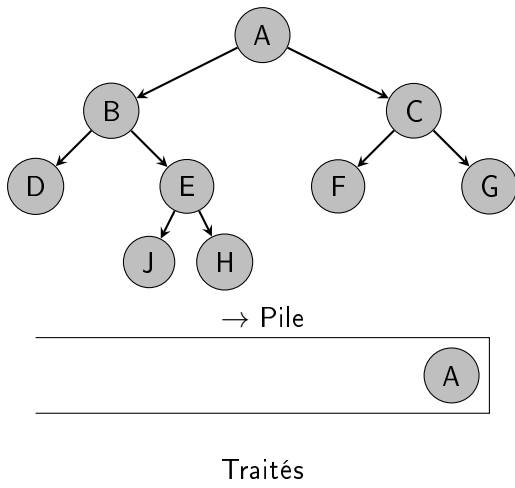
4 Dépiler pile $\rightarrow$ nœud

5 Empiler les enfants

6 TRAITEMENT(nœud)

7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.

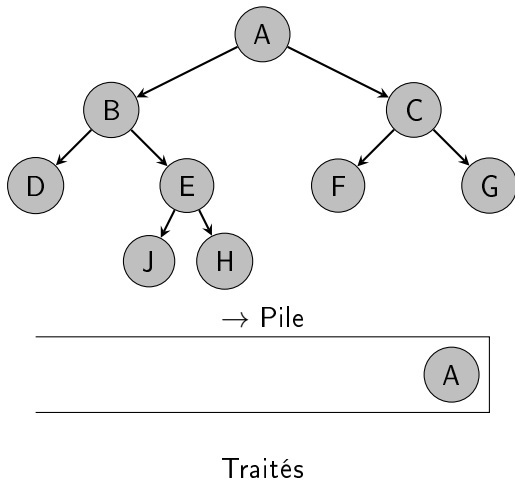


Algorithme utilisant une pile - 3

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.

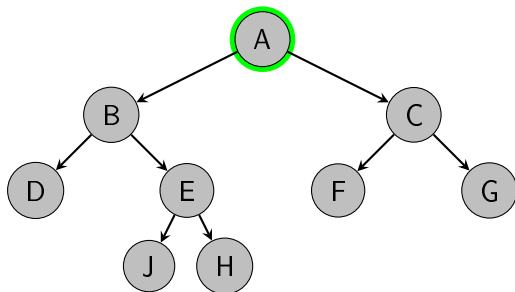


Algorithme utilisant une pile - 4

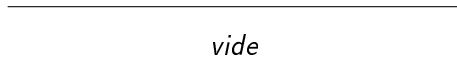
Entrées : arbre

```
1 Créer pile vide
2 Empiler racine de l'arbre
3 tant que Pile non vide
  faire
4   Dépiler pile → nœud
5   Empiler les enfants
6   TRAITEMENT(nœud)
7 fin
```

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



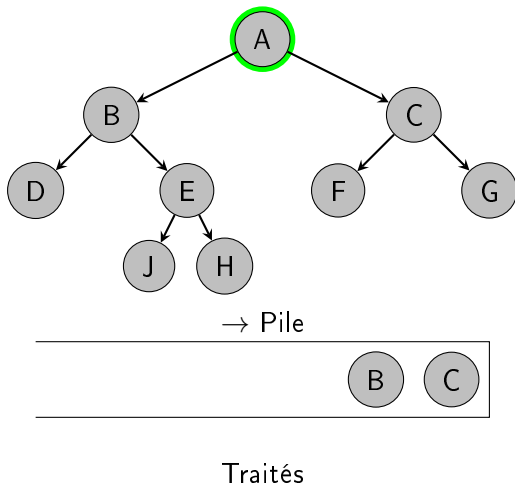
Traités

Algorithme utilisant une pile - 5

Entrées : arbre

```
1 Créer pile vide
2 Empiler racine de l'arbre
3 tant que Pile non vide
  faire
4   | Dépiler pile → nœud
5   | Empiler les enfants
6   | TRAITEMENT(nœud)
7 fin
```

Remarque : L'empilement des enfants se fait en ordre inverse.

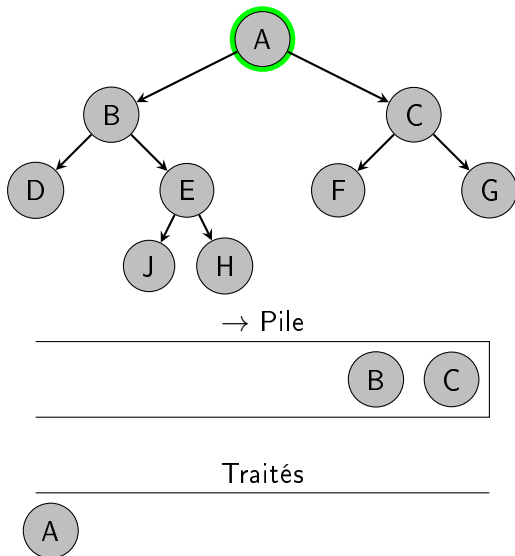


Algorithme utilisant une pile - 6

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 Dépiler pile → nœud
 - 5 Empiler les enfants
 - 6 **TRAITEMENT**(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.

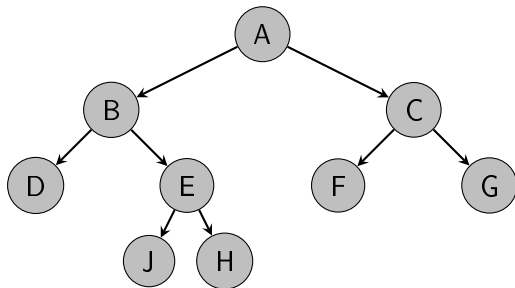


Algorithme utilisant une pile - 7

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

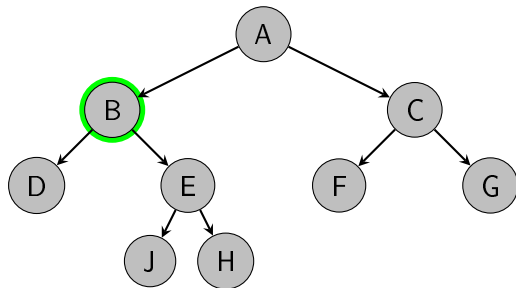


Algorithme utilisant une pile - 8

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 Dépiler pile → nœud
 - 5 Empiler les enfants
 - 6 TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

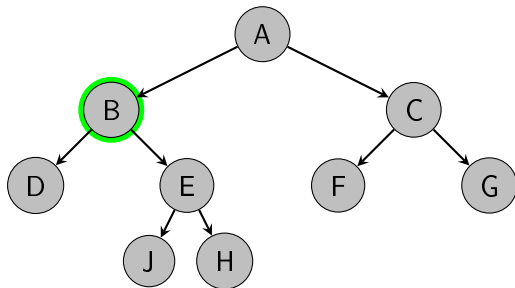


Algorithme utilisant une pile - 9

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 Dépiler pile → nœud
 - 5 Empiler les enfants
 - 6 TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

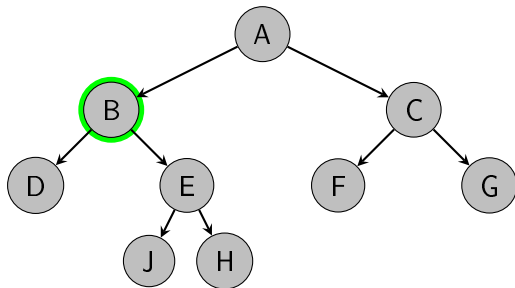


Algorithme utilisant une pile - 10

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | **TRAITEMENT**(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

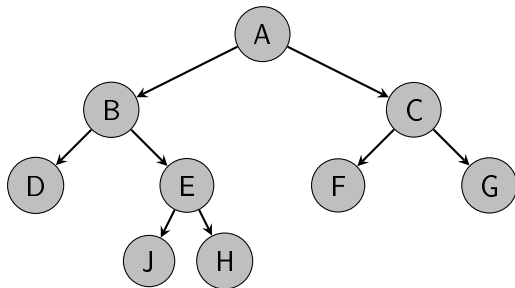


Algorithme utilisant une pile - 11

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

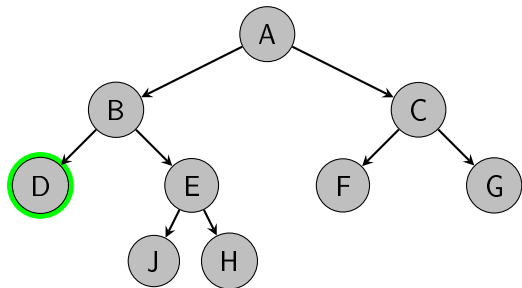


Algorithme utilisant une pile - 12

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | **Dépiler** pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

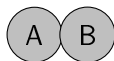
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

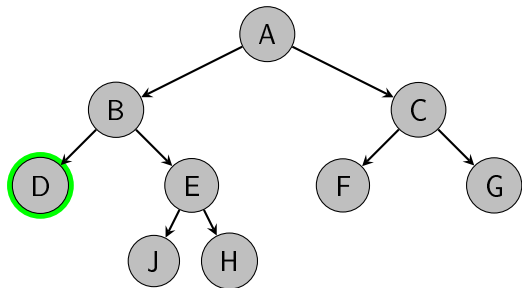


Algorithme utilisant une pile - 13

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | **Empiler les enfants**
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

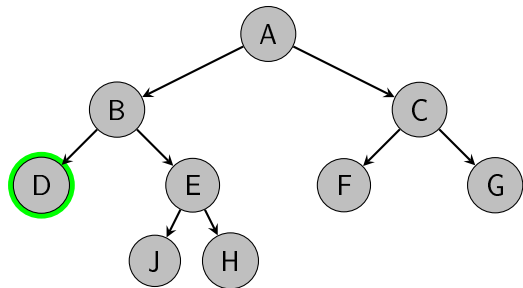


Algorithme utilisant une pile - 14

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | **TRAITEMENT**(nœud)
- 7 **fin**

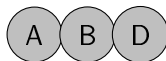
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

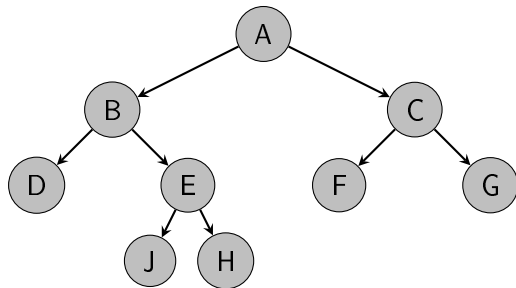


Algorithme utilisant une pile - 15

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

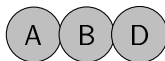
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

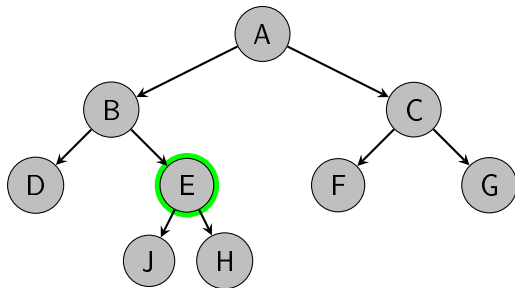


Algorithme utilisant une pile - 16

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | **Dépiler** pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

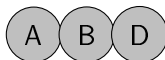
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

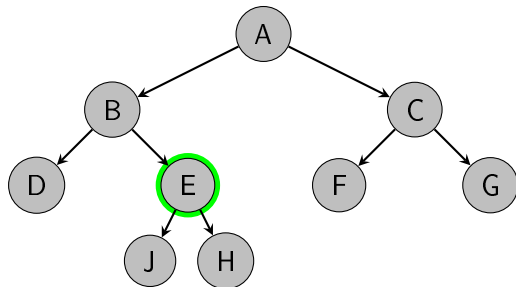


Algorithme utilisant une pile - 17

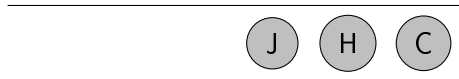
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | **Empiler les enfants**
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

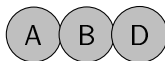
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

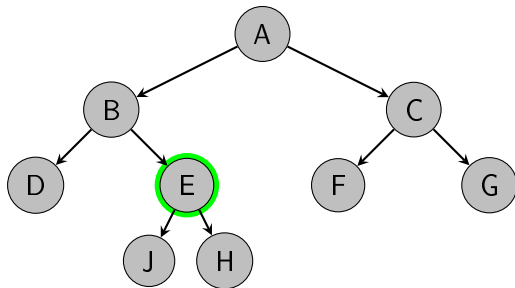


Algorithme utilisant une pile - 18

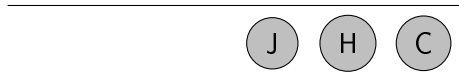
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | **TRAITEMENT**(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

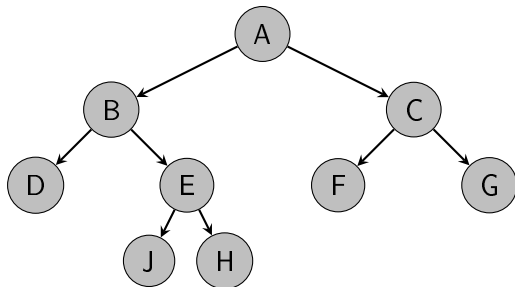


Algorithme utilisant une pile - 19

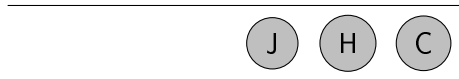
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

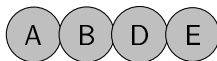
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

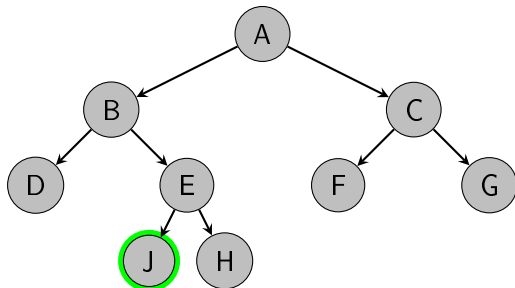


Algorithme utilisant une pile - 20

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

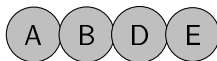
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

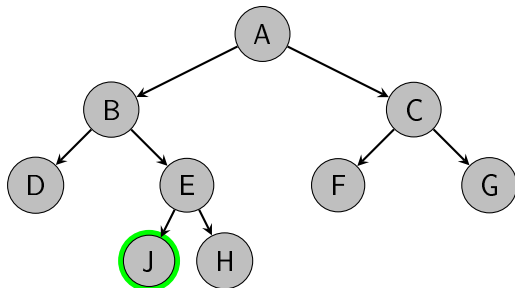


Algorithme utilisant une pile - 21

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | **Empiler les enfants**
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

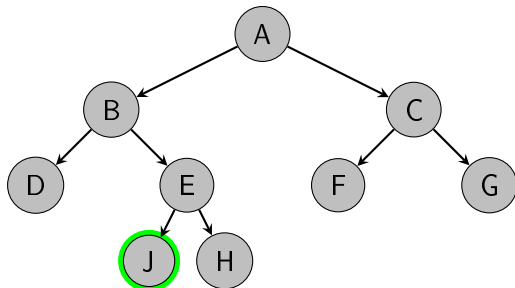


Algorithme utilisant une pile - 22

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | **TRAITEMENT**(nœud)
- 7 **fin**

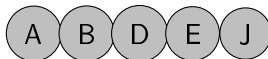
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

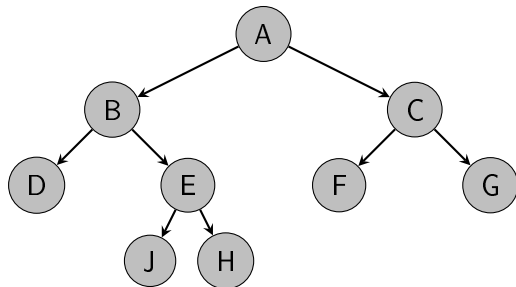


Algorithme utilisant une pile - 23

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

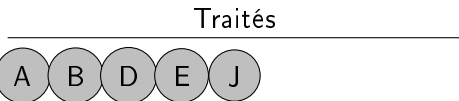
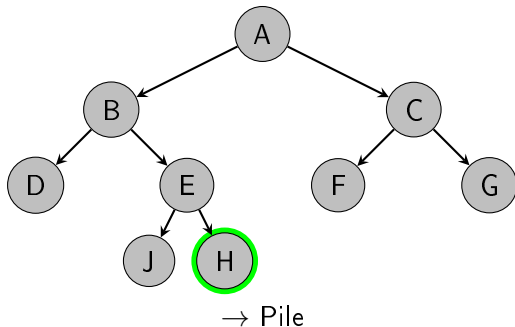


Algorithme utilisant une pile - 24

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.

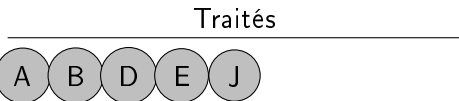
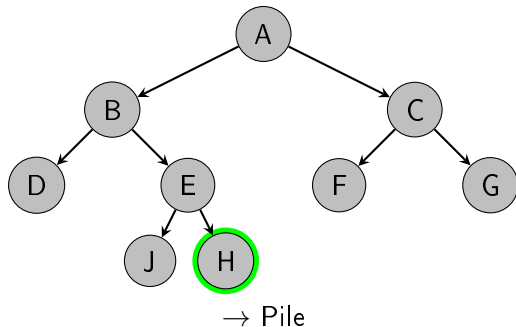


Algorithme utilisant une pile - 25

Entrées : arbre

```
1 Créer pile vide
2 Empiler racine de l'arbre
3 tant que Pile non vide
  faire
4   | Dépiler pile → nœud
5   | Empiler les enfants
6   | TRAITEMENT(nœud)
7 fin
```

Remarque : L'empilement des enfants se fait en ordre inverse.

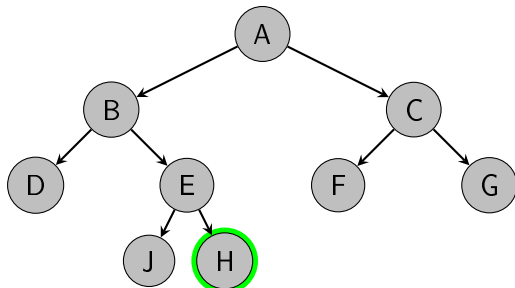


Algorithme utilisant une pile - 26

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 Dépiler pile → nœud
 - 5 Empiler les enfants
 - 6 **TRAITEMENT**(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

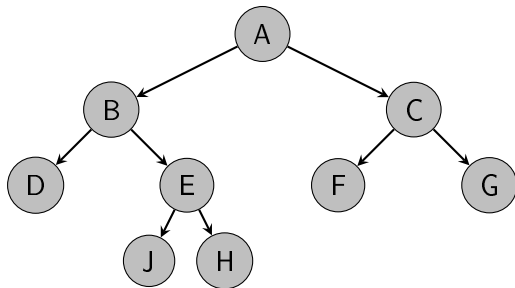


Algorithme utilisant une pile - 27

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

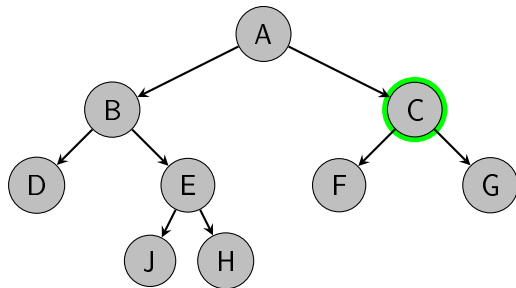


Algorithme utilisant une pile - 28

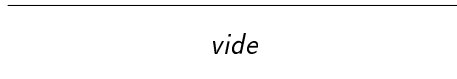
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

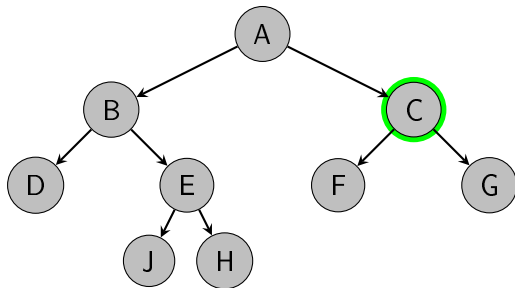


Algorithme utilisant une pile - 29

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 Dépiler pile → nœud
 - 5 Empiler les enfants
 - 6 TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

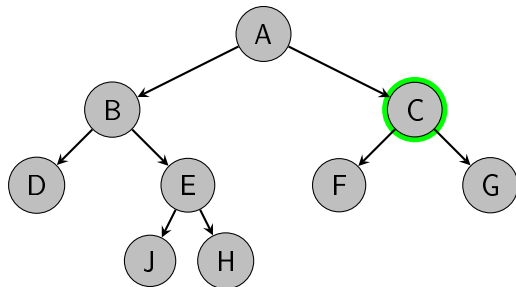


Algorithme utilisant une pile - 30

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 Dépiler pile → nœud
 - 5 Empiler les enfants
 - 6 **TRAITEMENT**(nœud)
- 7 **fin**

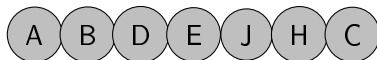
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

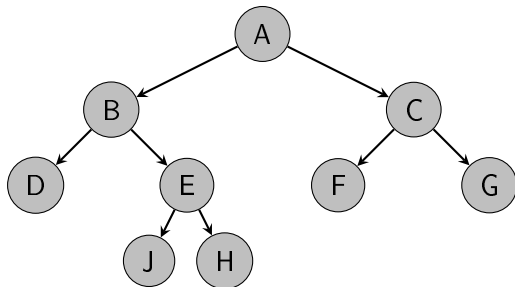


Algorithme utilisant une pile - 31

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

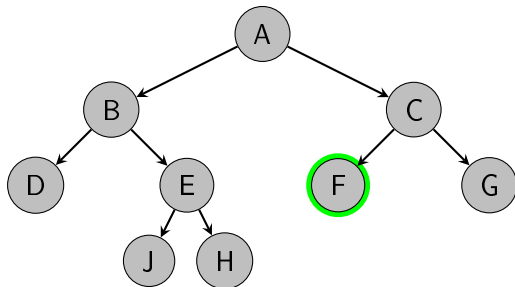


Algorithme utilisant une pile - 32

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

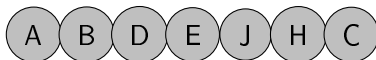
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

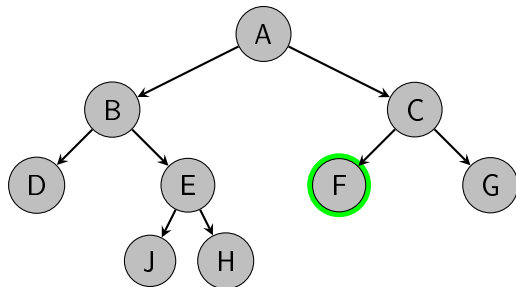


Algorithme utilisant une pile - 33

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | **Empiler les enfants**
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

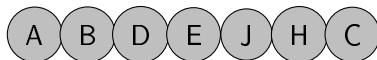
Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

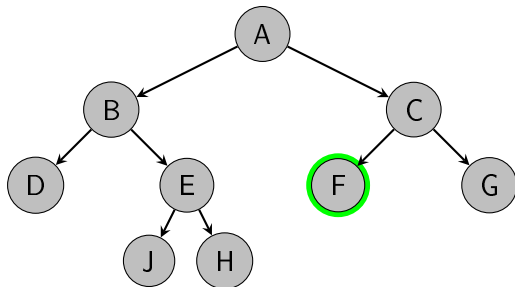


Algorithme utilisant une pile - 34

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | **TRAITEMENT**(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

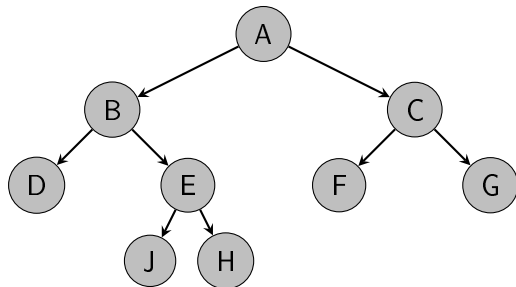


Algorithme utilisant une pile - 35

Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

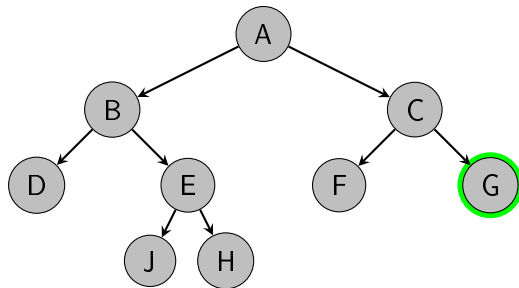


Algorithme utilisant une pile - 36

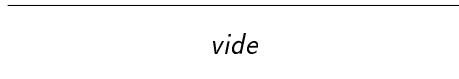
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 Dépiler pile → nœud
 - 5 Empiler les enfants
 - 6 TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

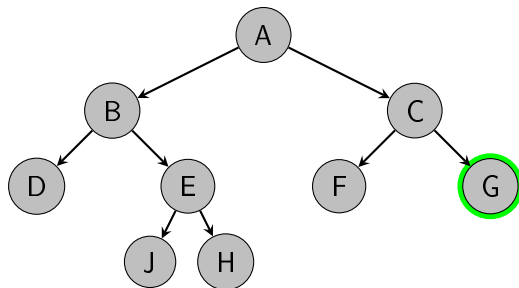


Algorithme utilisant une pile - 37

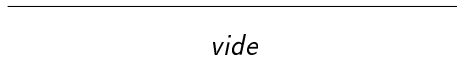
Entrées : arbre

```
1 Créer pile vide
2 Empiler racine de l'arbre
3 tant que Pile non vide
  faire
4   | Dépiler pile → nœud
5   | Empiler les enfants
6   | TRAITEMENT(nœud)
7 fin
```

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

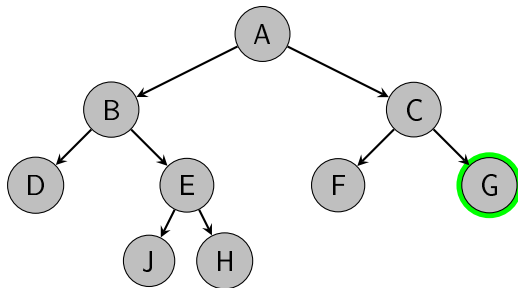


Algorithme utilisant une pile - 38

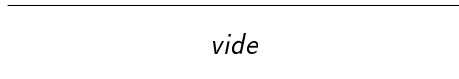
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | **TRAITEMENT**(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

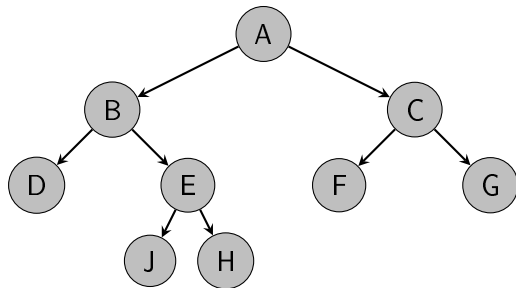


Algorithme utilisant une pile - 39

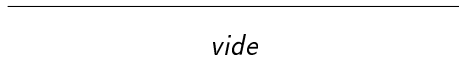
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

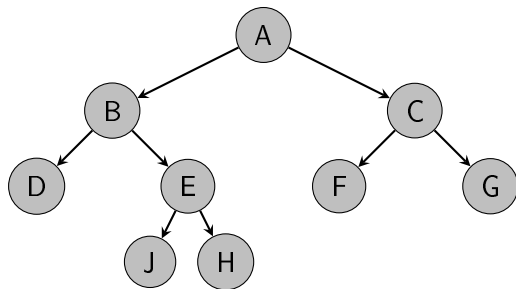


Algorithme utilisant une pile - Fini !

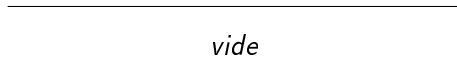
Entrées : arbre

- 1 Créer pile vide
- 2 Empiler racine de l'arbre
- 3 **tant que** *Pile non vide*
 faire
 - 4 | Dépiler pile → nœud
 - 5 | Empiler les enfants
 - 6 | TRAITEMENT(nœud)
- 7 **fin**

Remarque : L'empilement des enfants se fait en ordre inverse.



→ Pile



Traités

