

ALGORITMIQUE

4.5.6. Les piles

Une pile est une liste chaînée particulière où les insertions et les suppressions d'éléments se font toujours en tête de la liste que l'on appelle également sommet de la pile.

Le principe de fonctionnement d'une pile s'appelle LIFO (Last In First Out) : "le dernier arrivé est le premier servi"

Les piles s'utilisent lorsqu'on souhaite stocker des informations dans l'ordre de leur arrivée et les réutiliser dans l'ordre inverse de leur arrivée.

Par exemples,

- ✓ Une pile d'assiette où, la dernière assiette posée est la première prise ;
- ✓ Les appels d'une fonction récursive où, le dernier appel récursif (condition d'arrêt) est le premier à fournir une valeur ;
- ✓ Les pages Web visitées où, la dernière page visitée est la première proposée.

Les opérations sur les piles sont :

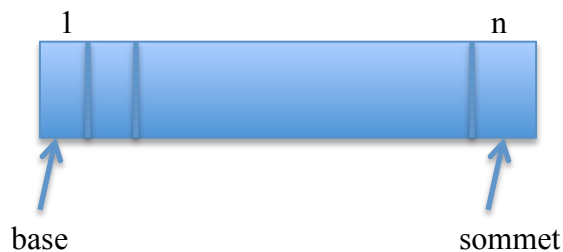
- ✓ Initialisation d'une pile ;
- ✓ Tester si une pile est vide ou pas ;
- ✓ Empiler (en Anglais, push) un élément dans une pile, c.-à-d., ajouter un élément à une pile ;
- ✓ Dépiler (en Anglais, pop) un élément d'une pile, c.-à-d., supprimer un élément d'une pile.

Les piles peuvent être représentés soit par des tableaux, soit par des listes chaînées.

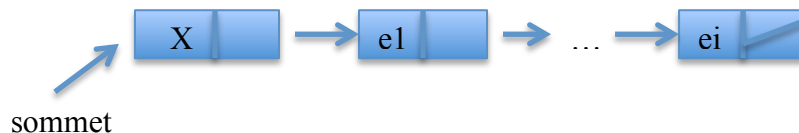
Dans le cas d'une représentation par tableau, il convient de noter que

- ✓ La taille de la pile est limitée par la taille du tableau ;

- ✓ Il faut limiter les éléments de la pile à un et un seul type ;
- ✓ Il faut utiliser une variable sommet qui indique à tout instant l'indice place de la tête de la pile (sommet de la pile) ;
- ✓ La base de la pile se trouve à l'indice 1 du tableau ;
- ✓ La pile est vide si et seulement si $\text{sommet} = 0$;
- ✓ Le dépilement peut avoir lieu si et seulement si la pile est non vide, c.-à-d., si et seulement si $\text{sommet} > 0$;
- ✓ L'empilement peut avoir lieu si et seulement si l'indice tête de la pile, à savoir, le sommet, est strictement inférieur à la taille du tableau.



Représentation d'une pile sous forme d'un tableau



Représentation d'une pile sous forme d'une liste chaînée

4.5.6.1. Algorithmes statiques

La représentation statique (sous forme d'un tableau) d'une pile se fait comme suit.

type pile = enregistrement

sommet : entier

T : tableau[1 ... n] de type t

Fin ;

Procédure initialiserpile (dr P : pile)
{On initialise la pile P, c.-à-d., on lui affecte la taille 0}

Début

P.sommet := 0

Fin ;

Fonction pilevide (d P : pile) : booléenne;
{On teste si la pile P est vide ou pas}

Début

Fin ;

Procédure empiler (dr P : pile ; d e :t)
{On ajoute à la pile P l'élément e}

Début

Fin ;

Procédure depiler (dr P : pile)
{On supprime de la pile P l'élément qui se trouve à l'indice sommet}

Début

Fin ;

Fonction sommetpile (d P : pile) : t;
 {On retourne le contenu du sommet de la pile}
 Début

Fin ;

4.5.6.2. Algorithmes dynamiques

La représentation dynamique d'une pile se fait sous forme d'une liste chaînée et ce, comme suit.

type pile = ^cellule

cellule = enregistrement

 valeur : t

 suivant : pile

Fin ;

Procédure initialiserpile (dr P : pile)
 {On initialise la pile P à nil}

Début

Fin ;

Fonction pilevide (d P : pile) : booléenne;
 {On teste si la pile P est vide ou pas}

Début

Fin ;

Procédure empiler (dr P : pile ; d e :t)

{On ajoute à la pile P l'élément e. Comme une pile est une liste chaînée et comme il est question d'ajouter e en tête de la pile, alors on fait appel à la procédure insertentete}

Début

Fin ;

Procédure depiler (dr P : pile)

{On supprime de la pile P l'élément qui se trouve à l'indice sommet}

var Q : pile

Début

Fin ;

Fonction sommetpile (d P : pile) : t;

{On retourne le contenu du sommet de la pile}

Début

Fin ;