

ALGORITMIQUE

4.5.7. Les files

Une file est une liste chaînée particulière où les insertions d'éléments se font toujours en queue de la liste (fin de la liste qui est l'entrée de la liste) et les suppressions d'éléments en tête de la liste (début de la liste qui est la sortie de liste).

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

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

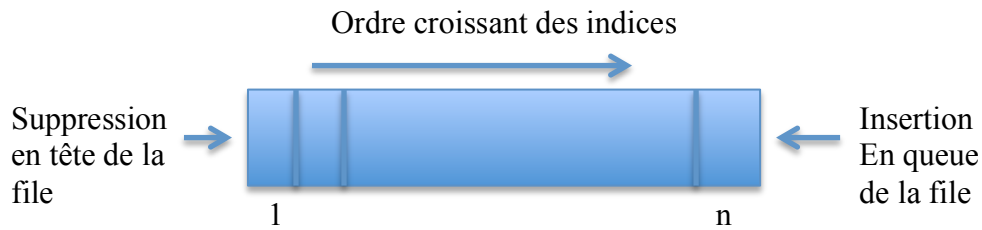
Par exemples,

- ✓ Une file d'attente où, la première personne arrivée est la première traitée;
- ✓ Une file de documents à imprimer, à traiter, etc.

Les opérations sur les files sont :

- ✓ Initialisation d'une file;
- ✓ Tester si une file est vide ou pas ;
- ✓ Enfiler un élément dans une file, c.-à-d., ajouter un élément en queue de la file ;
- ✓ Défiler un élément d'une file, c.-à-d., supprimer un élément de la tête de la file.

Les files peuvent être représentées soit par des tableaux, soit par des listes chaînées.



Représentation d'une file sous forme d'un tableau

Exemple

On souhaite insérer successivement les valeurs 4, 9, 5, 14 et 7 et ce, à partir d'une file vide que l'on représente sous forme d'un tableau pouvant contenir au plus 5 éléments.

La file résultante est la suivante.

| i | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|----|---|
| | 4 | 9 | 5 | 14 | 7 |

Maintenant, en supprimant la tête de la file nouvellement obtenue, on obtient la file suivante.

| i | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|----|---|
| | | 9 | 5 | 14 | 7 |

Comme vous pouvez le constater, la tête de la file n'est plus 1, mais 2.

Maintenant, en supprimant la tête de la file nouvellement obtenue, on obtient la file suivante.

| i | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|----|---|
| | | | 5 | 14 | 7 |

Comme vous pouvez le constater, la taille de la file est de 3 et il n'est plus possible d'ajouter des éléments à la file alors que le tableau peut aller jusqu'à 5 éléments.

Autrement dit, en représentant une file par un tableau, après des suppressions, on limite la capacité d'une file.

Pour remédier à ce problème, on va représenter une file par un tableau circulaire, c.-à-d., par un tableau où la case successeur de la dernière case du tableau est la première case du tableau.

En opérant de la sorte, on a tête = 3 et queue=1 (**queue est l'indice de la première case pouvant contenir le prochain élément de la file**).

4.5.7.1. Algorithmes statiques

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

type file = enregistrement

 queue : entier

 tête : entier

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

Fin ;

Procédure initialiserfile (dr F : file)

 {On initialise la file F}

 Début

 F.tête:= 1

 F.queue:= 1 {comme la tête est à 1, que l'indice de la première case pouvant contenir le prochain élément de la file est 1, et qu'il n'existe pas de case entre 1 et 1, alors la file est vide}

 Fin ;

Fonction filevide (d F : file) : booléenne;

 {On teste si la file F est vide ou pas}

 Début

Fin ;

Fonction filepleine (d F : file) : booléenne;

{On teste si la file F est pleine ou pas. Une file est pleine s'il ne reste qu'une case libre dans le tableau. Ainsi, une file ne peut contenir que $(n - 1)$ éléments}

Début

Fin ;

Procédure enfiler (dr F : file ; d e :t)

{On ajoute à file F l'élément e}

Début

Fin ;

Procédure defiler (dr F : file)

{On supprime le contenu de la tête de la file F. Le contenu supprimé de la file reste dans le tableau, mais est ignoré ; Donc cette procédure consiste simplement à incrémenter l'indice tête circulairement}

Début

Fin ;

4.5.7.2. Algorithmes dynamiques

La représentation dynamique (sous forme d'une liste chaînée) d'une file se fait comme suit.

type file = ^cellule

cellule = enregistrement

 valeur : t

 suivant : file

Fin ;

Procédure initialiserfile (dr F : file)

 {On initialise la file F à nil}

 Début

 F:= nil

 Fin ;

Fonction filevide (d F : file) : booléenne;

 {On teste si la file F est vide ou pas}

 Début

 filevide := (F = nil)

 Fin ;

Procédure enfiler (dr tete,queue : file ; d e :t)

 {On ajoute à la file l'élément e}

 Début

Fin ;

Procédure defiler (dr tete,queue : file ; r e :t)
{On supprime le contenu de la tête de la file}
var Q : file
Début

Fin ;