



CHAPITRE 6

Les Files d'attente (FIFO)

1. Définition
2. Principe, domaine d'application
3. Modèle
4. Implémentation
 - 4.1. Représentation statique
 - 4.2. Représentation dynamique

Chapitre 6 – Les files d'attente - FIFO

1



1. Définition

- Une file est une structure de données dynamique dans laquelle on insère des nouveaux éléments à la fin (queue) et on enlève des éléments au début (tête de file).
- L'application la plus classique est la file d'attente.
- La file sert beaucoup en simulation.
- Elle est aussi très utilisée aussi bien dans la vie courante que dans les systèmes informatiques. Par exemple, elle modélise la file d'attente:
 - des clients devant un guichet,
 - les travaux en attente d'exécution dans un système de traitement par lots,

Chapitre 6 – Les files d'attente - FIFO

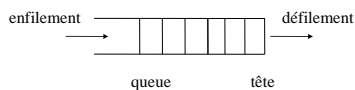
2



1. Définition

- ou encore les messages en attente dans un commutateur de réseau téléphonique.
- dans les programmes de traitement de transactions telle que les réservations de sièges d'avion ou de billets de théâtre.

Une file d'attente peut être définie comme une collection d'éléments dans laquelle tout nouveau élément est inséré à la fin et tout élément ne peut être supprimé que du début.



Chapitre 6 – Les files d'attente - FIFO

3

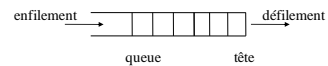


2. Principe, domaine d'application

C'est le principe "FIFO", abréviation de "First In, First Out" qui veut dire "premier entré premier servi".

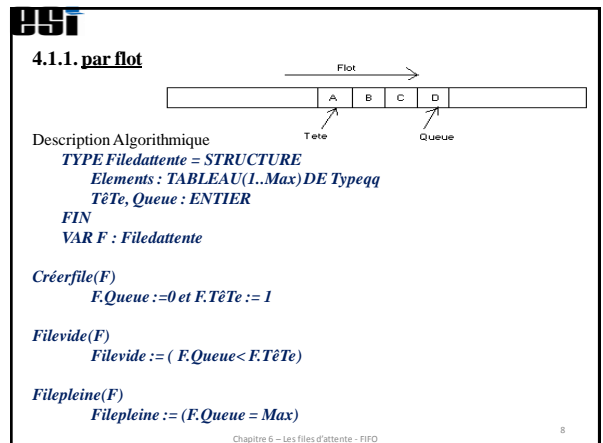
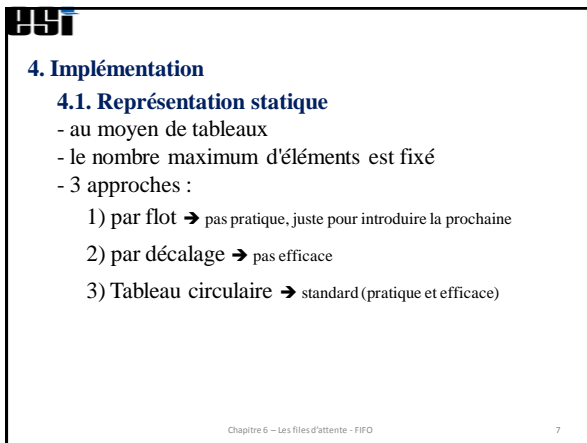
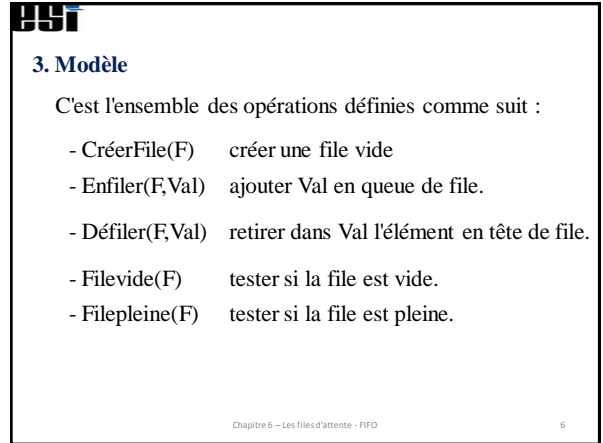
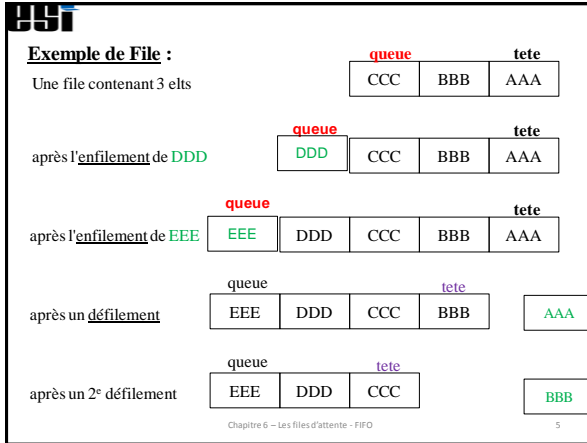
La file d'attente est très utilisée dans les systèmes d'exploitation des ordinateurs et surtout dans les problèmes de simulation.

Nous verrons aussi que la file d'attente peut être utilisée pour le parcours des arbres et pour résoudre tant d'autres problèmes.



Chapitre 6 – Les files d'attente - FIFO

4





4.1.1. par flot

Enfiler(F, X)

SI NON Filepleine(F)
 $F.Queue := F.Queue + 1$
 $F.Elements(F.Queue) := X$

SINON

"Overflow"

FSI

Defiler(F, X)

SI NON Filevide(F)
 $X := F.Elements(F.Tête)$
 $F.Tête := F.Tête + 1$

SINON

"Underflow"

FSI

Remarque 1 :

A tout moment le nombre d'éléments est $F.queue - F.tête + 1$

Remarque 2 :

La file n'est pas vide si $F.queue \geq F.tête$, donc la file est vide si non ($F.queue < F.tête$), c'est à dire $F.queue < F.tête$.

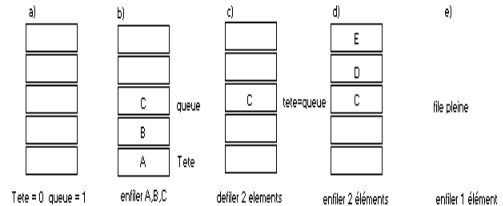
Chapitre 6 – Les files d'attente - FIFO

9



4.1.1. par flot

Exemple 1 :



C'est une solution inacceptable car on ne peut pas récupérer les emplacements X tels que $X < F.Tête$

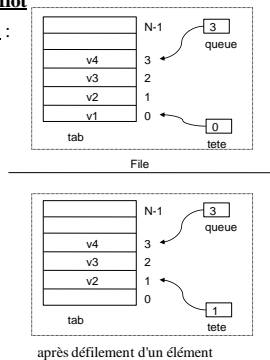
Chapitre 6 – Les files d'attente - FIFO

10



4.1.1. par flot

Exemple 2 :



L'inconvénient de cette solution est la perte des cases du tableau.

Chapitre 6 – Les files d'attente - FIFO

11



4.1.2. par décalage

A chaque défilement, on fait un décalage vers le bas. La tête n'est plus une caractéristique de la file d'attente puisqu'elle est toujours égale à 1.

Description Algorithmique

TYPE Filedattente = **STRUCTURE**
 Elements : **TABLEAU**(1..Max) de Type qq
 Queue : **ENTIER**
FIN

VAR F : Filedattente

Créerfile(F)
 $F.Queue := 0$

Filevide(F)
 $Filevide := (F.Queue = 0)$

Filepeine(F)
 $Filepeine := (F.Queue = Max)$

Chapitre 6 – Les files d'attente - FIFO

12



4.1.2. par décalage

Defiler(F, X)

SI NON Filevide(F)

X := F.Elements(1)

POUR I := 1 à F.Queue - 1

F.Elements(I) := F.Elements(I + 1)

FINPOUR

F.Queue := F.Queue - 1

SINON

" Filevide "

FSI

Enfiler(F, X)

SI NON Filepleine(F)

F.Queue := F.Queue + 1

F.Elements(F.Queue) := X

SINON

"Filepleine "

FSI

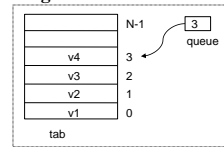
Chapitre 6 – Les files d'attente - FIFO

13

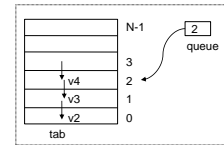


4.1.2. par décalage

Exemple :



File



après défilement d'un élément

Chapitre 6 – Les files d'attente - FIFO

14

L'inconvénient de cette solution est que pour chaque défilement, on fait un décalage



4.1.3. Tableau circulaire

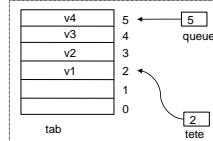
Revenons à la solution par flot et essayons d'utiliser le tableau de façon circulaire.

Considérons l'exemple suivant :

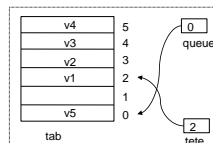
Comment initialiser la file?

$F.Queue < F.Tête$

impossible (contre exemple)



File contenant 4 éléments



après enfilement de v5

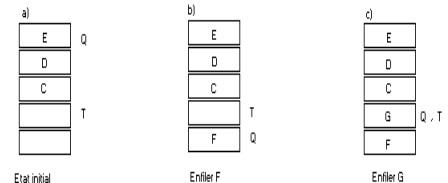
Chapitre 6 – Les files d'attente - FIFO

15



4.1.3. Tableau circulaire

Regardons ce qui se passe avec ces nouvelles considérations :



$F.tête = F.queue$

constitue aussi le cas file pleine

- même principe que l'approche par flots (même déclaration)
- les incréments se font modulo N → réutilisation des cases libérées.
- efficace : pas de boucle
- réutilise l'espace perdu par les défilements

Chapitre 6 – Les files d'attente - FIFO

16



4.1.3. Tableau circulaire

Solution :

F.Tête : pointe l'élément qui précède le premier.

F.Queue : pointe le dernier élément.

(F.Tête = F.Queue) constitue alors le cas file vide.

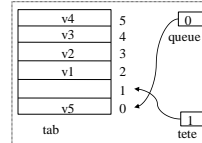
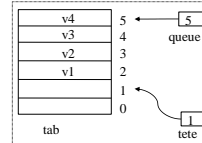
Initialisation (F.Tête = F.Queue := Max)

En d'autre terme :

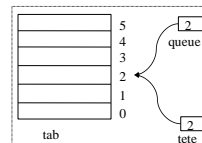
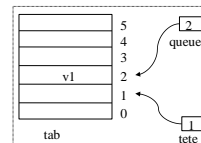
- par convention, l'élément d'indice tête sera sacrifié.
- le 1^{er} élément se trouve alors à l'indice (tête+1 mod N)



4.1.3. Tableau circulaire



FilePleine ssi :
 $tete = (queue + 1 \text{ mod } N)$



FileVide ssi :
 $tete = queue$



4.1.3. Tableau circulaire

Créerfile(F)

$F.Tête = F.Queue := Max$

Filevide(F)

$Filevide := (F.Tête = F.Queue)$

Filepleine(F)

$Filepleine := (F.Tête = (F.Queue \text{ Mod } Max + 1))$



4.1.3. Tableau circulaire

Enfiler(F, X)

SI NON Filepleine(F)

SI F.Queue = Max ()*

$F.Queue := 1$

SINON

$F.Queue := F.Queue + 1$

FSI

$F.éléments(F.Queue) := X$

SINON

"Overflow"

FSI

Defiler(F, X)

SI NON Filevide(F)

*SI F.Tête = Max (**)*

$F.Tête := 1$

SINON

$F.Tête := F.Tête + 1$

FSI

$X := F.éléments(F.Tête)$

SINON

"Underflow"

FSI

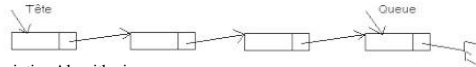
Remarque:

(*) $F.Queue = Max \Leftrightarrow F.Queue := F.Queue \text{ Mod } Max + 1$



4.2. Représentation dynamique

- au moyen des listes linéaires chaînées



Description Algorithmique

TYPE S = **STRUCTURE**

Info : Typeqg

Suiv : POINTEUR(S)

FIN

TYPE Filedattente = **STRUCTURE**

Tête, Queue : POINTEUR(S)

FIN

VAR F : Filedattente

Créerfile(F)

F.Tête := NIL

Filevide(F)

Filevide := (F.Tête = NIL)

Chapitre 6 – Les files d'attente - FIFO

21



4.2. Représentation dynamique

- au moyen des listes linéaires chaînées

Enfiler(F, X)

Allouer Q(S)

Aff_Val(Q, X)

Aff_Adr(Q, NIL)

SI NON Filevide(F)

Aff_Adr(F.Queue, Q)

SINON

F.Tête := Q

FSI

F.Queue := Q

Defiler(F, X)

SI NON Filevide(F)

Sauv := F.Tête

X := Valeur(F.Tête)

F.Tête := Suivant(F.Tête)

Liberer(Sauv)

SINON

"Underflow"

FSI

Chapitre 6 – Les files d'attente - FIFO

22