

# Chapitre 15 – Les files

---

Structures séquentielles.

Une file est un cas particulier de liste : on fait les ajouts à une extrémité et les suppressions se font à l'autre extrémité. FIFO : First In First Out.

# TAD File

---

Sorte : File

Utilise : Élément, Booléen

Opérations :

|                                      |                |
|--------------------------------------|----------------|
| créer-file-vide() $\rightarrow$ File | {constructeur} |
|--------------------------------------|----------------|

|                                           |                |
|-------------------------------------------|----------------|
| Ajouter(File, Élément) $\rightarrow$ File | {constructeur} |
|-------------------------------------------|----------------|

|                                  |                |
|----------------------------------|----------------|
| Retirer(File) $\rightarrow$ File | {constructeur} |
|----------------------------------|----------------|

|                                     |               |
|-------------------------------------|---------------|
| premier(File) $\rightarrow$ Élément | {observateur} |
|-------------------------------------|---------------|

|                                       |               |
|---------------------------------------|---------------|
| file-vide(File) $\rightarrow$ Booléen | {observateur} |
|---------------------------------------|---------------|

# Représentation contiguë (par tableau)

---

- ❑ tête : indice du premier élément de la file.
- ❑ queue : indice de la première case libre après la file.
- ❑ On fait progresser les indices tête et queue modulo la taille max du tableau.

```
Type file = Enregistrement
    t : tableau [1..n] d'élément
    tête, queue : entier
FinEnregistrement
Var f : file
```

L'implémentation du type file est triviale.

# File

---

☐ Dessin

# Représentation chaînée 1/2

---

Les suppressions et les insertions se font toujours en tête de liste.

```
Type    file = Enregistrement
           tête, queue : ^cellule
           FinEnregistrement
cellule = Enregistrement
           val : élément
           succ : ^cellule
           FinEnregistrement

Var f : file
```

# Représentation chaînée 2/2

Procédure ajouter (E e : élément, E/S f : file)

Var q : ^cellule

Début

q ← allouer(cellule)

q^.val ← e

q^.succ ← nil

f.queue^.succ ← q

f.queue ← q

Fin

Procédure retirer (E/S f : file)

Var q : ^cellule

Début

q ← f.tête

f.tête ← f.tête^.succ

recupérer(q)

Fin