

# Le TAD FILE et ses implémentations

7 février 2013

On veut implémenter de manière statique puis dynamique le type abstrait FILE permettant de gérer des files d'éléments. On utilisera ce TAD dans une application permettant la fusion de voies routières.

Pour l'implémentation statique, on utilisera une structure contenant un tableau et deux entiers (un pour le nombre d'éléments dans la file et l'autre pour indiquer la tête de file – on peut aussi ajouter un entier pour indiquer la queue de la file).

Pour l'implémentation dynamique, on utilisera une structure contenant deux pointeurs, un sur la tête de la file et l'autre sur la queue de la file.

La signature complète du type abstrait FILE est la suivante (voir dans le header une autre version possible de ce TAD – en commentaire pour chaque fonction) :

```
Sorte : FILE
Utilise : BOOLÉEN, NAT
Opérateurs_Constructeurs :
  CreerFile :  $\rightarrow$  FILE
  Enfiler : FILE  $\times$  ÉLÉMENT  $\rightarrow$  FILE
Opérateurs_Projecteurs :
  Appartient : FILE  $\times$  ÉLÉMENT  $\rightarrow$  BOOLÉEN
  EstVide : FILE  $\rightarrow$  BOOLÉEN
  Defiler : FILE  $\rightarrow$  FILE
  TailleFile : FILE  $\rightarrow$  NAT
  Position1 : FILE  $\times$  ÉLÉMENT  $\rightarrow$  NAT
  TêteFile : FILE  $\rightarrow$  ÉLÉMENT
  QueueFile : FILE  $\rightarrow$  ÉLÉMENT
  Concat : FILE  $\times$  FILE  $\rightarrow$  FILE
  Mixage : FILE  $\times$  FILE  $\rightarrow$  FILE

Pré-conditions :
  TêteFile( $f$ ) :  $\neg$ EstVide( $f$ )
  QueueFile( $f$ ) :  $\neg$ EstVide( $f$ )
Axiomes :
  Appartient(CreerFile(),  $x$ ) = faux
  Appartient(Enfiler( $f$ ,  $x$ ),  $y$ ) = ( $x = y$ )  $\vee$  Appartient( $f$ ,  $y$ )

  EstVide(CreerFile()) = vrai
  EstVide(Enfiler( $f$ ,  $x$ )) = faux

  TailleFile(CreerFile()) = 0
  TailleFile(Enfiler( $f$ ,  $x$ )) = TailleFile( $f$ ) + 1

  Position1(CreerFile(),  $x$ ) = 0
  Position1(Enfiler( $f$ ,  $x$ ),  $y$ ) =
    si Appartient ( $f$ ,  $y$ ) alors Position1( $f$ ,  $y$ )
    sinon si  $x = y$  alors TailleFile( $f$ ) + 1
    sinon 0

  Defiler(CreerFile()) = CreerFile()
  Defiler(Enfiler( $f$ ,  $x$ )) =
    si TailleFile( $f$ ) > 1 alors Enfiler(Defiler( $f$ ),  $x$ )
    sinon si TailleFile( $f$ ) = 1 alors Enfiler(CreerFile(),  $x$ )
    sinon CreerFile() /* c'est le cas où TailleFile ( $f$ ) = 0
```

$TêteFile(Enfiler(f,x)) =$   
     *si*  $TailleFile(f) \neq 0$  *alors*  $TêteFile(f)$   
     *sinon*  $x$

$QueueFile(Enfiler(f,x)) = x$

$Concat(CreerFile(), CreerFile()) = CreerFile()$   
 $Concat(CreerFile(), Enfiler(f,x)) = Enfiler(f,x)$   
 $Concat(Enfiler(f,x), CreerFile()) = Enfiler(f,x)$   
 $Concat(Enfiler(f_1,x_1), Enfiler(f_2,x_2)) =$   
      $Concat(Enfiler(Enfiler(f_1, x_1), TêteFile(Enfiler(f_2,x_2))),$   
      $Defiler(Enfiler(f_2,x_2)))$

$Mixage(CreerFile(), CreerFile()) = CreerFile()$   
 $Mixage(CreerFile(), Enfiler(f,x)) = Enfiler(f,x)$   
 $Mixage(Enfiler(f,x), CreerFile()) = Enfiler(f,x)$   
 $Mixage(Enfiler(f_1,x_1), Enfiler(f_2,x_2)) =$   
      $Concat(Enfiler(Enfiler(CreerFile(), TêteFile(Enfiler(f_1, x_1))), TêteFile(Enfiler(f_2,x_2))),$   
      $Mixage(Defiler(Enfiler(f_1,x_1)), Defiler(Enfiler(f_2,x_2))))$