

## TD5

**Exercice 1** Un algorithme de **triRapide** est basé sur le principe suivant :

1. partitionner le tableau  $T[g..d]$  en 2 sous-tableaux non vides  $T[g..p]$  et  $T[p+1..d]$  tels que tout élément de  $T[g..p]$  soit inférieur ou égal à tout élément de  $T[p+1..d]$ . A noter que l'indice  $p$  est calculé pendant le partitionnement ;
2. trier  $T[g..p]$  et  $T[p+1..d]$  par des appels récursifs ;
3. grâce au 1), pas besoin de recombinaison des tableaux  $T[g..p]$  et  $T[p+1..d]$ .

On donne l'algo de partitionnement (au départ **initPivot** renvoie simplement  $T[g]$ ) :

```
partitionner(T,g,d)
  pivot ← initPivot(t, g, d);
  i ← g - 1
  j ← d + 1
  fin ← faux
  while non fin do
    repeat
      j ← j - 1
    until T[j] ≤ pivot
    repeat
      i ← i + 1
    until T[i] ≥ pivot
    if i < j then
      permuter T[i] et T[j]
    else
      fin ← vrai
    end if
  end while
  retourner j
```

1. écrire la fonction partitionner à l'aide de l'algo. donné ;
2. écrire la fonction de **triRapide** selon le principe ci-dessus ;
3. appliquer la fonction **triRapide** au tableau  $t=\{3,17,9,15,2,7,10,9,8,12\}$ . Combien de permutations ont été réalisées ?
4. même question qu'en 3. avec **initPivot** qui renvoie maintenant  $t[(g+d)/2]$ .
5. modifier **initPivot** pour qu'il renvoie la médiane des valeurs de  $t$ . Même question qu'en 3. avec cette nouvelle fonction.

## Exercice 2

1. Ecrire une fonction qui prend en entrée deux tableaux triés d'entiers **t1** et **t2**, ayant  $n_1$  et  $n_2$  valeurs et renvoie comme résultats un troisième tableau trié qui est le résultat de la fusion de **t1** et **t2**, et la taille de ce troisième tableau. Le 3e tableau doit être de taille inférieure ou égale à une constante MAX.
2. Quel est :
  - le nombre de comparaisons réalisées si tous les éléments de  $t_1$  sont inférieurs aux éléments de  $t_2$  ?
  - le nombre de comparaisons réalisées si tous les éléments de  $t_1$  sont égaux aux éléments de  $t_2$  ?
  - le nombre de comparaisons réalisées si les 2 tableaux sont tels que :  
 $t_1[0] < t_2[0]$ ,  $t_1[1] > t_2[1]$ ,  $t_1[2] < t_2[2]$ , ... ?
  - l'ordre de grandeur de la complexité de votre fonction ?