

```

procédure partitionner(E/S t : Tableau[1..MAX] d'Entier; E debut,fin : Naturel;
S indicePivot : Naturel )
  Précondition(s)
    0 < debut ≤ fin ≤ MAX
  Déclaration
    i,j,pivot : Naturel
début
  pivot ← t[debut]
  i ← debut
  j ← fin
  tant que i≤j faire
    si t[i]≤pivot alors
      i i+1
    sinon
      si t[j]>pivot alors
        j ← j-1
      sinon
        echanger(t[i],t[j])
    finsi
  finsi
  fintantque
  indicePivot ← j
  echanger(t[debut],t[j])
fin

procédure quickSort(E/S t : Tableau[1..MAX] d'Entier; E debut,fin : Naturel )
  Précondition(s)
    0 < debut ≤ fin ≤ MAX
  Déclaration
    indicePivot : Naturel
début
  si debut < fin alors
    partitionner(t, debut, fin, indicePivot)
    quickSort(t, debut, indicePivot-1)
    quickSort(t, indicePivot+1, fin)
  finsi
fin

procédure trierParQuicksort(E/S t : Tableau[1..MAX] d'Entier; E nb : Naturel)
début
  quickSort(t,1,nb)
fin

```