

Série 2

EXERCICE 1.

Soit T un tableau de n entiers. T contient des 0, des 1 et des 2.
Ecrire un algorithme qui organise le tableau T de la façon suivante :
Tous les 0 sont regroupés en début de T , ils sont suivis d'une zone des 2 et à la fin de T la zone des 1.

EXERCICE 2.

Soit T un tableau à n entiers. On suppose que tous les éléments de T sont compris entre 1 et k . (k est un entier supérieur à 1)

Concevoir un algorithme qui :

- compte le nombre d'occurrences de chaque élément de T .
- trie le tableau T dans un tableau intermédiaire R en utilisant les nombres d'occurrences calculés en a).
- Evaluer la complexité de ce tri.

EXERCICE 3.

Ecrire, pour chacune des questions suivantes, un algorithme récursif pour calculer :

- Le pgcd de deux entiers positifs.
- La somme des chiffres d'un entier naturel n .
- Le nombre d'occurrence d'un entier x dans un tableau à n entiers.
- La moyenne $\frac{x_1+x_2+\dots+x_n}{n}$, $n \geq 1$.

EXERCICE 4.

Donner un algorithme récursif du tri par insertion d'un tableau à n éléments.

EXERCICE 5.

On considère l'algorithme récursif suivant :

```
Regle(g,d,h)
début
  m := (g + d) div 2 ;
  si h > 0 alors
    Marquer(m,h) ;
    Regle(g,m,h-1) ;
    Regle(m,d,h-1) ;
  fsi ;
fin
```

où **Marquer**(x,h) consiste à poser un trait vertical de hauteur h à la position x sur un axe horizontal d'origine 0.

- Faites dérouler l'exécution de **Regle**(0,8,3) en dressant l'arbre des appels récursifs.
- Calculer le nombre de marquages que nécessite l'exécution de **Regle**(0,2ⁿ, n). En déduire sa complexité.