

## EA4 – Examen du 27 Juin 2012 - durée : 2h30

La qualité de la rédaction sera prise en compte dans la notation. Justifiez toutes vos réponses et expliquez les fondements de vos algorithmes en français avant de les rédiger en pseudo-code compréhensible et commenté. Le barème est donné seulement à titre indicatif.

Les algorithmes donnés sans commentaires ou sans explications ne seront pas pris en compte par le correcteur. Documents non autorisés.

### Exercice 1 (4 points)

Simuler l'algorithme de tri rapide vu en cours pour trier le tableau : 6, 5, 7, 1, 4, 2, 8, 9, 3.

On détaillera chaque application de la procédure `Partition` en sachant qu'elle choisit comme pivot **toujours l'élément de trouvant au début de la partie du tableau à partitionner** (`pos_pivot` toujours égal à `d`). Vous n'avez pas le droit de choisir un autre élément comme pivot. On pourra représenter avec un graphe l'imbrication des appels récursifs effectués, ainsi que les tableaux qu'ils retournent.

Puisque plusieurs versions de la procédure `Partition` existent et pour uniformiser les copies, on appliquera *obligatoirement* la version ci-dessous.

```
Partition (T:tableau[0..n-1] d'entiers; d,f, pos_pivot: int) : int {
    échanger T[pos_pivot] et T[f];
    j := d;
    pour i de d à (f - 1) faire {
        si T[i] < T[f] alors {
            échanger T[i] et T[j];
            j := j + 1;
        }
    }
    échanger T[f] et T[j];
    renvoyer(j);
}
```

### Exercice 2 (10 points + 3 points bonus).

Soit la fonction suivante ayant un paramètre tableau d'entiers :

```
ViteOuLent (T : tableau [1, n] de int) : int {
    int d := 1;
    int f := n;
    int taille := f-d+1;
    tantque (taille >1) faire
        Si (taille est pair) alors d := (d+f)/2 + 1
            sinon {d++; f--};
        taille := f-d+1;
    ftq;
    retourne(T[d]);
}
```

1. Parmi tous les tableaux ayant une même longueur (fixée)  $n$ , quels tableaux réalisent le meilleur cas ? Quels tableaux réalisent le pire ?
2. Soit  $g(n)$  la fonction qui compte le nombre d'itérations de la boucle `tantque`.
  - Donnez une borne supérieure  $O(f_1(n))$  et une borne inférieure  $\Omega(f_2(n))$  pour  $g(n)$  ;
  - donnez les valeurs de  $n$  qui réalisent la borne  $f_1(n)$  et celles qui réalisent la borne  $f_2(n)$  ;
  - dites ce que renvoie la fonction dans ces deux cas. Autrement dit, calculez la valeur de  $d$  au moment du retour de  $T[d]$  pour les valeurs de  $n$  qui réalisent la borne  $f_1(n)$  et pour celles qui réalisent la borne  $f_2(n)$ .

Soit maintenant la fonction :

```
ViteOuLent2 (T : tableau [1, n] de int) : int {
  int d := 1;
  int f := n;
  int taille := f-d+1;
  tantque (taille >1) faire
    Si (T[d] est pair) alors d := (d+f)/2 + 1
      sinon {d++; f--};
    taille := f-d+1;
  ftq;
  retourne(T[d]);
}
```

(note : on a juste remplacé `taille` par `T[d]` dans la condition booléenne du `Si`)

3. Parmi tous les tableaux ayant une même longueur (fixée)  $n$ , lesquels réalisent le pire des cas ?
4. Que retourne la fonction dans ce cas et combien de fois la boucle `tantque` est-elle exécutée ?
5. Quelle est la probabilité de trouver un tableau qui réalise le pire de cas ?
6. Parmi tous les tableaux ayant une même longueur (fixée)  $n$ , lesquels réalisent le meilleur cas ?
7. Que retourne la fonction dans ce cas et combien de fois la boucle `tantque` est-elle exécutée ?
8. Quelle est la probabilité de trouver un tableau qui réalise le meilleur cas ?
9. *Questions bonus.* Soit  $n$  maintenant un entier quelconque et supposez de connaître sa décomposition en nombres premiers :

$$n = 2^{p_1} \cdot 3^{p_2} \cdot 5^{p_3} \cdot 7^{p_4} \cdot 11^{p_5} \dots$$

- Calculez la valeur exacte de  $g(n)$  où  $g$  est la fonction introduite à la question 2.
  - Calculez ce que la fonction `ViteOuLent` renvoie (c'est à dire, la valeur de  $d$  au moment du retour de  $T[d]$ ) pour un  $n$  quelconque.
- Toute l'information contenue dans la décomposition de  $n$  en nombres premiers est-elle nécessaire pour effectuer ces calculs ?

t

## Exercice 2 (6 points).

1. Insérez les éléments suivants dans un ABR initialement vide (1 point) :

93, 7, 84, 42, 12, 78, 66, 37, 29, 55

2. Supprimez ensuite l'élément 42. (1 point)
3. Insérez les mêmes éléments sur un AVL initialement vide et détaillait les opérations de rééquilibrage effectuées à chaque fois (2 points). Attention aux rotations doubles.
4. Supprimez ensuite l'élément 42 et enfin l'élément 93. (2 points)