Extraits de contrôles continus 2007/2008

.

Exercice 1 1. Soit h une fonction dont on cherche a estimer la complexité, peut on avoir :
- à la fois $h \in O(n^2)$ et $h \in O(n^3)$?
- à la fois $h \in \Omega(n^2)$ et $h \in \Omega(n^3)$?
- à la fois $h \in \theta(n^2)$ et $h \in \theta(n^3)$?

2. Est il vrai que $f \in \Omega(g)$ si et seulement si $g \in O(f)$
3. Donner une classe de complexité θ claire de $f(n) = n(2 + sin^2\ n)$

Exercice 2 On positionne les entiers suivant un triangle :

```
1
2 3
4 5 6
7 8 9 10
...
```

Donner une expression précise de la somme des entiers d'une ligne i

Exercice 3 La fonction f est définie par $f(n) = 3n^2 + 1$ lorsque $2^n < n^{log(n)}$ et $f(n) = n^3 + e^n$ sinon.

1. Quelle est la classe de complexite θ de f
2. Quelle est la classe de complexite θ de $(1/f)$
3. Montrer que lorsque $h \in \theta(g)$ alors $h \notin o(g)$

Exercice 4 Pour chacune des paires de fonctions suivantes $f(n)$ et $g(n)$, dire si f est $O(g)$, si g est $O(f)$ et si f est $\Theta(g)$:

a) $f(n) = ln(2n)$; $g(n) = ln(n^3)$
b) $f(n) = 2^{n/2}$; $g(n) = n^3$

Exercice 5 Donner la complexité de l'algorithme suivant en détaillant de manière exacte le nombre d'opérations élémentaires, puis en concluant :

```
FUNCTION amediter (Integer array: t)
BEGIN
  int x= 0;
  FOR 1 <= i <= longueur(t)
    FOR  i <= j <= longueur(t) - i
      IF t[j] < t[min] THEN
        x = x + 1
      ENDTHEN
      ELSE
        x=x-1
```

```
    ENDELSE
  ENDFOR
  ENDFOR
END
```

Exercice 6 Donner la complexité de l'algorithme suivant en détaillant le nombre d'opérations élémentaires, puis en concluant :

```
FUNCTION machin (Integer n)
BEGIN
  IF n = 1 THEN u = 1 ENDTHEN
  ELSE
     v = machin(n-1)
     IF pair(v) THEN u = 3 v + v/2 ENDTHEN
     ELSE u = (v -1)/ 2 ENDELSE
  ENDELSE
  RETURN u
END
```

où `impair(v)` retourne `vrai` si `v` est impair et `faux` sinon.

Exercice 7 Soit T un tableau trié d'entiers (positifs et négatifs) tous distincts. On sait que ce tableau est circulairement triée dans l'ordre croissant, c'est-à-dire qu'il existe un indice i tel que en commençant la lecture à $T[i]$ jusqu'a $T[n]$ et recommençant à $T[1]$ jusqu'à $T[i-1]$ on obtient une suite croissante. Par exemple, la suite $[15, 17, 20, 2, 3, 5, 8, 10]$ est circulairement triée, et l'indice de départ à considerer est $i = 4$.

Donner un algorithme linéaire qui permette de trouver l'indice i du plus petit élément du tableau.

Proposez un invariant et montrer que votre algorithme est correct.

On cherche à présent une solution logarithmique. Pour cela on remarque que quand on coupe en deux le tableau T, si on obtient une seule moitié monotone alors la solution recherchée se trouve dans l'autre moitié. Verifiez cette propriété sur le tableau donné en exemple.

Si on remplace 2 par 22, en coupant T au milieu on trouve 2 moitiés monotones. Comment conclut on dans ce cas la ?

Inspiré de la recherche dichotomique, donner à présent un algorithme qui trouve l'indice i du minimum en temps logarithmique.

Exercice 8 L'idée du tri fusion qui consiste à diviser le problème en 2 pour obtenir une meilleur complexité peut être explorée.

1. Ecrivez un algorithme de tri fusion qui sépare un tableau en 3 parties égales et fasse une fusion à 3. (3 points)
2. Calculez sa compléxité pour les cas où la taille n du tableau est une puissance de 3. (2 points)
3. La comparer au tri fusion classique. (1 point)