Examen partiel

Exercice 1 Donnez la complexité $\Theta(\cdot)$ de l'algorithme suivant en précisant les constantes c_1, c_2 et n_0 que vous avez utilisées.

```
public static void algo(int x, int n){
 for(int i=0;i<1000;++i)
    for(int j=i;j<x;++j)
      n += j;
}
```

Exercice 2 En utilisant **uniquement** la définition de log, montrer que $a^{\log_b n} = n^{\log_b a}$ pour tout $a,b,n \in \mathbb{N}$ positifs.

Exercice 3 Soit $T(n)$ une fonction (n entier) satifaisant:

$$T(n) = \begin{cases} 1 & \text{si } n = 1 \\ aT(\frac{n}{b}) + f(n) & \text{si } n > 1 \end{cases}$$

On suppose que $\log_b n = k \in \mathbb{N}$ avec b un entier ≥ 2. Montrer que pour tout $n \geq 2$

$$T(n) = n^{\log_b a} + \sum_{i=0}^{i=k-1} a^i f(\frac{n}{b^i})$$

Exercice 4 Soit $T(n)$ le temps d'exécution d'un algorithme en fonction de n satisfaisant $T(n) = 27T(n/3) + f(n)$. Donnez la valeur asymptotique $\Theta(\cdot)$ de $T(n)$ dans les tois cas suivants: (vous appliquerez le corollaire du théorème du cours en précisant la valeur de ε)

1) $f(n) = 3n^2 + 4\lg n$;
2) $f(n) = n^3 + 7n^2$;
3) $f(n) = \frac{n^4}{9}$; (vous donnerez d'abord une constante k et un n_0 tels que $f(3n) \leq kf(n)$ pour tout $n \geq n_0$).

Exercice 5 Soit x un réel positif.

1. Donnez une fonction $f(n)$, dépendante des valeurs de x, telle que

$$1 + x + x^2 + \ldots + x^n = \Theta(f(n))$$

2. Démontrez-le.

Exercice 6 Elaborer un algorithme en $O(n^{2.81})$ pour la multiplication de matrices $n \times n$. Vous décrirez l'algorithme et justifierez sa complexité.

Indication:

$$\begin{aligned} A_{11}B_{11} + A_{12}B_{21} &= P_1 + P_4 - P_5 + P_7 \\ A_{21}B_{12} + A_{22}B_{22} &= P_1 + P_3 - P_2 + P_6 \end{aligned}$$

avec

$$\begin{aligned} P_1 &= (A_{11} + A_{22})(B_{11} + B_{22}) \\ P_2 &= (A_{21} + A_{22})(B_{11}) \\ P_3 &= (A_{11})(B_{12} - B_{22}) \\ P_4 &= (A_{22})(B_{21} - B_{11}) \\ P_5 &= (A_{11} + A_{12})(B_{22}) \\ P_6 &= (A_{21} - A_{11})(B_{11} + B_{12}) \\ P_7 &= (A_{12} - A_{22})(B_{21} + B_{22}) \end{aligned}$$