Correction TD5

Exercice 1. Par définition, il existe k_1, c_1 et k_2, c_2 tels que $f(x) \leqslant k_1 g(x)$ pour $x > c_1$ et $g(x) \leqslant k_2 h(x)$ pour $x > c_2$. On pose alors $k_3 = k_1 k_2$ et $c_3 = \max(c_1, c_2)$, pour obtenir

$$\forall x > c_3, \quad f(x) \leqslant k_1 g(x) \leqslant k_1 k_2 h(x) = k_3 h(x)$$

Et donc $f \in O(h)$.

Exercice 2.
1. Le logarithme en base 2 peut être défini à partir du logarithme népérien (i.e en base e) de la façon suivante :

$$\log_2(n) = \frac{\ln(n)}{\ln(2)}$$

En effet, on a

$$2^{\frac{\ln(n)}{\ln(2)}} = e^{\frac{\ln(n)}{\ln(2)}\ln(2)} = e^{\ln(n)} = n$$

On a donc

$$L(n) = \log_2(n) + 1 = \frac{\ln(n)}{\ln(2)} + 1 \in O(\ln(n))$$

2. Les propriétés des logarithmes donnent :

$$\log_2(n^m) = m\log_2(n) = m(k-1)$$

Et donc $L(n^m) = m(k-1) + 1 \in O(m(k-1)) \subset O(mk)$.

3. La première propriété est juste dire que $\log_2$ est une fonction (strictement) croissante. On a alors

$$L(n!) = \log_2(n!) + 1 = \sum_{i=1}^{n} \log_2(i) + 1 \leqslant nk + 1 \in O(nk)$$

On note que c'est une majoration assez violente, la longueur de $n!$ va en réalité croître moins vite que la fonction $n\ln(n)$.

Exercice 3. Commençons par rappeler la formule du produit matriciel. On pose $M := (a_{i,j})_{i,j\in[\![1,n]\!]}$, $M' := (b_{i,j})_{i,j\in[\![1,n]\!]}$ et $MM' := (c_{i,j})_{i,j\in[\![1,n]\!]}$. On a par définition

$$\forall i,j \in [\![1,n]\!], \quad c_{i,j} = \sum_{k=1}^{n} a_{i,k} b_{k,j}$$

On rappelle que le produit de deux entiers de longueur bornée par un entier L prend au plus L^2 opérations élémentaires. De même, la somme de deux entiers de longueur bornée par un entier L prend au plus L opérations élémentaires. Donc le calcul de $c_{i,j}$ prends

- n produits $a_{i,k}b_{k,j}$ prenant chacun au plus $L(m)$ opérations élémentaires.
- La somme des n entiers obtenus, chacun de longueur au plus $L(m)^2$. Donc au plus $nL(m)^2$ opérations.

Au total, le calcul de $c_{i,j}$ prends $nL(m) + nL(m)^2 \in O(n\ln(m) + n\ln(m)^2)$ opérations. Il y a n^2 tels coefficients à calculer, d'où un total d'opérations en $O(n^3(\ln(m) + \ln(m)^2))$

Exercice 4.
L'exponentiation naïve requiert m multiplications de n avec lui même, d'où une complexité en $O(m)$. Pour l'exponentiation rapide, on calcule n^{2^i} pour $i \leqslant L(m)$, soit $L(m)$ multiplications. On fait ensuite le produit des n^{2^i}, soit au plus $L(m)$ multiplications, soit une complexité totale en $O(2L(m)) \subset O(2\ln(m))$ comme annoncé.

Exercice 5.
1. Comme on a $k \leqslant n$, le calcul de k^2 prends au plus $L(n)^2$ opérations élémentaires. La somme des n entiers k^2 se fait en n sommes prenant chacune au plus $L(k) \leqslant L(n)^2$ opérations. Le nombre total d'opérations est alors donné par $2nL^2(n) \in O(n\ln^2(n))$

2. On a trois entiers $n, n+1, 2n+1$, avec $L(n+1), L(2n+1) \in O(L(n))$. Le produit de n et $n+1$ se fait en $O(L^2(n))$ opérations et donne un entier de longueur en $O(L(n))$. Le produit de $n(n+1)$ avec $2n+1$ se fait également en $O(L^2(n))$ opérations. Bien-sûr le produit par $\frac{1}{6}$ ne change pas le comportement asymptotique (c'est un nombre constant d'opérations). D'où le résultat total en $O(L^2(n)) \subset O(\ln^2(n))$.

Exercice 6.
1. On pose $M = (a_{i,j})_{i,j\in[\![1,n]\!]}$. On commence par traiter la première colonne de M :

1. Il faut d'abord trouver un coefficient $a_{i,1}$ non nul, ce qui prend au plus n tests de non nullité (si tous les $a_{i,1}$ sont nuls, le traitement de la première colonne s'arrête là, en n opérations).
2. Une fois trouvé un $a_{i_0,1}$ non nul, on échange les lignes L_1 et L_i, ce qui prends $2n$ opérations d'échange.
3. Maintenant, on peut supposer $a_{1,1}$ non nul. On effectue alors, pour tout $i \in [\![2,n]\!]$
   - Calculer $\frac{a_{i,1}}{a_{1,1}}$, soit une opération.
   - Remplacer, pour tout $j \in [\![1,n]\!]$, $a_{i,j}$ par $a_{i,j} - \frac{a_{i,1}}{a_{1,1}} a_{1,j}$, soit un total de $2n$ opérations.

Donc $n(2n+1) = 2n^2 + n \in O(n^2)$ opérations pour traiter la première colonne. Le traitement des colonnes suivante revient à opérer l'algorithme de Gauss sur la sous-matrice $(a_{i,j})_{i,j\in[\![2,n]\!]}$, d'où $T(n) \in O(T(n-1) + n^2)$.

2. Par une récurrence immédiate, on obtient $T(n) \in O\left(\sum_{i=0}^{n} i^2\right) = O\left(n(n+1)(2n+1)\right) \subset O(n^3)$

Exercice 7.
1. Là encore, on va procéder par récurrence. On pose $M = (a_{i,j})_{i,j\in[\![1,n]\!]}$, et $M_{i,j}$ la matrice M privée de sa i-ème ligne et de sa j-ème colonne. Développer le déterminant par rapport à la première colonne nous donne

$$\det(M) = \sum_{i=1}^{n} (-1)^i a_{i,1} \det(M_{i,j})$$

On doit donc calculer n déterminants de taille $n-1 \times n-1$, les $\det(M_{i,j})$: $nT(n-1)$ opérations. Ensuite, il faut faire les produits $(-1)^i a_{i,1} \det(M_{i,j})$, soit n produits, et enfin n additions. Au total on a $nT(n-1) + 2n = n(T(n-1)+2)$ opérations. On a alors

$$\frac{T(n)}{n!} = \frac{n(T(n-1)+2)}{n!} = \frac{T(n-1)}{(n-1)!} + \frac{2}{n!} = \frac{T(1)}{1} + \sum_{i=1}^{n} \frac{2}{i!}$$

Cette dernière somme converge, donc $\frac{T(n)}{n!}$ est borné et $T(n) \in O(n!)$.

2. En appliquant l'algorithme du pivot de Gauss sur M, on obtient une matrice diagonale, et chaque transformation ne modifie pas le déterminant de la matrice : au pire, on échange deux lignes et on multiplie le déterminant par -1, soit au plus n nouvelles opérations, ce qui est non significatif par rapport au n^3 opérations du pivot de Gauss. Il suffit ensuite de faire le produit des coefficients diagonaux de M pour obtenir son déterminant. Au total la complexité de cette méthode est la même que celle du pivot de Gauss : $O(n^3)$.