Structures de Données - L2, S4

TD 4 - Éléments de Complexité

Exercice 4.1 *Complexité de boucles*
Calculer la complexité en temps pour chacun des algorithmes suivants :

1. ```
   s = 0;
   for (i =0; i <=n; i++)
      s +=i
   ```
2. ```
   s = 0;
   for (i =0; i <=n; i++)
       for (j=i; j<=n; j++)
           s +=i
   ```
3. ```
   for (k=2; k<=n; k++) {
       x= a[k]
       j=k-1
       while (j!=0) {
            if (x<a[j])
               j=j-1
            else
               j=0
       }
   }
   ```
4. ```
   c=n
   while (c > 1)
      c = c/5
   ```

Exercice 4.2 *Notations O*

1. Montrez que $(n+a)^b = O(n^b)$ pour des constantes a et b.
2. Est-ce que la phrase suivante a un sens "Le temps d'exécution de l'algorithme est au moins $O(n^2)$" ? Et pourquoi ?
3. Pourquoi $2^{n+1} = O(2^n)$ et $2^{2n} \neq O(2^n)$?

Exercice 4.3 *Codage des relations binaires*
Rappelons qu'une relation binaire R sur V est un sous-ensemble de $V \times V$. On a deux possibilités pour représenter une relation binaire R : soit par une matrice A, appelée *matrice d'adjacence*, où $A[x,y] = 1$ si $(x,y) \in R$, ou bien par un tableau T indexé par V, appelé *liste d'adjacence*, et pour tout $x \in V$, $T[x]$ est une liste représentant $\{y \in V \mid (x,y) \in R\}$ [1].

1. en fait on suppose qu'il existe une fonction bijective $b : V \to [|V|]$ et $T[i]$ est la liste représentant $\{y \in V \mid (b^{-1}(i), y) \in R\}$.

1. Proposez un encodage concis pour la matrice d'adjacence et la liste d'adjacence d'une relation binaire.
2. Montrez que l'on peut passer de l'une à l'autre en temps polynomial.

Exercice 4.4* *Un peu de robustesse pour P*

1. Montrez qu'un algorithme qui fait appel à un nombre constant d'algorithmes polynomiaux est également polynomial.
2. Montrez qu'un algorithme qui fait appel à un nombre polynomial d'algorithmes polynomiaux peut avoir une complexité en temps exponentiel.

Exercice 4.5* *Fermeture de P*

Soit $\mathcal{F}$ l'ensemble des fonctions $f : \{0,1\}^* \to \{0,1\}$. Une telle fonction f peut être par exemple une fonction qui teste si deux chaînes de caractères sont égales. Pour $f \in \mathcal{F}$, posons $L_f := \{x \in \{0,1\}^* \mid f(x) = 1\}$.

1. Est-ce que dire que $f \in \mathcal{F}$ peut être calculé en temps polynomial est équivalent à dire qu'il existe un algorithme reconnaissant les éléments de L_f en temps polynomial ?
2. Soient $f, g \in \mathcal{F}$ et supposons pour chacune d'elle il existe un algorithme en temps polynomial la calculant. Montrez que les éléments des ensembles suivants peuvent être également reconnus en temps polynomial : $L_f \cup L_g$, $L_f \cap L_g$, $L_f \times L_g$, $L_{f'} := \{x \in \{0,1\}^* \mid x \notin L_f\}$.