L3-INFO Algorithmique et Modélisation

Quick – 24 février 2023 – durée 1 h

Sont interdits : les documents, les ordinateurs, les téléphones (incluant smartphone, tablettes,... tout ce qui contient un dispositif électronique).
Seuls les dictionnaires papier pour les personnes de langue étrangère sont autorisés.
Il sera tenu compte de la qualité de la rédaction et de la clarté de la présentation (2 pts).
Une phrase courte et claire vaut mieux qu'une expression longue confuse avec des ratures
En cas d'incompréhension du sujet, préciser les hypothèses de travail que vous faites et continuer.
Le barème **indicatif** : Exercice 1 : 6 pts (5 à 10 mn); Exercice 2 : 3 pts (10 à 20 mn), Exercice 3 : 9 pts (20 à 30 mn) Relecture ($\sim 5mn$)
Les 3 exercices sont indépendants.

I : Le tri rapide

Dans l'analyse en moyenne du tri rapide, vue en travaux dirigés, on a établi que le coût moyen M_n en nombre de comparaisons entre éléments pour un tableau de taille n vérifiait une équation de récurrence

$$M_n = (n-1) + \sum_{i=0}^{n-1} (M_i + M_{n-1-i}) \times \frac{1}{n} \text{ avec } M_0 = 0.$$

Un étudiant de licence 2 vous propose un programme récursif, écrit en Python, pour calculer M_n :

```
def analyse_moyenne_QS(n):
    if n == 0:
        return 0
    else:
        S = n-1
        for i in range(n): # i prend les valeurs 0,1,... , n-1
            S = S + (analyse_moyenne_QS(i) + analyse_moyenne_QS(n-1-i))/n
        return S
```

I.1. Ce programme est-il correct ?

I.2. Est-il efficace (on calculera un ordre de grandeur de sa complexité en nombre d'appels récursifs) ?

I.3. Proposer un commentaire expliquant la difficulté rencontrée.

I.4. Proposer un algorithme efficace de calcul de M_n et évaluer sa complexité (on n'utilisera pas de formule explicite de la valeur de la complexité).

II : Contrôle d'intégrité

Afin de s'assurer qu'un fichier a été correctement téléchargé (pas d'altérations lors de la transmission) on utilise la fonction `md5` qui produit une signature associée au fichier. On suppose qu'il n'y a pas de question de sécurité et que l'on souhaite juste s'assurer de l'intégrité des données.

MD 5 (extrait de l'article wikipedia `https://fr.wikipedia.org/wiki/MD5`)

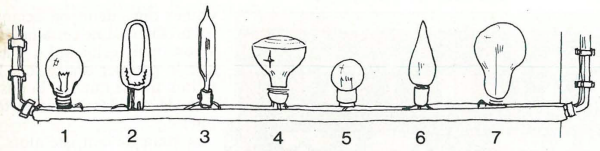
> *Le MD5, pour Message Digest 5, est une fonction de hachage cryptographique qui permet d'obtenir l'empreinte numérique d'un fichier (on parle souvent de message). Il a été inventé par Ronald Rivest en 1991.*
>
> *Si l'algorithme MD5 présente un intérêt historique important, il est aujourd'hui considéré comme dépassé et absolument impropre à toute utilisation en cryptographie ou en sécurité. Il est toutefois encore utilisé pour vérifier l'intégrité d'un fichier après un téléchargement.*
>
> *Historique : MD5 (Message Digest 5) est une fonction de hachage cryptographique qui calcule, à partir d'un fichier numérique, son empreinte numérique (en l'occurrence une séquence de 128 bits ou 32 caractères en notation hexadécimale) avec une probabilité très forte que deux fichiers différents donnent deux empreintes différentes.*

II.1. Donner un ordre de grandeur de la probabilité que la signature de 2 fichiers soient identiques, on justifiera les hypothèses probabilistes retenues.

II.2. Commenter le résultat.

III : Faire la lumière (extrait de Jeux et Stratégie mars 1990)

CIRCUITS LOGIQUES



Pour allumer ces sept lampes, on dispose de 7 interrupteurs. Chacun d'eux change l'état de certaines lampes qui lui sont spécifiques (si la lampe est éteinte, elle s'allume ; si la lampe est allumée, elle s'éteint).

L'interrupteur A change l'état des lampes 1, 3 et 5
L'interrupteur B change l'état des lampes 2 et 7
L'interrupteur C change l'état des lampes 3, 4, 6 et 7
L'interrupteur D change l'état des lampes 1, 4, 5 et 7
L'interrupteur E change l'état des lampes 1 et 6
L'interrupteur F change l'état des lampes 2 et 3
L'interrupteur G change l'état des lampes 2, 4 et 6

Les sept lampes sont éteintes. Sur quels interrupteurs faut- il agir pour que les sept lampes soient allumées simultanément ?

III.1. Résoudre, si c'est possible, le problème posé (ne pas passer trop de temps 5mn max).

III.2. Modéliser le problème dans le cas général d'un ensemble de n lampes et m interrupteurs, à chaque interrupteur on associe un ensemble de lampes.

III.3. En vous inspirant de l'algorithme d'énumération des parties proposer un algorithme de résolution de ce problème.

III.4. Dessiner un arbre des appels associé à l'algorithme sur un exemple judicieusement choisi. Peut-on améliorer l'algorithme en élaguant l'arbre, si oui comment ?

III.5. Calculer le coût de cet algorithme.

III.6. (bonus) Mettre ce problème sous la forme d'un système linéaire à résoudre modulo 2. Quel est dans ce cas la complexité de l'algorithme ?