Algorithmique — M1

Examen du 18 juin 2009

Université Paris Diderot

Documents autorisés : Deux feuilles de papier format A4 Durée : 3h

On applique les cours

Exercice 1 – *Multiplication*

En utilisant l'algorithme de Karatsuba-Offman multipliez 2 nombres binaires : 10011011 et 10111010.

Exercice 2 – *Récurrence*

Étant donné que

$$T(n) = 8T(\lfloor n/2 \rfloor) + n^3 + 3$$

trouvez le comportement asymptotique de $T(n)$. Justifiez votre réponse.

Exercice 3 – *Les mariages*

5 enseignants (A,B,C,D,E) doivent enseigner 5 cours (I,J,K,L,M). Il faut avoir un enseignant pour chaque cours et un cours pour chaque enseignant. Les compétences des enseignants sont représentées par un tableau suivant :

	I	J	K	L	M
A	✓		✓		✓
B		✓	✓	✓	
C	✓	✓			
D				✓	✓
E		✓		✓	

Pour affecter les cours aux enseignants :

1. choisissez un algorithme de cours ;

2. appliquez l'algorithme (écrivez/dessinez toutes ses étapes) ;

3. donnez le résultat final : quel enseignant fait quel cours.

On invente des algorithmes

Exercice 4 – *Méthode imposée*

On cherche le maximum d'un tableau de B de n éléments.

1. Écrivez un algorithme de type diviser-pour-régner qui résout ce problème.
2. Analysez sa complexité.

Exercice 5 – *Un mauvais algorithme*

Étant donné un graphe non-orienté $G = (V, E)$ (de N sommets) représenté par une matrice d'adjacence $M[i, j]$ (de taille $N \times N$) et un entier M on cherche une clique de C sommets. Dans cet exercice il faut trouver un algorithme retour-arrière qui résout ce problème. On va appeler une solution partielle un tableau d'entiers $[i_1, \ldots, i_k]$ dans l'ordre croissant tel que les sommets $[v_{i_1}, \ldots, v_{i_k}]$ forment une clique.

1. Écrivez une fonction booléenne $\mathtt{test}(B, k, M, N)$ qui teste est-ce que le tableau $B[k]$ est une solution partielle pour un graphe représenté par un tableau (matrice d'adjacence) $M[N, N]$.
2. Comment trouver une solution partielle de taille 0 ? Comment à partir d'une solution partielle de taille k passer à ses extensions de taille $k + 1$? Comment dire est-ce qu'on a déjà trouvé la clique de taille C ?
3. Écrivez un algorithme retour-arrière de recherche de clique.
4. Estimez la complexité de votre algorithme.

Exercice 6 – *La somme*

Dans un tableau d'entiers B on cherche un sous-ensemble de B qui a la somme S donnée. Par exemple pour $B = [3, 4, 5, 11]$ et $S = 16$ il y a une solution $16 = 11 + 5$. Pour le même B et pour $S = 10$ il n'y a pas de solution.

Le problème algorithmique à résoudre dans cet exercice est le suivant : étant donné $B = p_1, p_2, \ldots, p_m$ (trié) et S répondre (VRAI ou FAUX) est-ce que B a une partie de somme S. On utilisera la programmation dynamique pour concevoir un algorithme qui résolve ce problème.

1. Soit $P(i, j)$ un booléen qui est vrai ssi on peut obtenir j comme somme d'une partie de i premiers éléments de tableau : $p_1, \ldots, p_i$ Écrivez les équations de récurrence pour cette fonction sans oublier les cas de base.
2. Écrivez un algorithme efficace (récursif avec "marquage" ou itératif) pour calculer P.
3. Analysez la complexité de votre algorithme.
4. Appliquez votre algorithme à l'exemple ci-dessus ($B = [3, 4, 5, 11]$ et $S = 9$).