Examen MIM1 – Algorithmique

Yves Robert

Mardi 16 Janvier 2001, 9h–12h

Les notes de cours sont autorisées. Les trois parties sont indépendantes.

1 Algorithmique (facile)

1.1 Tas d-aire

Soit $d \geq 3$ un entier. Un tas d-aire est l'équivalent d'un tas binaire, mais chaque sommet à d fils au lieu de 2.

1. Expliquer soigneusement quelle structure de données choisir pour implanter un tas d-aire, et comment calculer les relations entre père et fils.
2. Quel est le coût d'une insertion et de la suppression de la racine dans un tas d-aire ?
3. Conclusion ?

1.2 Les i plus grands éléments triés

Etant donnés un ensemble de n nombres, on veut obtenir les i plus grands d'entre eux triés. Pour chacune des trois méthodes suivantes, donner la structure de données utilisée, et la complexité en fonction de n et de i. Quelle est la meilleure méthode ?

(a) Trier les n nombres et renvoyer les i plus grands

(b) Construire un tas avec les n nombres et extraire la racine i fois

(c) Trouver le i-ème élément, partitionner et trier les i plus grands

2 NP-complétude (moyen)

Montrer que les variantes suivantes de problèmes NP-complets classiques sont toujours NP-complètes :

Vertex Cover avec degré pair Soient $G = (V, E)$ un graphe dont tous les sommets sont de degré pair, et $k \geq 1$ un entier. Existe-t-il un ensemble de sommets de G couvrant toutes les arêtes et de taille inférieure ou égale à k ?

2-Partition avec même cardinal Soient $n = 2p$ un entier pair, et n entiers strictement positifs $a_1, a_2, \ldots, a_n$. Existe-t-il une partition de $\{1, 2, \ldots n\}$ en deux ensembles I et I' de même cardinal p et tels que $\sum_{i \in I} a_i = \sum_{i \in I'} a_i$?

Clique pour graphe régulier Soient $G = (V, E)$ un graphe dont tous les sommets sont de même degré, et $k \geq 1$ un entier. Existe-t-il une clique de taille supérieure ou égale à k ?

3 Approximation (difficile)

On s'intéresse au problème du sac-à-dos :
Instance : Un ensemble fini X d'objets ; pour chaque objet $x_i \in X$, une valeur $p_i \in \mathbb{N}$ et une taille $a_i \in \mathbb{N}$. Une capacité $B \in \mathbb{N}$.
Solution : Un sous-ensemble Y de X tel que $\sum_{x_i \in Y} a_i \leq B$
Mesure : La valeur totale des objets choisis, i.e. $\sum_{x_i \in Y} p_i$.
On notera $m^*(x)$ la mesure optimale (maximale).

3.1 Glouton

Dans l'algorithme glouton, on trie les éléments par rapport p_i/a_i décroissant, et on choisit chaque élément examiné dans cet ordre s'il y a la place pour lui. Soit $m_g(x)$ la mesure de l'algorithme glouton.

1. Soit K un entier arbitrairement grand. Construire une instance x telle que $m^*(x)/m_g(x) > K$.
2. Soit p_{max} la valeur maximale d'un objet. Montrer que $m^*(x)/\max\{m_g(x), p_{max}\} < 2$.
   Indication : soit j l'indice du premier élément que l'algorithme glouton ne prend pas ; montrer que $m^*(x) \leq \sum_{i=1}^{j} p_i$.

3.2 Programmation dynamique

1. Pour $1 \leq k \leq n$ et $0 \leq p \leq \sum_{i=1}^{n} p_i$, on cherche, parmi les sous-ensembles de $\{x_1, x_2, \ldots, x_k\}$ de valeur totale égale à p et de taille majorée par B, un qui soit de taille minimale. On note $M^*(k,p)$ une solution optimale de ce problème et $S^*(k,p)$ la taille correspondante.
   Expliquer comment calculer les $M^*(k,p)$ par programmation dynamique. Quelle est la complexité de la résolution du problème du sac-à-dos ?
2. Pout tout rationnel $r > 1$, on considère le schéma d'approximation suivant : soit p_{max} la valeur maximale d'un objet et $t = \lfloor \log(\frac{r-1}{r}\frac{p_{max}}{n}) \rfloor$. On résout par programmation dynamique comme plus haut une instance modifiée du problème original : les tailles des n objets sont toujours a_i mais les valeurs sont $p'_i = \lfloor$