Sac à dos

Épreuve pratique d'algorithmique et de programmation
Concours commun des écoles normales supérieures

Durée de l'épreuve: 3 heures 30 minutes

Juillet 2011

ATTENTION !

N'oubliez en aucun cas de recopier votre u_0
à l'emplacement prévu sur votre fiche réponse

Important.

Sur votre table est indiqué un numéro u_0 qui servira d'entrée à vos programmes. Les réponses attendues sont généralement courtes et doivent être données sur la fiche réponse fournie à la fin du sujet. À la fin du sujet, vous trouverez en fait deux fiches réponses. La première est un exemple des réponses attendues pour un $\widetilde{u_0}$ particulier (précisé sur cette même fiche et que nous notons avec un tilde pour éviter toute confusion!). Cette fiche est destinée à vous aider à vérifier le résultat de vos programmes en les testant avec $\widetilde{u_0}$ au lieu de u_0. Vous indiquerez vos réponses (correspondant à votre u_0) sur la seconde et vous la remettrez à l'examinateur à la fin de l'épreuve.

En ce qui concerne la partie orale de l'examen, lorsque la description d'un algorithme est demandée, vous devez présenter son fonctionnement de façon schématique, courte et précise. Vous ne devez en aucun cas recopier le code de vos procédures!
Quand on demande la complexité en temps ou en mémoire d'un algorithme en fonction d'un paramètre n, on demande l'ordre de grandeur en fonction du paramètre, par exemple: $O(n^2)$, $O(n \log n)$,...
Il est recommandé de commencer par lancer vos programmes sur de petites valeurs des paramètres et de ***tester vos programmes sur des petits exemples que vous aurez résolus préalablement à la main ou bien à l'aide de la fiche réponse type fournie en annexe***. Enfin, il est recommandé de lire l'intégralité du sujet avant de commencer afin d'effectuer les bons choix de structures de données dès le début.

1 Introduction

Dans ce sujet, on s'intéresse à une classe de problèmes d'optimisation connus sous le nom général de problème du **sac à dos**. Informellement, ce problème est défini de la manière suivante : durant un cambriolage un voleur possède un sac dont la capacité (en poids par exemple) est limitée. Il se trouve face à un ensemble d'objets qu'il peut dérober. Chacun de ces objets est caractérisé par sa valeur et son poids. Le voleur souhaite optimiser la valeur totale des objets qu'il va dérober tout en ne dépassant pas le poids maximal supporté par son sac.
Ce problème est une abstraction pour un grand nombre d'autres problèmes d'optimisation. Il est aussi utilisé en cryptographie comme une base pour différents schémas de chiffrement. Il faut cependant noter que la plupart de ces schémas de chiffrement ne sont plus actuellement considérés comme sûrs.

1.1 Générateur d'entiers pseudo-aléatoires

Considérons $(u_k)_{k\geq 0}$ la suite d'entiers définie par :

$$u_k = \begin{cases} \text{votre } u_0 \quad (\textit{à reporter sur votre fiche réponse}) & \text{si } k = 0 \\ 15091 \times u_{k-1} \mod 64007 & \text{k>0.} \end{cases}$$

Question 1 *Que valent :*
a) u_{30} **b)** u_{300} **c)** u_{3000}.

1.2 Sac à dos

Comme évoqué dans l'introduction, le problème du sac à dos est caractérisé par une liste d'objets possédant chacun un poids et une valeur. On définit par SD_{n,v_B,p_B} (avec $p_B \geq 7$) le problème du sac à dos pour n objets dont les valeurs et poids sont donnés par la suite de couples $(v_i, p_i)_{0\leq i\leq(n-1)}$ tels que d'une part ces couples respectent la propriété suivante :

$$\forall i \quad 0 \leq i < n \quad \begin{cases} 1 \leq v_i \leq v_B \\ 7 \leq p_i \leq p_B \end{cases}$$

Et d'autre part, leurs valeurs respectives sont dérivées de la suite (u_k) de la manière suivante :

$$\begin{cases} v_i = (u_{2i} \mod v_B) + 1 \\ p_i = (u_{2i+1} \mod (p_B - 6)) + 7 \end{cases}$$

Par ailleurs le sac à dos possède une contenance maximale. Il ne peut donc pas transporter plus qu'un poids maximal dont on indiquera la valeur par le biais de la variable P_{max}.

Question 2 *Donnez la valeur ainsi que le poids des objets associés au problème suivant :*
a) $SD_{5,50,20}$.

Les parties 2, 3 et 4 sont indépendantes.

2 Sac à dos avec répétitions

Dans cette section, on suppose que chaque pièce est disponible en quantité **illimitée**. Cela signifie que pour augmenter la valeur contenue dans le sac à dos, on peut prendre plusieurs exemplaires d'un même objet (par exemple lorsque son poids est faible, mais que sa valeur est grande). Le contenu d'un sac est donc une liste d'objets chacun d'eux étant associé à son nombre d'apparition dans le sac.
Un sac est dit **maximal** dès lors qu'on ne peut plus lui ajouter ne serait-ce qu'un seul objet sans lui faire dépasser le poids maximal P_{max}.

2.1 Résolution du problème pour de petites instances

Question 3 *Donnez le nombre de sacs maximaux associés aux problèmes suivants :*
a) $SD_{5,50,20}, P_{max} = 20$ ***b)*** $SD_{10,50,20}, P_{max} = 30$ ***c)*** $SD_{5,50,20}, P_{max} = 60$.

Question à développer pendant l'oral : Expliquez l'algorithme que vous avez développé. Quelle est sa complexité en temps ?

Parmi tous les sacs maximaux que vous avez calculés, l'un (ou plusieurs d'entre eux) possède une valeur qui est maximale. C'est donc la composition du sac qui minimise la perte de contenance du sac, tout en maximisant la valeur transportée. On qualifiera un tel sac d'**optimal**.

Question 4 *Donnez la valeur totale des objets transportés par un sac optimal pour les problèmes suivants :*
a) $SD_{5,50,20}, P_{max} = 20$ ***b)*** $SD_{10,50,20}, P_{max} = 30$ ***c)*** $SD_{5,50,20}, P_{max} = 60$.

2.2 Pour aller plus loin ...

Pour un problème SD_{n,v_B,p_B} donné, on notera $V(p)$ la valeur totale transportée par un sac optimal lorsque son poids ne peut excéder p. La question 6 répondait donc au calcul de $V(P_{max})$ pour certaines instances du problème.
De manière plus générale, exprimez $V(p)$, en fonction de $V(p')$ pour $p' < p$ et les valeurs et poids des différents objets associés à un problème donné. Déduisez en un algorithme qui permette un calcul de $V(P_{max})$.

Question 5 *Donnez la valeur totale des objets transportés par un sac optimal pour les problèmes suivants :*
a) $SD_{100,50,40}, P_{max} = 300 + (u_{20} \mod 200)$,
b) $SD_{200,50,100}, P_{max} = 1000 + (u_{20} \mod 400)$,
c) $SD_{500,400,50}, P_{max} = 1000 + (u_{30} \mod 400)$.

Question à développer pendant l'oral : Expliquez l'algorithme que vous avez développé. Quelle est sa complexité en temps et en espace ? Voyez-vous un moyen de l'améliorer ?

Question 6 *Donnez le poids total ainsi que le nombre d'objets (en comptant leur multiplicité), transportés par le sac optimal (celui-ci est unique pour les instances considérées) :*
a) $SD_{100,50,40}, P_{max} = 300 + (u_{20} \mod 200)$,
b) $SD_{200,50,100}, P_{max} = 1000 + (u_{20} \mod 400)$,
c) $SD_{500,400,50}, P_{max} = 1000 + (u_{30} \mod 400)$.

Question à développer pendant l'oral : Détaillez l'algorithme que vous avez développé. Donnez sa complexité en temps et en espace. Comme indiqué précédemment, le sac optimal peut ne pas être unique. Suggérez un algorithme qui permette de vérifier que le sac optimal est unique. Quelle est sa complexité en temps et en espace ?

3 Sac à dos fractionnaire

Dans cette partie, on imagine que les objets ne sont disponibles qu'en un unique exemplaire. Mais ils peuvent par contre être fractionnés. On peut ne prendre qu'une fraction $0 \leq \mu \leq 1$ d'un objet (v_i, p_i). Dans ce cas celui-ci pèse μp_i, et sa valeur vaut μv_i. Vous devez développer un algorithme qui résout ce problème de manière optimale.

Question 7 *Vous indiquerez la valeur du sac à dos optimal (arrondie à 2 chiffres après la virgule) pour les instances suivantes :*
a) $SD_{5,50,20}, P_{max} = 30$ ***b)*** $SD_{1000,50,100}, P_{max} = 30000$ ***c)*** $SD_{5000,400,50}, P_{max} = 100000$.

Question à développer pendant l'oral : Prouvez que l'algorithme que vous avez proposé est optimal. Quelle est sa complexité ?

4 Sac à dos sans répétition

Dans cette section, on suppose que chaque pièce n'est disponible qu'en un unique exemplaire. Le contenu d'un sac est donc simplement la liste des objets qu'il contient sans multiplicité. Chaque objet ne peut donc apparaître qu'une seule et unique fois.
Là encore on dit qu'un sac est dit **maximal** dès lors qu'on ne peut plus lui ajouter même un seul objet sans lui faire dépasser son poids maximal P_{max}.

4.1 Résolution du problème pour de petites instances

Question 8 *Donnez le nombre de sacs maximaux associés aux problèmes suivants :*
a) $SD_{5,50,20}, P_{max} = 30$ ***b)*** $SD_{10,50,20}, P_{max} = 20$ ***c)*** $SD_{20,50,20}, P_{max} = 30$.

Question à développer pendant l'oral : Expliquez l'algorithme que vous avez développé. Quelle est sa complexité en temps ?

Question 9 *Donnez la valeur totale des objets transportés par un sac optimal pour les problèmes suivants :*
a) $SD_{5,50,20}, P_{max} = 30$ ***b)*** $SD_{10,50,20}, P_{max} = 20$ ***c)*** $SD_{20,50,20}, P_{max} = 30$.

4.2 Pour aller plus loin ...

Pour un problème SD_{n,v_B,p_B} donné, on notera $V(p,i)$ la valeur totale transportée par un sac optimal lorsque le poids du sac ne peut dépasser p et que seuls les i premiers objets associés au problème sont utilisés. Trouvez une expression donnant $V(p,i)$ en fonction des $V(p',i')$ pour $p' \leq p$ et $i' < i$.

Question 10 *Donnez la valeur totale des objets transportés par un sac optimal pour les problèmes suivants :*
a) $SD_{100,50,100}, P_{max} = 1000$ ***b)*** $SD_{200,50,100}, P_{max} = 1000$ ***c)*** $SD_{500,400,50}, P_{max} = 1000$.

Question à développer pendant l'oral : Expliquez l'algorithme que vous avez développé. Quelle est sa complexité en temps et en espace ? Voyez-vous un moyen de l'améliorer ?

4.3 Instances particulières du problème sans répétition

On considère dans cette section, des instances du problèmes du sac à dos pour lesquelles les objets possèdent des poids qui suivent une loi particulière. Plus précisément :
On définit par $SD^{\star}_{n,v_B,p_B}$ le problème du sac à dos pour n objets dont les valeurs et poids sont donnés par la suite de couples $(v_i,p_i)_{0\leq i\leq(n-1)}$, telle la suite $(p_i)_{0\leq i\leq n-1}$ possède la propriété suivante :

$$p_i > \sum_{j=0}^{i-1} p_j, \text{pour } i > 0.$$

Question à développer pendant l'oral : Montrez que $p_i \geq 2^i$. En déduire un algorithme efficace pour la résolution du problème du sac à dos dans ce cas précis. Donnez sa complexité en temps.

◊
◊ ◊
◊

Fiche réponse type: Sac à dos

$\widetilde{u_0}$: 1

Question 1

a) 47793

b) 11233

c) 59895

Question 2

a) (v=2, p=20),(v=26, p=20),(v=23, p=20),(v=28, p=17),(v=8, p=18)

Question 3

a) 5

b) 14

c) 35

Question 4

a) 28

b) 72

c) 84

Question 5

a) 2610

b) 9350

c) 67490

Question 6

a) 406,58

b) 1309,187

c) 1190,170

Question 7

a) 44.90

b) 21548.65

c) 921593.80

Question 8

a) 5

b) 10

c) 155

Question 9

a) 28

b) 36

c) 127

Question 10

a) 1196

b)	1800
c)	23649

◊ ◊ ◊ ◊

Fiche réponse: Sac à dos

Nom, prénom, u_0: ....

Question 1

a)

b)

c)

Question 2

a)

Question 3

a)

b)

c)

Question 4

a)

b)

c)

Question 5

a)

b)

c)

Question 6

a)

b)

c)

Question 7

a)

b)

c)

Question 8

a)

b)

c)

Question 9

a)

b)

c)

Question 10

a)

b)

c)