

Esiee-Paris - cours d'algorithmique - feuille d'exercices numéro 1

Septembre 2023 – R. Natowicz, I. Alamé, A. Çela, X. Hilaire, T. Wu, W. Xu

La solution de chaque exercice doit être programmée en langage Java, voir l'ébauche de programme.

Exercice 1. Spots et slots. Quand une marque commerciale veut diffuser un “spot” à la télévision, elle s’adresse à une société de diffusion. Cette dernière dispose d’un “slot” c’est-à-dire d’un intervalle de temps à l’intérieur duquel elle diffuse des spots que ses clients lui ont confiés.

Le slot de la société de diffusion est de durée T et son carnet de commandes contient K spots. Le spot k , $0 \leq k < K$, est de durée $d(k)$. La diffusion du spot k lui apporte un gain $g(k)$. Ce gain est le prix négocié avec la marque commerciale diminué du montant lui-même négocié qui sera reversé à la chaîne de télévision.

Juliette, étudiante à l’ESIEE, est en stage dans cette société de diffusion. Elle observe le “commercial” de la société. Pour décider des spots qui seront diffusés dans le slot de durée T , il classe les spots par gains décroissants et les diffuse dans cet ordre. A chaque étape de son procédé il ajoute le spot de gain maximum dont la durée est inférieure ou égale au temps encore disponible dans le slot.

Le slot est de durée $T = 100$ secondes. Le carnet de commandes contient les dix spots suivants.

k	$d(k)$	$g(k)$	k	$d(k)$	$g(k)$
0	20	25	5	40	35
1	20	25	6	10	15
2	70	65	7	80	75
3	10	15	8	10	15
4	10	5	9	40	45

Quel est le gain total obtenu par le “commercial” de cette société?

Juliette a suivi l’excellent cours d’algorithmique de son école. Elle décide de traiter le problème d’une autre façon. Elle établit l’équation de récurrence du gain maximum, elle écrit un programme qui calcule et mémorise dans un tableau M les valeurs maximum de tous les sous-problèmes, puis elle écrit un programme d’affichage des spots à diffuser dans le slot pour obtenir le gain maximum.

Quelle est le gain total de la solution de Juliette?

Écrire un programme en langage Java qui calcule et affiche un sous-ensemble de spots de gain total maximum. L’exécuter sur l’exemple ci-dessus, et sur d’autres que vous construirez vous-mêmes.

Exercice 2. Sous-ensemble de somme S . On considère l’ensemble $X = \{x_0, \dots, x_{n-1}\}$ d’entiers positifs ou nul, et une valeur entière S . On veut répondre à la question “Existe-t-il un sous-ensemble de X , de somme S ?” et, s’il en existe, on veut en afficher un.

Exemple: il existe un sous-ensemble de $X = \{1, 3, 5\}$ de somme $S = 6$ car $1 + 5 = 6$ mais il n’en existe pas de somme $S = 7$.

On note $e(k, s)$ la vérité de la proposition “Il existe un sous-ensemble de $X[0 : k]$, de somme s ”. Avec cette notation, répondre à la question posée ci-dessus c’est donner la valeur de vérité de la proposition $e(n, S)$.

– Supposons le problème résolu. Donner l’expression de la valeur $e(n, S)$.

– Généralisation: donner l’expression de la valeur $e(k, s)$ pour tous k et s , $1 \leq k < n + 1$ et $0 \leq s < S + 1$.

– Base de la récurrence: donner les valeurs $e(0, s)$ pour tout s , $0 \leq s < S + 1$.

– Écrire une fonction `boolean[][] calculerE(int[] X, int S)` qui calcule et retourne un tableau booléen $E[0 : n + 1][0 : S + 1]$ de terme général $E[k][s] = e(k, s)$. Quelle est la forme de son temps de calcul?

– Écrire une fonction `void ase(int[][] E, int[] X, int k, int s)` qui affiche un sous-ensemble de $X[0 : k]$ de somme s (si un tel sous-ensemble existe.) L’appel principal de cette fonction est `void ase(E, X, n, S)`.

Remarque: cet appel de fonction n’a aucun sens quand $e(n, S)$ est faux.

Exercice 3. Sous-ensemble de somme S avec répétitions autorisées. Reprendre le même problème à une différence près: les répétitions de valeurs sont autorisées.

Exemples: 1) avec $X = \{1, 3, 5\}$ on peut obtenir la somme $S = 7$ car $1 + 3 + 3 = 7$ de même que $1 + 3 + 1 + 1 + 1$, $1 + 1 + 5$, $1 + 1 + 1 + 1 + 1 + 1 + 1$, cette somme étant impossible à obtenir sans répétition. 2) Avec X ne contenant que des nombres pairs, obtenir une somme S impaire est impossible.

Donner l’équation de récurrence des valeurs $e(k, s)$ et écrire le programme Java qui résout le problème.

Question subsidiaire: lorsque les répétitions sont autorisées, les éléments de l’ensemble X doivent être *strictement* positifs. Pourquoi?