

ÉPREUVE D'INFORMATIQUE

(Durée : 2 heures)

L'utilisation des calculatrices **n'est pas autorisée** pour cette épreuve.

Le langage de programmation choisi par le candidat doit être spécifié en tête de copie.

On attachera une grande importance à la concision, à la clarté, et à la précision de la rédaction.

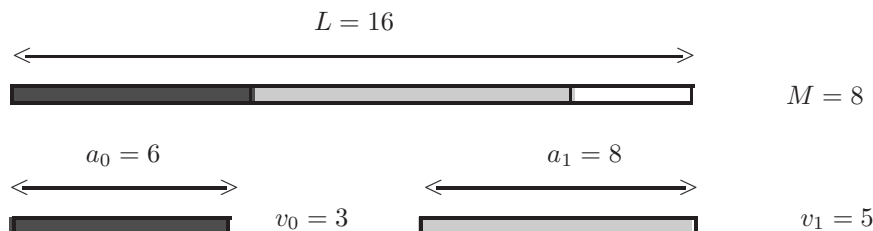
Découpe de tissu

Le temps d'exécution $T(f)$ d'une fonction f est le nombre d'opérations élémentaires (addition, soustraction, multiplication, division, affectation, etc.) nécessaire au calcul de f . Lorsque ce temps d'exécution dépend d'un paramètre n , il sera noté $T_n(f)$. On dit que la fonction f s'exécute en temps $O(n^\alpha)$ s'il existe $K > 0$ tel que pour tout n , on a $T_n(f) \leq Kn^\alpha$.

Un tailleur de tissus dispose d'une grande pièce de tissu ainsi que d'une liste de n produits qu'il peut fabriquer à l'aide de ce tissu. La pièce de tissu n'a qu'une seule dimension, il s'agit donc d'un ruban de longueur L . Pour chaque produit, le tailleur sait quelle quantité de tissu est nécessaire et il connaît son prix de vente. Le tailleur doit décider quels produits il va réaliser pour maximiser sa rentrée d'argent M . Chacun des n produits réalisables a une longueur a_i et un prix de vente v_i , ($0 \leq i < n$). Toutes les dimensions et prix de vente sont des entiers strictement positifs.

Partie I. Coupe d'un ruban avec répétition

Dans cette partie, chaque produit peut être réalisé un nombre arbitraire de fois (éventuellement nul). Un découpage (non optimal) possible est illustré ci-dessous pour $n = 2$.



Informatiquement, on stockera la suite des longueurs $\langle a_0, a_1, \dots, a_{n-1} \rangle$ et des prix de vente $\langle v_0, v_1, \dots, v_{n-1} \rangle$ dans deux tableaux a et v de n éléments. On remarquera aussi que la découpe du tissu est commutative, puisque découper le produit i puis le produit j donne le même résultat que découper le produit j puis le produit i .

Question 1 Considérer l'exemple suivant : $n = 2$, $a_0 = 6$, $a_1 = 8$, $v_0 = 3$ et $v_1 = 5$. Calculer M lorsque $L = 14, 16, 18, 24$.

Question 2 Reprendre l'exemple précédent avec $v_0 = 4$.

On se place d'abord dans les cas $n = 2$ ou $n = 3$. Pour $n = 2$, on calcule tous les prix de vente obtenus en coupant $0, 1, 2, \dots$, ou $L \div a_0$ fois le premier produit et le reste du tissu avec le

deuxième produit. Et on retient le M maximal. De même pour $n = 3$, on commence par couper plusieurs fois le premier produit, puis le deuxième produit et le reste avec le troisième produit.

Question 3 Écrire la fonction `coutRuban2Produits`(a_0, v_0, a_1, v_1, L) qui prend en arguments les longueurs a_0, a_1 et les prix de ventes v_0, v_1 de deux produits et la longueur totale L du ruban de tissu ; et qui retourne le prix de vente maximal M possible après découpe.

Question 4 Écrire la fonction `coutRuban3Produits`($a_0, v_0, a_1, v_1, a_2, v_2, L$) qui prend en arguments les longueurs a_0, a_1, a_2 et les prix de ventes v_0, v_1, v_2 de trois produits et la longueur totale L du ruban de tissu ; et qui retourne le prix de vente maximal M possible après découpe.

On peut montrer qu'un tel algorithme de calcul prend un temps exponentiel en $O(L^n)$ dans le cas général de n produits. On va donc utiliser une méthode dite de « programmation dynamique » dans ce cas général en prenant pour acquise la formule de récurrence suivante, où $M(x)$ est le coût maximal obtenu après découpe des n produits pour un ruban de longueur x :

$$M(x) = \max \{M(x - a_i) + v_i \mid 0 \leq i < n, a_i \leq x\}$$

Par convention, on posera que le maximum de l'ensemble vide vaut 0. On stockera les valeurs de $M(x)$ pour $0 \leq x \leq L$ dans un tableau M .

Question 5 Écrire la fonction `coutRuban`(a, v, n, L) qui prend en arguments les tableaux a des longueurs et v des prix de vente des n produits, ainsi que la longueur L du ruban ; et qui retourne le prix de vente maximal $M(L)$ en calculant le tableau M décrit précédemment. Donner un ordre de grandeur du temps d'exécution de cette fonction.

Pour imprimer la solution donnant le prix de vente maximal, il faut se souvenir de la coupe pratiquée pour obtenir ce $M(x)$ maximal. Pour cela, on utilise un tableau D tel que $D[x]$ donne l'indice i du produit tel que $M(x - a_i) + v_i$ soit maximal. Lorsque $M(x) = 0$, on posera $D[x] = -1$. Une liste de découpes réalisant la valeur $M(L)$ peut alors être retrouvée à partir du tableau D . Par exemple, la première coupe est $i_0 = D[L]$, la seconde $i_1 = D[L - a[i_0]]$, ... jusqu'à ce qu'on tombe sur la valeur par défaut -1 .

Question 6 Écrire la fonction `decoupageRuban`(a, v, n, L) qui prend en arguments les tableaux a des longueurs et v des prix de vente, ainsi que la longueur L du ruban ; et qui imprime la liste des découpes successives produisant la valeur $M(L)$, en calculant le tableau D . Donner un ordre de grandeur du temps d'exécution de cette fonction.

Partie II. Coupe d'un ruban sans répétition

Dans cette partie, le tailleur ne peut fabriquer plus d'une fois le produit i car la demande n'excède pas ce nombre.

Question 7 Trouver une formule de récurrence donnant le coût maximal $M(i, x)$ que l'on peut obtenir avec un ruban de longueur x en ne découpant que parmi les produits $0, 1, \dots, i - 1$.

On stockera les valeurs de $M(i, x)$ pour $0 \leq i \leq n$ et $0 \leq x \leq L$ dans un tableau M à deux dimensions.

Question 8 Écrire une fonction `coutRubanSansRepetitions`(a, v, n, L) qui prend en arguments les tableaux a des longueurs et v des prix de vente des n produits ainsi que la longueur L du ruban ; et qui retourne le prix de vente maximal $M(n, L)$ en calculant le tableau M décrit précédemment. Donner un ordre de grandeur du temps d'exécution de cette fonction.

Question 9 Réécrire cette fonction en ne se servant que de deux tableaux de $L + 1$ éléments, réduisant la taille mémoire utilisée à $O(2(L + 1))$.

★ ★
★