

IA Challenge – Surface totale du papier cadeau

laurent.jospin.59@free.fr, <http://jospin.lstl.fr>

Lycée Saint-Louis, Paris

On considère un ensemble de cadeaux à emballer de tailles différentes. Pour un cadeau de taille : $L \times l \times h$, on considérera qu'il faut utiliser un rectangle de papier de dimensions $(2(L + l) + 2) \times (l + h + 2)$, mais on pourra permuer librement les trois dimensions du cadeau. Toutes les dimensions seront entières naturelles.

L'objectif est de disposer les rectangles nécessaires à l'emballage de tous les cadeaux dans le plan de sorte que le rectangle qui les englobent tous soit de surface minimale.

Le nombre de cadeaux pourra varier entre 1 et 100 suivant les instances du problèmes sur lequel votre programme sera testé.

1 Informatique de tronc commun (en Python)

L'objectif est d'écrire un programme comportant une fonction principale `papier_minimal(cadeaux)` (et toutes les fonctions auxiliaires dont vous aurez besoin) et qui renvoie une liste contenant pour chaque cadeau l'ordre choisi (L, l, h) pour les dimensions et les coordonnées entières (x_1, y_1, x_2, y_2) avec $x_1 < x_2$ et $y_1 < y_2$ dans le plan correspondant aux abscisses et ordonnées minimales et maximales du rectangle dans le plan. Vous devrez précisément renvoyer une liste de 7-uplets de la forme : $[(L, l, h, x_1, y_1, x_2, y_2), \dots]$. Les rectangles peuvent se toucher mais pas se superposer.

L'ordre des rectangles doit correspondre à l'ordre des cadeaux.

Le temps d'exécution de vos programmes ne doit pas dépasser 30 secondes quelque soit l'instance, vous pouvez donc adapter votre stratégie en fonction de la taille de l'instance.

2 MP2I (en C ou en Ocaml, éventuellement en Python)

Vous écrirez un programme qui lit sur l'entrée standard le nombre de cadeaux puis, pour chaque cadeau, ses 3 dimensions séparées par des espaces, et vous écrirez sur la sortie standard les 7 valeurs expliquées ci-dessus à raison d'une ligne par cadeau dans le même ordre que les cadeaux d'origine.

Remarque. Les coordonnées des rectangles ne doivent pas nécessairement avoir pour origine (0,0) et être positive, ce qui compte c'est que les rectangles ne se superposent pas mais ils peuvent se toucher. La surface totale du papier cadeau utilisé par votre programme sera donnée par $(\max_{\text{cadeaux}}(x_2) - \min_{\text{cadeaux}}(x_1)) \times (\max_{\text{cadeaux}}(y_2) - \min_{\text{cadeaux}}(y_1))$