

Problème : intervalles

Soit IS un ensemble d'intervalles de forme $IS = \{[x_1, y_1], [x_2, y_2], \dots, [x_n, y_n]\}$, avec $x_i \leq y_i$. Dans cet ensemble les intervalles ne forment pas une partition, ça veut dire qu'un point peut se retrouver dans plusieurs intervalles. Nous définissons l'aire de l'ensemble IS comme étant $area(IS) = \sum_{i=1}^n (b_i - a_i)$. Deux ensembles IS_1 et IS_2 sont équivalents s'ils contiennent les mêmes points : $IS_1 \equiv IS_2 \Leftrightarrow (\forall x \in IS_1 \Rightarrow x \in IS_2) \wedge (\forall x \in IS_2 \Rightarrow x \in IS_1)$. Décrivez un algorithme qui produit un ensemble IS' équivalent avec IS avec une aire minimale.

Exemple : $IS = \{[-8, 2], [1, 5], [7, 9]\} \Rightarrow IS' = \{[-8, 5], [7, 9]\}$

Description de l'entrée (IN) : les intervalles formant l'ensemble IS , un par ligne (deux nombres séparés par un espace). La première ligne va contenir le nombre d'intervalles.

Description de la sortie (OUT) : les intervalles formant l'ensemble IS' , un par ligne (deux nombres séparés par un espace). La sortie doit être triée en ordre croissant par rapport à la borne gauche. La première ligne doit contenir le nombre d'intervalles.

Observation : chaque ligne affichée sera terminée par deux caractères (CR et LF), correspondant à la chaîne "\r\n".

Exemple 1

IN :

```
3
-8 2
1 5
7 9
```

OUT :

```
2
-8 5
7 9
```

Exemple 2

IN :

```
2
-8 2
7 9
```

OUT :

2
-8 2
7 9