

B. Bike Parking

Nom du problème	bikeparking
Limite de temps	1 seconde
Limite mémoire	1 gigaoctet

Sanne a récemment lancé un nouveau business: louer des places de vélos premium à la station de train d'Eindhoven. Pour maximiser ses profits, elle partitionne les places de vélo en N catégories différentes, numérotées de 0 à $N - 1$. La catégorie 0, la catégorie premium, est située très proche des quais de la gare. Les catégories avec un numéro supérieur sont moins bien situées (plus le numéro est élevé, plus les places sont loin). Le nombre de places de parking dans la catégorie t est x_t .

Les utilisateurs garant leur vélo se voient attribuer une place de parking via une application. Chaque utilisateur a un niveau d'abonnement et s'attend à recevoir une place de parking dans la catégorie correspondant à son niveau d'abonnement. Cependant, les conditions d'utilisation ne garantissent pas aux utilisations une place de parking dans leur catégorie respective.

Si un utilisateur avec un niveau d'abonnement s se voit assigner une place de parking de catégorie t , il se passe alors l'une des trois choses suivantes:

1. Si $t < s$, l'utilisateur sera heureux et donnera un pouce vers le haut à l'application.
2. Si $t = s$, l'utilisateur ne fera rien.
3. Si $t > s$, l'utilisateur sera furieux et donnera un pouce vers le bas à l'application.

A ce jour, l'application de Sanne a $y_0 + y_1 + \dots + y_{N-1}$ utilisateurs, où y_s est le nombre d'utilisateurs avec un niveau d'abonnement s . Elle a besoin de votre aide pour attribuer des places de parking aux utilisateurs. Chaque utilisateur devrait obtenir exactement une place de parking. Une place de parking ne peut être assignée qu'à un unique utilisateur, mais il est aussi possible qu'une place de parking ne soit assignée à aucun utilisateur.

Sanne souhaite maximiser la note de son application. Vous devez maximiser $U - D$, avec U le nombre de pouces vers le haut qu'a reçus l'application, et D le nombre de pouces vers le bas.

Entrée

La première ligne contient un entier N , le nombre de niveaux d'abonnement.

La deuxième ligne contient N entiers x_0, x_1, \dots, x_{N-1} , le nombre de places de parking dans chacune des catégories.

La troisième ligne contient N entiers y_0, y_1, \dots, y_{N-1} , le nombre d'utilisateurs dans chacun des niveaux d'abonnement.

Sortie

Vous devez afficher un entier, la valeur maximum de $U - D$ en attribuant des places de parking aux utilisateurs de manière optimale.

Contraintes et Répartition des points

- $1 \leq N \leq 3 \cdot 10^5$.
- $0 \leq x_i, y_i \leq 10^9$ pour $i = 0, 1, \dots, N - 1$.
- $y_0 + y_1 + \dots + y_{N-1} \leq x_0 + x_1 + \dots + x_{N-1} \leq 10^9$.

Votre solution sera testée sur un ensemble de sous-tâches, rapportant chacune un certain nombre de points. Chaque sous-tâche contient un ensemble de tests. Pour avoir les points d'une sous-tâche, vous devez valider tous les tests de cette sous-tâche.

Sous-tâche	Points	Contraintes
1	16	$N = 2, x_i \leq 100, y_i \leq 100$
2	9	$x_i = x_j = y_i = y_j$ pour tout i, j . C'est à dire, tous les x et y de l'entrée sont égaux.
3	19	$x_i, y_i \leq 1$
4	24	$N, x_i, y_i \leq 100$
5	32	Pas de contraintes supplémentaires.

Exemples

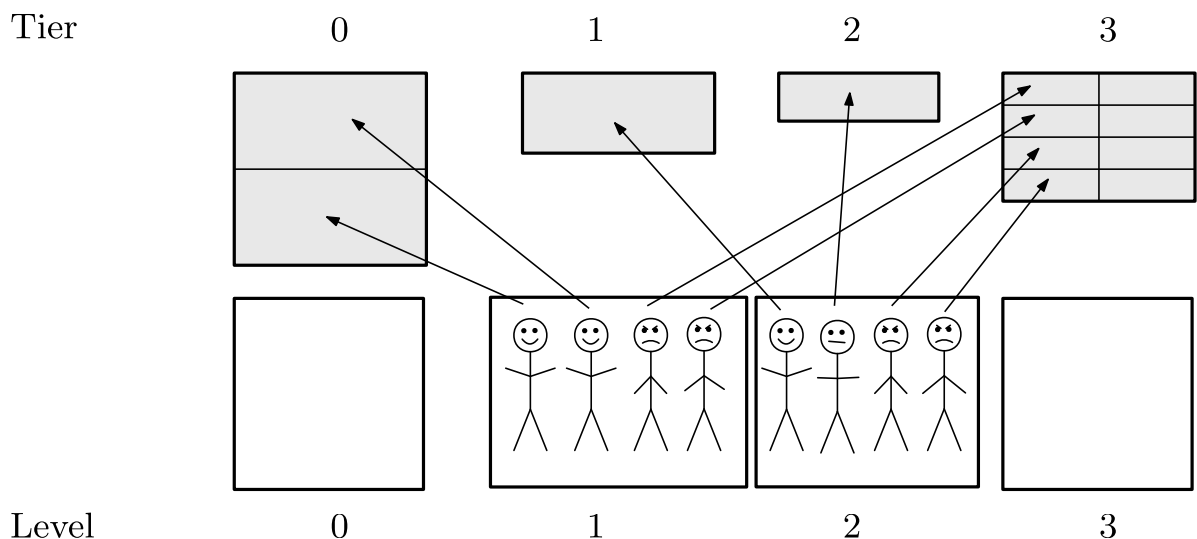
Remarquez que certains de ces exemples ne sont pas inclus dans toutes les sous-tâches. L'exemple i est au moins inclus dans la sous-tâche i .

Dans le premier exemple, vous pouvez assigner les places de parking de catégorie 0 aux utilisateurs avec un abonnement de niveau 0, assigner aux deux utilisateurs de niveau 1 des places de catégorie 0 (ce qui génère 2 pouces vers le haut), et assigner des places de niveau 1 aux autres utilisateurs de niveau 1. La note de l'application est donc 2.

Dans le deuxième exemple, vous pouvez assigner des places de catégorie 0 aux utilisateurs de niveau 1, des places de niveau 1 aux utilisateurs de niveau 2, et une place de niveau 2 à l'utilisateur de niveau 0. L'application reçoit 2 pouces vers le haut et un 1 pouce vers le bas, la note de l'application est donc 1.

Dans le troisième exemple, vous pouvez assigner aux utilisateurs de niveau 1 des places de catégorie 0, aux utilisateurs de niveau 0 des places de catégorie 2, et à l'utilisateur de niveau 4 une place de catégorie 3. L'application reçoit à nouveau 2 pouces vers le haut et 1 pouce vers le bas, la note de l'application est donc 1.

Le quatrième exemple est illustré ci-dessous. Vous pouvez assigner des places de catégories 0, 0, 3 et 3 aux utilisateurs de niveau 1, ce qui génère 2 pouces vers le haut et 2 pouces vers le bas. Ensuite, vous pouvez assigner des places de catégories 1, 2, 3 et 3 aux utilisateurs de niveau 2, ce qui génère 1 pouce vers le haut et 2 pouces vers le bas. Au total, l'application reçoit 3 pouces vers le haut et 4 pouces vers le bas. La note de l'application est donc -1 .



Dans le cinquième exemple, vous pouvez assigner à chaque utilisateur une place de parking qui correspond à leur niveau d'abonnement, la note de l'application est donc 0.

Entrée	Sortie
<pre> 2 3 3 1 3 </pre>	<pre> 2 </pre>
<pre> 3 1 1 1 1 1 1 </pre>	<pre> 1 </pre>
<pre> 6 1 0 1 1 0 1 1 1 0 0 1 0 </pre>	<pre> 1 </pre>
<pre> 4 2 1 1 8 0 4 4 0 </pre>	<pre> -1 </pre>
<pre> 1 1000000000 1000000000 </pre>	<pre> 0 </pre>