

B. Estacionamento de Bicicleta

Nome do problema	Estacionamento de bicicleta
Tempo Limite	1 segundo
Limite de Memória	1 gigabyte

Recentemente, Sanne concebeu uma ideia de negócio lucrativo: aluguel e estacionamento premium de bicicleta na estação de trem de Eindhoven. Para maximizar seu lucro, ela dividiu as vagas do estacionamento de bicicleta em N níveis diferentes (*tiers*), numerados de 0 a $N - 1$.

Nível 0, o nível premium, está localizado muito próximo das plataformas de trem. Níveis com números altos consistem em vagas de estacionamento que são piores (quanto mais alto o nível, pior a vaga). O número de vagas no nível t é x_t .

As vagas para os usuários que estacionam as suas bicicletas são alocadas por meio de um app. Cada usuário tem um nível para sua subscrição (*level*) no serviço e espera uma vaga de estacionamento no nível correspondente. Entretanto, os termos do serviço não garantem aos usuários uma vaga em seu respectivo nível.

Se um usuário com nível de subscrição s é atribuído a uma vaga no nível t , então uma das três coisas seguintes acontece:

1. Se $t < s$, o usuário ficará feliz e dará um voto positivo para o app.
2. Se $t = s$, o usuário ficará satisfeito e não fará nada.
3. Se $t > s$, o usuário ficará com raiva e dará um voto negativo para o app.

Hoje, o app de Sanne tem $y_0 + y_1 + \dots + y_{N-1}$ usuários, onde y_s é o número de usuários com subscrição de nível s . Ela precisa sua ajuda para alocar os usuários às vagas de estacionamento. Cada usuário deveria receber exatamente uma vaga. Nenhuma vaga pode ser atribuída a mais de um usuário, mas é aceitável que algumas vagas de estacionamento não sejam atribuídas a nenhum usuário.

Sanne quer maximizar a classificação do seu app. Seja U o número de votos positivos e D o número de votos negativos. Sua tarefa é maximizar $U - D$.

Entrada

A primeira linha contém um inteiro N , o número de níveis das vagas de estacionamento ou níveis de subscrição.

A segunda linha contém N inteiros x_0, x_1, \dots, x_{N-1} , o número de vagas nos diferentes níveis.

A terceira linha contém N inteiros y_0, y_1, \dots, y_{N-1} , o número de usuários em cada nível de subscrição.

Saída

A saída deve ser um inteiro, o valor máximo possível para $U - D$, alcançado com a alocação ótima de usuários para as vagas de estacionamento.

Restrições e Pontuação

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

Sua solução será testada em um conjunto de grupos de teste, cada um valendo um número de pontos. Cada grupo de teste contém um conjunto de casos de teste. Para conseguir os pontos em um grupo de teste, você precisa resolver todos os casos de teste naquele grupo de teste.

Grupo	Pontuação	Limites
1	16	$N = 2, x_i \leq 100, y_i \leq 100$
2	9	$x_i = x_j = y_i = y_j$ para todo i, j . Em outras palavras todos os x 's e y 's na entrada são iguais.
3	19	$x_i, y_i \leq 1$
4	24	$N, x_i, y_i \leq 100$
5	32	Sem restrições adicionais.

Exemplos

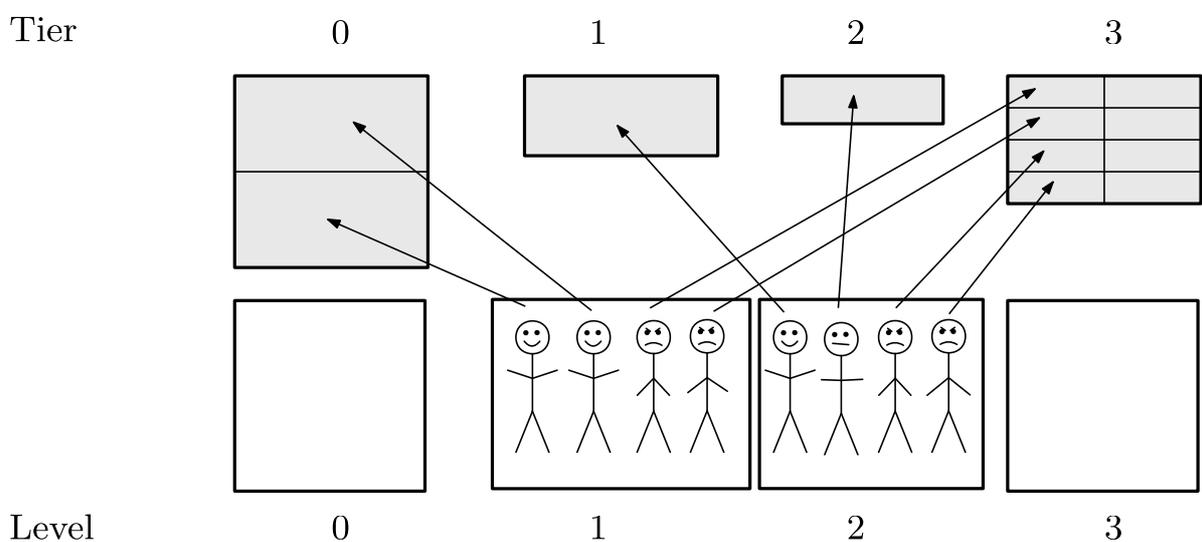
Note que alguns dos exemplos não são entradas válidas para todos os grupos de teste. O i -ésimo exemplo é válido, pelo menos, para o i -ésimo grupo de teste.

No primeiro exemplo, você pode atribuir o usuário com subscrição de nível 0 à uma vaga de nível 0, atribuir dois usuários do nível 1 a vagas no nível 0 (resultando em 2 votos positivos), e atribuir o usuário do nível 1 restante para uma vaga do nível 1. Isto resulta em uma classificação de 2.

No segundo exemplo, você pode atribuir o usuário do nível 1 à vaga do nível 0, o usuário do nível 2 à vaga do nível 1, e o usuário do nível 0 à vaga do nível 2. Isto dá 2 votos positivos e 1 voto negativo, resultando em uma classificação de 1.

No terceiro exemplo, você pode atribuir o usuário do nível 1 à vaga do nível 0, o usuário do nível 0 à vaga do nível 2, e o usuário do nível 4 à vaga do nível 3. Novamente, isto dá 2 votos positivos e 1 voto negativo, resultando em uma classificação de 1.

O quarto exemplo está ilustrado abaixo. Você pode atribuir os usuários do nível 1 à vagas dos níveis 0, 0, 3 e 3, resultando em 2 votos positivos e 2 votos negativos. Na sequência, atribua os usuários do nível 2 às vagas dos níveis 1, 2, 3 e 3, resultando em 1 voto positivo e 2 votos negativos. Isto equivale a 3 votos positivos e 4 votos negativos, então a classificação resultante é -1.



No quinto exemplo, você pode atribuir todos os usuários para uma vaga que corresponde à seu nível de subscrição, então a classificação será 0.

Entrada	Saída
<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>