

# Comércio de Vinhos na Gergóvia

Contest Local, Universidade de Ulm  Alemanha

Timelimit: 2

Como você deve saber do cômico "Astérix e o Escudo de Arverne", Gergóvia consiste de uma única rua e cada habitante da cidade é um vendedor de vinho. Você quer saber como essa economia funciona? Bem simples: todos compram vinhos dos outros habitantes da cidade. Cada dia, cada habitante decide quanto vinho ele quer comprar ou vender. Curiosamente, a demanda e o fornecimento são sempre os mesmos de modo que cada habitante consegue o que deseja.

Contudo, há um problema: transportar o vinho de uma casa para outra resulta em trabalho. Sendo todos os vinhos igualmente bons, os habitantes de Gergóvia não se importam com quais pessoas eles irão comercializar, eles estão somente interessados em vender e comprar um quantidade específica de vinho. Eles são espertos o suficiente para imaginar uma forma de negociar de modo que todo o montante de trabalho necessário para o transporte seja minimizado.

Nesse problema você está sendo inquerido para reconstruir o comércio durante um dia em Gergóvia. Para simplificar, nós assumimos que as casas são construídas ao longo de uma linha reta com a mesma distância entre as casas adjacentes. Transportar uma garrafa de vinho de uma casa para uma casa adjacente resulta em uma unidade de trabalho.

## Entrada

A entrada consiste de vários casos de teste.

Cada caso de teste inicia com o número de habitantes  $n$  ( $2 \leq n \leq 100000$ ). A linha seguinte contém  $n$  inteiros  $a_i$  ( $-1000 \leq a_i \leq 1000$ ). Se  $a_i \geq 0$ , isso significa que cada habitante que vive na  $i^{\text{th}}$  casa, deseja comprar  $a_i$  garrafas de vinho, caso contrário se  $a_i < 0$ , ele deseja vender  $-a_i$  garrafas de vinho. Você pode assumir que os números  $a_i$  resumem a 0.

O último caso de teste é seguido por uma linha contendo 0.

## Saída

Para cada caso de teste, imprima a quantidade mínima de unidades de trabalho necessárias para que todo habitante tenha sua demanda cumprida. Você pode assumir que este número cabe em um inteiro de 64 bits com sinal (em C/C++ você pode usar o tipo de dados "long long", em JAVA o tipo de dados "long").

Exemplo de Entrada	Exemplo de Saída
5 5 -4 1 -3 1 6 -1000 -1000 -1000 1000 1000 1000 0	9 9000