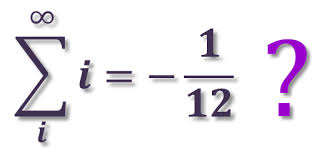
**Problema A: Soma Natural**

*TL: 1 segundo (~50 casos)*

*ML: 64 MB*

Foi descoberto recentemente que a soma de todos os números naturais até o infinito vale (-1/12). Ou seja, a soma de 1+2+3+4+5+.... vale (-1/12). É um resultado muito contra-intuitivo, por isso nós vamos nos limitar a calcular para valores menores.



Dado um número **Y**, imprima até que número **X** devemos chegar para que a soma de 1+2+3+...+**X** seja maior ou igual à um determinado valor **Y**.

**Entrada**

Há varios testes. A primeira linha de cada caso de teste contém um número **Y** (1 <= **Y** <= 10^9). Leia até o **EOF**.

**Saída**

Para cada valor da entrada imprima **X**.

|  |  |
| --- | --- |
| **Entrada de Teste**  1  9  1000  100000000 | **Saída de Teste**  1  4  45  14142 |



**Problema B: Lemmings**

*TL: 2 segundos (~50 casos)*

*ML: 64 MB*

Oh não! Mais Lemmings! No planeta dos Lemmings uma grande batalha está sendo travada entre os impérios dos Lemmings azuis e verdes.

Cada Lemming tem um poder de batalha. Quando dois inimigos batalham, aquele com maior poder sobrevive, e tem o seu poder reduzido pela quantidade do outro. Suponha dois Lemmings, um de força 50 e outro de 40. Após a batalha, o primeiro ficará com força 10 e o segundo irá desaparecer. Se ambos tem a mesma força então os dois desaparecem.

A estratégia adotada pelos impérios é de sempre mandar o Lemming mais forte para a batalha, ou seja, sempre os dois mais fortes de cada império lutam entre si. O processo é repetido até que algum dos dois lados não tenha mais soldados para lutar, sendo assim declarado o fim do combate.

Sua tarefa aqui é, dado a força dos soldados Lemmings, simular a batalha e imprimir o número de soldados restantes de cada lado.

**Entrada**

A primeira linha de cada caso de teste contém o número **N** (1 <= **N** <= 5\*10^4) e **M** (1 <= **M** <= 5\*10^4), o número de soldados de cada exército. Na linha seguinte há **N** números separados por espaço indicando a força de cada soldado do lado azul. Na outra linha há **M** números indicando a força de cada soldado do lado verde. As forças caberão em um número inteiro (**int**) e serão maiores ou iguais a zero. Leia até o **EOF**.

**Saída**

Para cada caso imprima separado por apenas um espaço o número restante de cada exército, no formato **azul verde**.

|  |  |
| --- | --- |
| **Entrada de Teste**  3 3  1 1 1  2 1 1  4 1  1 1 1 1  4  2 2  5 0  2 2 | **Saída de Teste**  0 1  0 0  2 0 |

**Problema C: Fossa o Bebum**

*TL: 1 segundo (~50 casos)*

*ML: 64 MB*

Fossa está acostumado a beber. Ele não perde nenhuma festa e bebe até cair em todas elas. Fossa está diante de várias garrafas de cerveja e quer abrí-las. O problema é que ele não tem um abridor. Então usará uma garrafa para abrir a outra. Ele é um gênio.

Cada garrafa possui uma marca e só consegue abrir garrafas de uma determinada marca. Garrafas abertas podem ser utilizadas para abrir as fechadas, ou seja, você pode reutilizar as garrafas! Você deve dizer quantas garrafas Fossa **não** conseguirá abrir.

**Entrada**

A primeira linha de cada caso de teste contém o número **N** (1 <= **N** <= 100) de garrafas. Cada uma das **N** linhas seguintes representa uma garrafa. Há dois números em cada linha, onde o primeiro valor é a marca da garrafa e o segundo valor é o tipo de marca que essa garrafa é capaz de abrir (1 <= **marca** <= 1000). Leia até o **EOF**.

**Saída**

Imprima o número de garrafas que Fossa o Bebum **não** conseguirá abrir.

|  |  |
| --- | --- |
| **Entrada de Teste**  4  1 1  2 2  3 3  4 4  4  1 2  2 3  3 4  4 1  5 1 1 1 2 2 3 3 4 4 3 | **Saída de Teste**  4  0  1 |

**Problema D: Invasão de Dragões**

*TL: 1 segundo (~50 casos)*

*ML: 64 MB*

Dragões estão invadindo o reino. Como rei, você convocou cavaleiros para matar todos os dragões.

Cada dragão tem uma espessura da sua escama. Para matar um dragão, um cavaleiro deve ter uma espada afiada com poder estritamente maior do que a espessura da escama do dragão.

Cada cavaleiro pode ser designado para matar apenas um dragão (se sua espada tiver um poder maior do que a espessura do dragão, como dito anteriormente).

Dadas as informações dos dragões e cavaleiros, existe alguma estratégia possível para acabar com todos os dragões?

**Entrada**

Há vários casos de testes. A primeira linha de cada caso de teste contém os número **N** (1 <= **N** <= **M**) e **M** (1 <= **M** <= 10^5), indicando o número de dragões e cavaleiros respectivamente. Na linha seguinte temos **N** números inteiros separados por espaço indicando as espessuras das escamas dos dragões. Na linha seguinte temos **M** números inteiros separados por espaço indicando o poder das espadas dos cavaleiros. Os valores caberão em um inteiro. Leia até o **EOF**.

**Saída**

Imprima **S** se existe uma estratégia ou **N** caso contrário.

|  |  |
| --- | --- |
| **Entrada de Teste**  3 3  2 1 2  2 3 3  3 4  1 2 3  1 2 3 3 1 1  100  200 | **Saída de Teste**  S  N  S |



**Problema E: Ordonhando As Idades**

*TL: 2 segundos (~1000 casos)*

*ML: 64 MB*

Em uma determinada cidade, o prefeito decidiu fazer um gráfico com as idades das pessoas. Para isso ele precisa ordenar as idades de todas as pessoas.

Sua tarefa como auxiliar do prefeito é de com os dados das pessoas na entrada, ordenar e imprimir as idades todas ordenadas. Ajude o prefeito! (Atenção: são 1000 casos de teste!!).

**Entrada**

Para cada caso você deve ler um inteiro **N** (1 <= **N** <= 10^6), indicando o número de pessoas na cidade. Na linha seguinte há **N** números inteiros separados por espaço indicando as idades das pessoas (1 <= **idade** <= 50). Você deve ler até o **EOF.**

**Saída**

Imprima para cada caso de teste as idades das pessoas da cidade de forma ordenada. Imprima todas na mesma linha, após cada idade coloque um espaço em branco (MESMO PARA A ÚLTIMA IDADE!!).

|  |  |
| --- | --- |
| **Entrada de Teste**  3  3 2 1  5  1 50 50 50 1  1  1 | **Saída de Teste**  1 2 3  1 1 50 50 50  1 |

**Problema F: FormiguinhaZ**

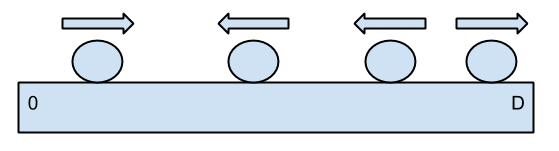
*TL: 1 segundo (~50 casos)*

*ML: 64 MB*

Em um tronco de árvore há algumas formigas. O tronco está deitado no chão na horizontal e tem tamanho **D** centímetros. Podemos representar como posição 0 a posição mais à esquerda do tronco e a posição **D** como a mais à direita.

Cada formiga está em uma posição distinta. As formigas estão andando em uma velocidade constante de um centímetro por segundo, ou para a direita ou para a esquerda. Quando as formigas se encontram no tronco ambas mudam sua direção e, caso elas ultrapassem os limites dos troncos, elas caem dele. Uma formiga caiu do tronco se e somente se sua posição for menor do que **0** ou maior do que **D**.

Dado o tamanho do tronco e as posições e direções iniciais das formigas, quanto tempo (em segundos) irá levar para que todas caiam? É garantido que isto vai acontecer em algum determinado momento.



**Entrada**

Há varios testes. A primeira linha de cada caso de teste contém os número **N** (1 <= **N** <= 10^5) e **D** (1 <= **D** <= 10^9), indicando o número de formigas e o tamanho do tronco respectivamente. Nas **N** linhas seguintes temos a descrição das formigas, indicando a posição e direção delas (**1** para andando inicialmente para esquerda ou **0** andando inicialmente para direita). Os valores caberão em um inteiro e serão válidos. Leia até o **EOF**.

**Saída**

Imprima o tempo levado para que todas as formigas caiam do tronco.

|  |  |
| --- | --- |
| **Entrada de Teste**  1 3  0 0  2 3  0 0  3 1  2 3  1 0  2 1 | **Saída de Teste**  3  3  2 |