

A. Distância Binária

time limit per test: 2 seconds
memory limit per test: 256 megabytes

Na Teoria da Informação, a distância de hamming entre duas strings de mesmo comprimento é o número de posições nas quais elas diferem entre si. De uma maneira simplificada, essa distância corresponde ao menor número de substituições necessárias para transformar uma string na outra. Por exemplo, considerando os caracteres de uma string, a distância de hamming para as palavras "cebola" e "cevada", pode ser calculada da seguinte forma:

```
c e b o l a
c e v a d a
0 1 2 3 4 5
  | | |
```

Ou seja, para transformar cebola em cevada, basta alterar, na palavra "cevada", as posições 2, 3 e 4. Assim, a distância de hamming para essas duas palavras é 3.

Essa distância é amplamente utilizada em computação, em áreas como teoria da informação, teoria da codificação, telecomunicações e criptografia. Em telecomunicações ela é utilizada para contar o número de bits corrompidos na transmissão de uma mensagem de um determinado comprimento. Neste caso, a distância é calculada a partir do alfabeto binário $\{0, 1\}$ (palavras compostas por zeros e uns). Por exemplo, os números binários 101 e 011 tem distância igual a dois porque é necessário alterar dois bits para transformar um no outro.

Assim, sua tarefa é dados dois números inteiros positivos, calcular a distância Hamming entre eles.

Input

A primeira linha contém um inteiro T ($1 \leq T \leq 100000$), que representa o número de casos de teste.

Cada uma das N linhas seguintes contém dois inteiros positivos, separados por espaço, na base decimal X e Y ($0 \leq X, Y \leq 2^{63}$), que representam os números cujo a distância de hamming deve ser calculada.

Output

A saída deve possuir N linhas, sendo uma para cada caso de teste contendo a distância hamming das representações binárias de X e Y .

Examples

input	Copy
1 6 7	
output	Copy
1	

input	Copy
2 8 23 15 8	
output	Copy
5 3	

Note

No primeiro caso de testes, os números 6 e 7 na base binária são 110 e 111, e apenas o bit menos significativo precisa ser modificado para transformar um em outro.

IDP - TAA - 2025/02

Private

Participant



→ About Group



Este grupo tem o objetivo de organizar as atividades de programação da disciplina de Técnicas de Programação e Análise de Algoritmos.

[Group website](#)

→ Group Contests

- TAA - LEA 05
- TAA - LEE 05
- TAA - LEA 04
- TAA - LEE 04
- TAA - AS 01
- TAA - LEA 03
- TAA - LEE 03
- TAA - LEA 02
- TAA - LEE 02
- TAA - LEA 01
- TAA - LEE 01
- ET - Exercício de Testes

TAA - LEE 02

Finished

Practice



→ Submit?

Language: GNU G++17 7.3.0

Choose
file:

Escolher Arquivo

Nenhu...colhido

Submit

→ Last submissions		
Submission	Time	Verdict
341115659	Sep/29/2025 23:11	Accepted
337459071	Sep/07/2025 21:14	Accepted

[Codeforces](#) (c) Copyright 2010-2025 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Nov/11/2025 18:19:22^{UTC-3} (j2).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#) | [Terms and Conditions](#)

Supported by

