
Hashing astut?**P78794_ca**Divuitè Concurs de Programació de la FME (2021-12-22)

Abans de plantejar aquest problema, us recordem (o informem) que:

Donats dos enters a i b , en C++ $(a \wedge b)$ és el XOR d' a i b . És a dir, n'és el "or exclusiu bit a bit": el resultat d'operar cada parell de bits és 1 si els dos bits són diferents. Per exemple, $(9 \wedge 12) = (1001_2 \wedge 1100_2) = 0101_2 = 5$.

Donats dos enters x i n , en C++ $(x \ll n)$ és el resultat de desplaçar x n bits cap a l'esquerra. Si no hi ha sobreiximents, dóna el resultat de multiplicar x per 2 elevat a n . Per exemple, $(9 \ll 2) = (1001_2 \ll 2) = 100100_2 = 36$.

Ara, el problema. L'Edgar ha descobert un nou algorisme de hashing al qual li veu un gran futur en aplicacions criptogràfiques. Semblant a tants altres mètodes, combina moltes operacions de bits, per la qual cosa podria ser complicat recuperar la informació original.

L'Edgar ha decidit que treballarà amb enters de $B = 30$ bits (és a dir, mòdul 2^B). Donats dos enters a i b , defineix:

$$h(a, b) = \left((a \wedge b) \ll 0 \right) \wedge \left((a \wedge b) \ll 1 \right) \wedge \dots \wedge \left((a \wedge b) \ll (B-1) \right) \% (1 \ll B)$$

L'Edgar ens assegura (i té raó) que donats a i $c = h(a, b)$, només hi ha una b possible. Ell creu que el seu mètode és prou segur, és a dir, que no es pot obtenir b eficientment a partir d' a i de c . Demostreu que l'Edgar és un *palomo*.

Entrada

L'entrada consisteix en diversos casos, cadascun amb a i c , dos enters entre 0 i $2^{30} - 1$.

Sortida

Per a cada cas, escriuiu la b corresponent.

Exemple d'entrada

```
23 42
100 20
0 0
123 456
1000000000 987654321
1073741823 1073741822
```

Exemple de sortida

```
105
88
0
547
888697811
1073741821
```

Informació del problema

Autor : Izan Beltrán

Generació : 2021-12-22 11:42:09

© Jutge.org, 2006–2021.

<https://jutge.org>