## Hashing astut? P78794_ca

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

Donats dos enters @a@ i @b@, en C++ `(@a@^@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 sii els dos bits són diferents. Per exemple, `(912)` $= (1001_2 \hat{} 1100_2) = 0101_2 =$ `5`.

Donats dos enters @x@ i @n@, en C++ `(@x@<<@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<<2)` $= (1001_2$`<<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 métodes, 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, módul $2^B$). Donats dos enters $a$ i $b$, defineix:

$$h(a,b) = \texttt{( ((}a \hat{} b\texttt{)<<0)^ ((}a \hat{} b\texttt{)<<1)^ .\hat{.}. ((}a \hat{} b\texttt{)<<(B-1)) ) \% (1<<B)}$$

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, escriviu la $b$ corresponent.

### Exemple d'entrada 1

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

### Exemple de sortida 1

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

### Informació del problema

Autoria: Izan Beltrán

Generació: 2026-01-25T12:00:52.089Z

© *Jutge.org*, 2006–2026.
https://jutge.org