E. Maxim and Matrix

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Maxim loves to fill in a matrix in a special manner. Here is a pseudocode of filling in a matrix of size $(m+1) \times (m+1)$:

Maxim asks you to count, how many numbers m $(1 \le m \le n)$ are there, such that the sum of values in the cells in the row number $m+1$ of the resulting matrix equals t.

Expression (x *xor* y) means applying the operation of bitwise excluding "OR" to numbers x and y. The given operation exists in all modern programming languages. For example, in languages *C++* and *Java* it is represented by character "^", in *Pascal* — by "`xor`".

Input

A single line contains two integers n and t $(1 \le n, t \le 10^{12}, t \le n+1)$.

Please, do not use the `%lld` specifier to read or write 64-bit integers in *C++*. It is preferred to use the `cin`, `cout` streams or the `%I64d` specifier.

Output

In a single line print a single integer — the answer to the problem.

Examples

input
1 1
output
1

input
3 2
output
1

input
3 3
output
0

input
1000000000000 1048576
output
118606527258