Problem H
Heap Structure

Time limit: 3 seconds
Memory limit: 1024 megabytes

Problem Description

A minimum binary heap, often simply referred to as a "min-heap," is a specialized type of binary tree-based data structure used in computer science. It is a binary tree that has two main properties:

1. Heap Property: In a min-heap, for any given node, the value of that node is less than or equal to the values of its child nodes. This means that the minimum value in the heap is always found at the root node. In other words, the smallest element in the heap is at the top.

2. Binary Tree Structure: A min-heap is typically a complete binary tree, which means that all levels of the tree are fully filled except possibly for the last level, which is filled from left to right. This structural property allows min-heaps to be efficiently implemented using arrays.

Min-heaps are often used to implement priority queues, which are data structures that maintain a collection of elements with associated priorities. By keeping the minimum element at the root, min-heaps can quickly retrieve and remove the element with the highest priority (lowest value) in logarithmic time. However, removing any other element from a min-heap may need linear time.

Hank recently learned min-heaps. He wonders how many nodes can keep the k-th smallest element in a min-heap of n-nodes. Please write a program to help him.

Input Format

The input contains two space-separated positive integers n and k.

Output Format

Output the number of nodes that can keep the k-th smallest element in a min-heap of n nodes.

Technical Specification

$1 \leq k \leq n \leq 10^{18}$

Sample Input 1

```
100 1
```

Sample Output 1

```
1
```

Sample Input 2

```
12 3
```

Sample Output 2

```
6
```