Given an integer n, you must transform it into 0 using the following operations any number of times:

* Change the rightmost (0th) bit in the binary representation of n.
* Change the ith bit in the binary representation of n if the (i-1)th bit is set to 1 and the (i-2)th through 0th bits are set to 0.

Return *the minimum number of operations to transform* n *into* 0*.*

**Example 1:**

Input: n = 3  
Output: 2  
Explanation: The binary representation of 3 is "11".  
"11" -> "01" with the 2nd operation since the 0th bit is 1.  
"01" -> "00" with the 1st operation.

**Example 2:**

Input: n = 6  
Output: 4  
Explanation: The binary representation of 6 is "110".  
"110" -> "010" with the 2nd operation since the 1st bit is 1 and 0th through 0th bits are 0.  
"010" -> "011" with the 1st operation.  
"011" -> "001" with the 2nd operation since the 0th bit is 1.  
"001" -> "000" with the 1st operation.

**Constraints:**

* 0 <= n <= 109