# 题目

给你一个正整数n，你可以执行下述操作任意次：

n加上或减去2的某个幂

返回使n等于0需要执行的最少操作数。

如果x == 2i且其中i >= 0，则数字x是2的幂。

示例 1：

输入：n = 39

输出：3

解释：我们可以执行下述操作：

- n 加上 20 = 1 ，得到 n = 40 。

- n 减去 23 = 8 ，得到 n = 32 。

- n 减去 25 = 32 ，得到 n = 0 。

可以证明使 n 等于 0 需要执行的最少操作数是 3 。

示例 2：

输入：n = 54

输出：3

解释：我们可以执行下述操作：

- n 加上 21 = 2 ，得到 n = 56 。

- n 加上 23 = 8 ，得到 n = 64 。

- n 减去 26 = 64 ，得到 n = 0 。

使 n 等于 0 需要执行的最少操作数是 3 。

提示：

1 <= n <= 105

# 分析

class Solution {

public:

int minOperations(int n) {

int cnt = 0;

while (n) {

int p = round(log2(n));

// 这里不关注是加还是减2的幂

n = abs(n - pow(2, p));

cnt++;

}

return cnt;

}

};