

位 1 的个数

编写一个函数，输入是一个无符号整数（以二进制串的形式），返回其二进制表达式中数字位数为 '1' 的个数（也被称为**汉明重量**）。

提示：

- 请注意，在某些语言（如 **Java**）中，没有无符号整数类型。在这种情况下，输入和输出都将被指定为有符号整数类型，并且不应影响您的实现，因为无论整数是有符号的还是无符号的，其内部的二进制表示形式都是相同的。
- 在 **Java** 中，编译器使用**二进制补码**记法来表示有符号整数。因此，在上面的**示例 3**中，输入表示有符号整数 **-3**。

示例 1:

输入：0000000000000000000000000000001011

输出: 3

解释：输入的二进制串 `000000000000000000000000000001011` 中，共有三位为 '1'。

示例 2:

输入：0000000000000000000000000000000001000000

输出：1

解释： 输入的二进制串 `000000000000000000000000010000000` 中，共有一位为 '1'。

示例 3:

输入: 11111111111111111111111111111111111101

输出： 31

解释： 输入的二进制串 **1111111111111111111111111111111101** 中，共有 31 位为 '1'。

```
class Solution {
public:
    int hammingWeight(uint32_t n) {
        int count=0;
        uint32_t flag=1;
        while(flag)
        {
```

```
        if(n&flag)
            count++;
        flag=flag<<1;
    }
    return count;
}
};
```