

/\*

A binary gap within a positive integer N is any maximal sequence of consecutive zeros that is surrounded by ones at both ends in the binary representation of N.

For example, number 9 has binary representation 1001 and contains a binary gap of length 2. The number 529 has binary representation 1000010001 and contains two binary gaps: one of length 4 and one of length 3. The number 20 has binary representation 10100 and contains one binary gap of length 1. The number 15 has binary representation 1111 and has no binary gaps.

给定一个正整数 N，找到并返回 N 的二进制表示中两个连续的 1 之间的最长距离。

如果没有两个连续的 1，返回 0。

示例 1：

输入：22

输出：2

解释：

22 的二进制是 0b10110。

在 22 的二进制表示中，有三个 1，组成两对连续的 1。

第一对连续的 1 中，两个 1 之间的距离为 2。

第二对连续的 1 中，两个 1 之间的距离为 1。

答案取两个距离之中最大的，也就是 2。

示例 2：

输入：5

输出：2

解释：

5 的二进制是 0b101。

示例 3：

输入：6

输出：1

解释：

6 的二进制是 0b110。

示例 4：

输入：8

输出：0

解释：

8 的二进制是 0b1000。

在 8 的二进制表示中没有连续的 1，所以返回 0。

提示：

$1 \leq N \leq 10^9$

来源：力扣 (LeetCode)

链接：<https://leetcode-cn.com/problems/binary-gap>

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

\*/

分析:

- 根据定义,需要遍历所有相邻的1的情况,并记录这其中的最大值返回(方法一)

方法一: C\_Solution

```
int binaryGap(int N)
{
    int arr[32] =
    {
        0x00000001,0x00000002,0x00000004,0x00000008,
        0x00000010,0x00000020,0x00000040,0x00000080,
        0x00000100,0x00000200,0x00000400,0x00000800,
        0x00001000,0x00002000,0x00004000,0x00008000,
        0x00010000,0x00020000,0x00040000,0x00080000,
        0x00100000,0x00200000,0x00400000,0x00800000,
        0x01000000,0x02000000,0x04000000,0x08000000,
        0x10000000,0x20000000,0x40000000,0x80000000
    };
    int index1    =  -1 ;
    int index2    =  -1 ;
    int i         =   0  ;
    int max_marge =   0  ;
    int temp_marge =  0  ;
    while(i<=31)
    {
        if(N&arr[i])
        {
            index1 = index2;
            index2 = i;
        }

        if( (index1!=-1)
            &&(index2!=-1)
        )
        {
            temp_marge = index2 - index1;
            if(temp_marge>max_marge)
            {
                max_marge = temp_marge;
            }
        }
        i++;
    }
    return max_marge;
}
```

```
    }  
    }  
    i++;  
}  
return max_marge;  
}
```

/\*

执行结果：

通过

[显示详情](#)

执行用时 :8 ms, 在所有 C 提交中击败了30.19%的用户

内存消耗 :6.7 MB, 在所有 C 提交中击败了88.00%的用户

\*/

---

AlimyBreak

2019.07.27