# 题目

给定一个正整数，检查他是否为交替位二进制数：换句话说，就是他的二进制数相邻的两个位数永不相等。

**示例 1:**

输入: 5

输出: True

解释:

5的二进制数是: 101

**示例 2:**

输入: 7

输出: False

解释:

7的二进制数是: 111

**示例 3:**

输入: 11

输出: False

解释:

11的二进制数是: 1011

**示例 4:**

输入: 10

输出: True

解释:

10的二进制数是: 1010

# 分析

## 方法一：位运算

class Solution {

public:

bool hasAlternatingBits(int n) {

n = (n ^ (n >> 1));

//如果是10交叉的情况,此时最后结果必为1111

return (n & ((long)n + 1)) == 0;

//1111 & (1111+1)=1111 & 0000=0000,其他非0

}

};