

693 Binary Number with Alternating Bits

2018年5月3日 16:43

交替位二进制数

Given a positive integer, check whether it has alternating bits: namely, if two adjacent bits will always have different values.

Example 1:

Input: 5

Output: True

Explanation:

The binary representation of 5 is: 101

Example 2:

Input: 7

Output: False

Explanation:

The binary representation of 7 is: 111.

Example 3:

Input: 11

Output: False

Explanation:

The binary representation of 11 is: 1011.

Example 4:

Input: 10

Output: True

Explanation:

The binary representation of 10 is: 1010.

来自 <<https://leetcode.com/problems/binary-number-with-alternating-bits/description/>>

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。

Solution for Python3:

```
1 class Solution1:
2     def hasAlternatingBits(self, n):
3         """
4         :type n: int
5         :rtype: bool
6         """
7         bits = bin(n)
8         return all(bits[i] != bits[i+1] for i in range(len(bits) - 1))
9
10 class Solution2:
11     def hasAlternatingBits(self, n):
12         """
13         :type n: int
14         :rtype: bool
15         """
16         n, cur = n >> 1, n & 1
17         # n, cur = divmod(n, 2)
```

```

18         while n:
19             if cur == n & 1:
20                 return False
21             n, cur = n >> 1, n & 1
22         return True
23
24 class Solution3:
25     def hasAlternatingBits(self, n):
26         """
27         :type n: int
28         :rtype: bool
29         """
30         return not ((n - (n >> 2)) & (n - (n >> 2) - 1))

```

Solution for C++:

```

1  class Solution1 {
2  public:
3      bool hasAlternatingBits(int n) {
4          int cur = n & 1;
5          n >>= 1;
6          while (n > 0) {
7              if (cur == (n & 1))
8                  return false;
9              cur = n & 1;
10             n >>= 1;
11         }
12         return true;
13     }
14 };
15
16 class Solution2 {
17 public:
18     bool hasAlternatingBits(int n) {
19         return !((n - (n >> 2)) & (n - (n >> 2) - 1));
20     }
21 };

```

Appendix:

Python divmod() 函数

- 1) python divmod() 函数把除数和余数运算结果结合起来，返回一个包含商和余数的元组(a // b, a % b)。
- 2) 数字的二进制表示时，bit位相邻不相等
 - a. 10101 >>2 101 相减 10000 是2的power
 - b. 01010 >>2 010 相减 01000是2的power

3) 判断一个数是否是2的power

$\text{not } (n \& (n-1))$ 若n是2的power,则n与n-1进行&运算得到0。