

```

1  /*
2  爱丽丝和鲍勃一起玩游戏，他们轮流行动。爱丽丝先手开局。
3  最初，黑板上有一个数字  $N$ 。在每个玩家的回合，玩家需要执行以下操作：
4      选出任一  $x$ ，满足  $0 < x < N$  且  $N \% x == 0$ 。
5      用  $N - x$  替换黑板上的数字  $N$ 。
6  如果玩家无法执行这些操作，就会输掉游戏。
7  只有在爱丽丝在游戏中取得胜利时才返回 True，否则返回 false。假设两个玩家都以最佳状态参与游戏。
8  示例 1:
9  输入: 2
10 输出: true
11 解释: 爱丽丝选择 1，鲍勃无法进行操作。
12 示例 2:
13 输入: 3
14 输出: false
15 解释: 爱丽丝选择 1，鲍勃也选择 1，然后爱丽丝无法进行操作。
16 提示:
17     1 <= N <= 1000
18 来源：力扣（LeetCode）
19 链接：https://leetcode-cn.com/problems/divisor-game
20 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。
21 */

```

分析:

- 根据游戏的规则进行推进, N 会慢慢减小,极限情况就是 $N = 2$ 或 $N = 1$,只要先手保证在自己拿的时候是偶数即可.
- 若当前 N 为偶数先手拿走 $x = 1$ 即可,若先手 N 为奇数,不管怎么拿 $N - x$ 一定为偶数,后手每次拿 $x = 1$ 即可.
- 所以"最佳状态"就是保证每次让自己遇到偶数,让对手遇到奇数.
- 对于先手来说:只要判断传入的 N 是否是偶数即可,是偶数有必胜法,是奇数则必输(遇到最佳状态的Bob).

方法一:C_常规检测 N 的奇偶性

```

1  bool divisorGame(int N)
2  {
3      return (N>0)&&(!(N&0x01));
4  }
5
6  /*
7  执行结果:
8  通过
9  显示详情
10 执行用时 :0 ms, 在所有 C 提交中击败了100.00% 的用户
11 内存消耗 :6.7 MB, 在所有 C 提交中击败了100.00%的用户
12 */

```

