1. 题目
2. 题干

爱丽丝和鲍勃一起玩游戏，他们轮流行动。爱丽丝先手开局。

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：

选出任一 x，满足 0 < x < N 且 N % x == 0 。

用 N - x 替换黑板上的数字 N 。

如果玩家无法执行这些操作，就会输掉游戏。

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。

1. 示例

示例 1：

输入：2

输出：true

解释：爱丽丝选择 1，鲍勃无法进行操作。

示例 2：

输入：3

输出：false

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。

二．题解

（一）思路

初看题目，这是啥玩意，简直是阅读理解题目。细细思考，这又是一个动态规划的问题，从小值开始不断地增长，记录从N=1开始，先手是否会成功。因为，不论Alice开始先手的N是多少，Bob得到的数一定是小于N的，那么通过判断Bob得到的数所对应先手成功与否，就能够直接判断出Alice是否会成功。如果Bob得到的数先手一定会失败，那么Alice一定会成功。

1. 代码实现

Java：

class Solution {

    public boolean divisorGame(int N) {

        boolean[] flag = new boolean[N+5];

        flag[1] = false;

        flag[2] = true;

        for(int i = 3; i <= N; ++i){

            for(int j = 1; j < i; ++j){

                if(i % j == 0 && !flag[i-j]){

                    flag[i] = true;

                    break;

                }

            }

        }

        return flag[N];

    }

}