1025. Divisor Game

Alice and Bob take turns playing a game, with Alice starting first.

Initially, there is a number `n` on the chalkboard. On each player's turn, that player makes a move consisting of:

- Choosing any `x` with `0 < x < n` and `n % x == 0`.
- Replacing the number `n` on the chalkboard with `n - x`.

Also, if a player cannot make a move, they lose the game.

Return `true` *if and only if Alice wins the game, assuming both players play optimally*.

Example 1:

Input: n = 2
Output: true
Explanation: Alice chooses 1, and Bob has no more moves.

Example 2:

Input: n = 3
Output: false
Explanation: Alice chooses 1, Bob chooses 1, and Alice has no more moves.

```
def divisorGame(self, n: int) -> bool:
        if n==1 or n==3:
            return False
        if n==2:
            return True
        dp = [False for i in range(n+1)]
        dp[1] = False
        dp[2] = True
        dp[3] = False

        for i in range(4,n+1):
            if i%2!=0:
                dp[i] = dp[i-1] and dp[i-2]
            else:
                dp[i] = True
```

```
        return dp[n]
        return (n & 1)==0 #Method 2
```

Optimal strategy is if N is even Alice always wins as she will subtract 1 from N and give Odd to Bob who will only be able to subtract an odd from

N-1. Now, again Alice will do the same until Alice reach N=2 where she wins.

If N is odd Alice will always loose as she too wont have an option of subtracting anything but an odd from N.