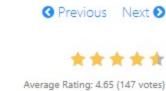
# 292. Nim Game 2

April 6, 2016 | 82.5K views



**6** 0 0

You are playing the following Nim Game with your friend: There is a heap of stones on the table, each time one of you take turns to remove 1 to 3 stones. The one who removes the last stone will be the winner. You will take the first turn to remove the stones.

Both of you are very clever and have optimal strategies for the game. Write a function to determine whether you can win the game given the number of stones in the heap.

#### Example:

```
Input: 4
Output: false
Explanation: If there are 4 stones in the heap, then you will never win the game;
             No matter 1, 2, or 3 stones you remove, the last stone will always be
             removed by your friend.
```

## Solution

You can always win a Nim game if the number of stones n in the pile is not divisible by 4.

#### Reasoning

Let us think of the small cases. It is clear that if there are only one, two, or three stones in the pile, and it is your turn, you can win the game by taking all of them. Like the problem description says, if there are exactly four stones in the pile, you will lose. Because no matter how many you take, you will leave some stones behind for your opponent to take and win the game. So in order to win, you have to ensure that you never reach the situation where there are exactly four stones on the pile on your turn.

Similarly, if there are five, six, or seven stones you can win by taking just enough to leave four stones for your opponent so that they lose. But if there are eight stones on the pile, you will inevitably lose, because regardless whether you pick one, two or three stones from the pile, your opponent can pick three, two or one stone to ensure that, again, four stones will be left to you on your turn.

It is obvious that the same pattern repeats itself for  $n=4,8,12,16,\ldots$ , basically all multiples of 4.

#### Java

```
public boolean canWinNim(int n) {
    return (n % 4 != 0);
}
```

#### Complexity Analysis

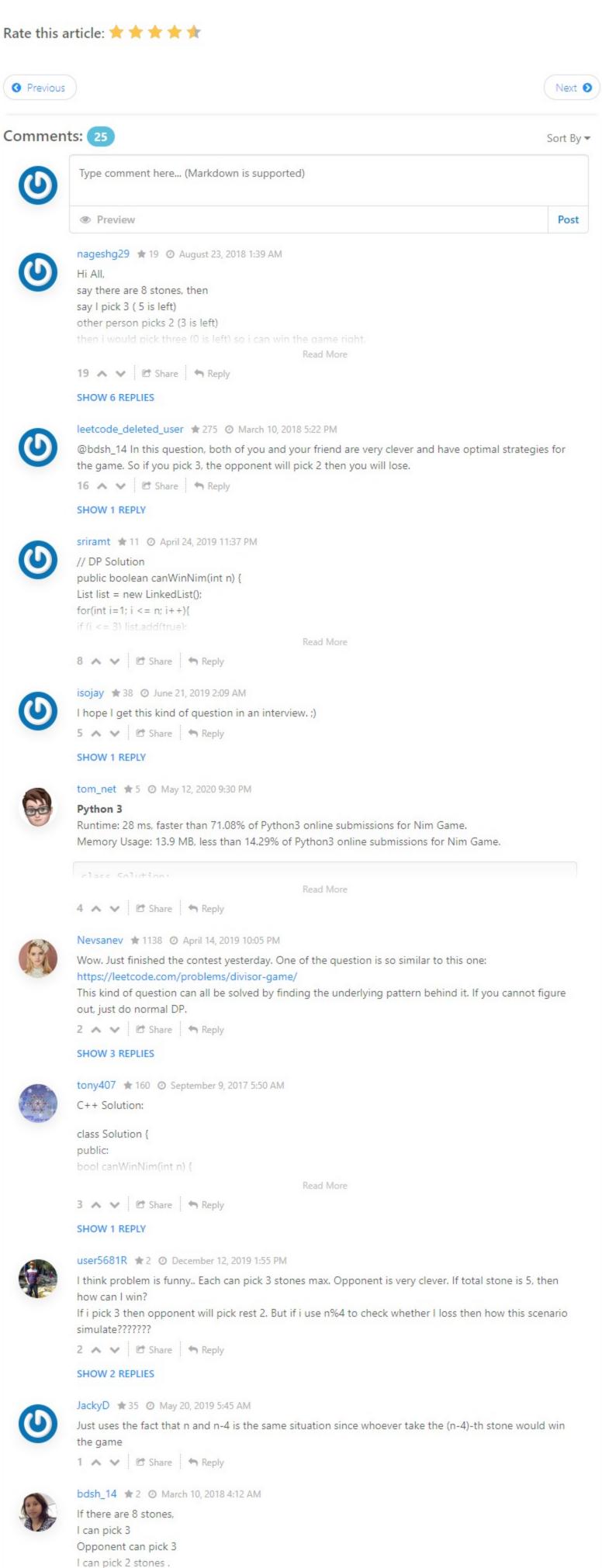
Time complexity is O(1), since only one check is performed. No additional space is used, so space complexity is also O(1).

### References

Lecture on Nim Games from University of Maryland: MATH 199: Math, Game Theory and the Theory of Games, Summer 2006.

Analysis written by: @noran





Read More



**SHOW 14 REPLIES** 

1 A V C Share Reply