The game of Nim

In the study of impartial combinatorical games, the simple game of Nim has a central role: it can be demonstrated that every impartial game is equivalent to some single Nim heap.

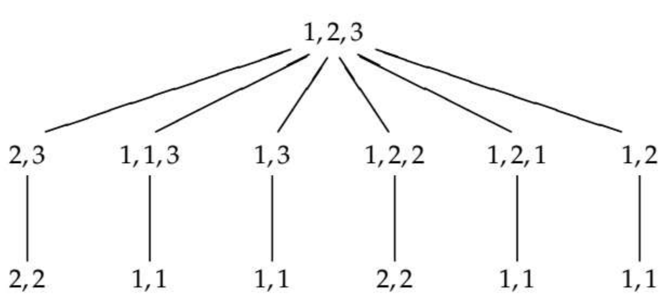
A Nim game is described by some number of "heaps", each containing some number of tokens. Players take turns removing as many tokens as they choose from one and only one heap. (In Poker Nim, players also may add tokens to a heap as their move during their turn.) We use the notation *n to denote a heap of size n, with options *0, *1, *2, ..., *(n-1). Note that *0 is the empty heap with no tokens.

Using the copycat principle, it is easy to see that two Nim heaps *n and *m are equivalent if and only n = m, that is, if they have the same number of tokens. For example, *5 and *3 are not equivalent because *5 + *3 is winning and *3 + *3 is losing.

Lemma 1.7. *Two Nim heaps are equivalent if and only if they have equal size.*

A general Nim position is a game sum of multiple Nim heaps, and can be represented as *a +*b +... Like all impartial combinatorical games, any general Nim game must be either losing or winning.

Consider the following Nim game *1 + *2 + *3, where it is Player I's turn to move. Note that this is a losing position, because all of its options are winning positions for Player II. For any move Player I may make, Player II has a winning move that creates two equal-sized heaps, which is a losing position by the copycat principle.



The winning strategy for Nim (C. Bouton, 1902)

Every Nim game G, no matter how many heaps, is equivalent to a single Nim heap of a particular size. This size is called the **Nim value** of G, and it fully describes whether G is winning or losing, as well as the winning move in G, if such a move exists.

To find the Nim value of a game with multiple heaps, calculate the **Nim sum** of the sizes of these heaps by writing each Nim heap's size in binary and adding the binary digits separately, modulo 2. More specifically:

<u>1. Represent every Nim heap *n as an equivalent game sum of smaller heaps, each with size of a distinct power of two.</u>

This decomposition is analogous to the binary representation of the integer n, by the following theorem:

Theorem 1.15. *Let* $n \geq 1$, *and* $n = 2^a + 2^b + 2^c + \cdots$, *where* $a > b > c > \cdots \geq 0$. *Then*

$$*n \equiv *(2^a) + *(2^b) + *(2^c) + \cdots . \tag{1.18}$$

(Note that every nonnegative integer n can be written uniquely as the sum of distinct powers of 2.)

<u>2. Cancel out repeated powers of two in pairs so that only distinct powers of two are left.</u>

This amounts to disposing of losing games (which are equivalent to the 0 game).

<u>3. Perform addition of the remaining numbers to find the Nim sum.</u>

A heap of this size is equivalent to the original game because adding this heap will create a losing game. (Note that the Nim sum of two numbers never exceeds their ordinary sum; if both numbers are less than some power of 2, then so is their Nim sum.)

If the Nim sum is 0, then the game is in a losing position. If the Nim sum is not 0, then the game is in a winning position, and the winning move must create a game that has a Nim sum of 0.

It is easier to see how to perform this winning move by writing the binary representations in columns, where the winning move is to flip the "bits" so that the binary sum is zero.

<u>This is best illustrated with an example.</u>

Example

Three Nim heaps $*5$, $*11$, $*8$.

(1) $*5 \equiv *4 + *1$, $*11 \equiv *8 + *2 + *1$, $*8 \equiv *8$.

(2) $*8$ and $*1$ occur an even number of times and cancel.
$*4$ and $*2$ occur an odd number of times and remain.

(3) $s = 6 = 4 + 2$. We have $*5 + *11 + *8 \equiv *s$.

(4) $*5 + *s \equiv *4 + *1 + *4 + *2 \equiv *1 + *2 \equiv *3$.
The move from $*5$ to $*3$ is winning, creating the position
$*3 + *11 + *8 \equiv *5 + *s + *11 + *8 \equiv *0$.

(4') $*11 + *s \equiv *8 + *2 + *1 + *4 + *2 \equiv *8 + *4 + *1 \equiv *13$.
Moving from $*11$ to $*13$ would be winning but is not allowed.

(4'') $*8 + *s \equiv *8 + *4 + *2 \equiv *14$.
Moving from $*8$ to $*14$ would be winning but is not allowed.

Using the binary system

heap	8	4	2	1
5	0	1	0	1
11	1	0	1	1
8	1	0	0	0
$s = 6$	0	1	1	0

The winning move: flip the '1' bits of the Nim sum s

	heap	8	4	2	1
→	3	0	0	1	1
	11	1	0	1	1
	8	1	0	0	0
	$s = 0$	0	0	0	0

The position s=0 is losing because a player must flip at least one "bit", resulting in at least one column with an odd number of 1's and creating a winning Nim position.

Extensions to other impartial games, and the mex rule

Recall that every impartial game is equivalent to a Nim heap. One method for discovering the Nim value of any impartial game is the **mex rule**, which states that a game's Nim value is the smallest integer not contained in the Nim values of the game's options.

The *mex rule* says that if the options of a game G are equivalent to Nim heaps with sizes from a set S (like $S = \{0,1,2,5,9\}$ above), then G is equivalent to a Nim heap of size m, where m is the *smallest non-negative integer not contained in* S. This number m is written $\operatorname{mex}(S)$, where *mex* stands for *minimum excluded number*. That is,

$$m = \operatorname{mex}(S) = \min\{k \geq 0 \mid k \notin S\}. \tag{1.17}$$

For example, $\operatorname{mex}(\{0,1,2,3,5,6\}) = 4$, $\operatorname{mex}(\{1,2,3,4,5\}) = 0$, and $\operatorname{mex}(\emptyset) = 0$.

Theorem 1.14 (The mex rule). *Consider an impartial game G. Then G is equivalent to a Nim heap of size m, where m is uniquely determined as follows: For each option H of G, let H be equivalent to a Nim heap of size s_H, and let $S = \{s_H \mid H \text{ is an option of } G\}$. Then $m = \operatorname{mex}(S)$, that is, $G \equiv *(\operatorname{mex}(S))$.*

Note that mex(S) = 0 (that is, the Nim value is 0) implies that G is a losing game; indeed, mex(S) only if all options of G are winning positions. As in the winning strategy for Nim, if the mex(S) is not 0 (that is, the Nim value is not 0), then the game is in a winning position, and the winning move must create a game with options S' such that mex(S') = 0.

The mex rule, as it ought to, also applies trivially to single Nim heaps, whose options are the set of non-negative integers smaller than their size:

$$\text{options of } *n: \quad *0,\ *1,\ *2, \ldots, *(n-1). \tag{1.16}$$

The following are several examples that use the mex rule to analyze various impartial games:

<u>A Rook-move game</u>

In the Rook-move game, players take turns moving the rook either up or left, for however many squares they choose. The player trapped at the upper-leftmost square loses.

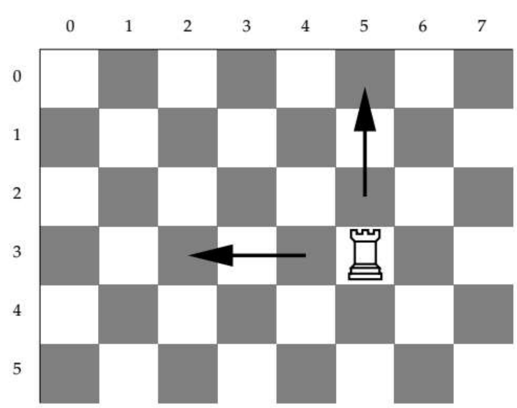


Figure 1.1 Rook-move game, where the player may move the rook on the Chess board in the direction of the arrows.

	0	1	2	3	4	5	6	7	8	9	10
0	0	*1	*2	*3	*4	*5	*6	*7	*8	*9	*10
1	*1	0	*3	*2	*5	*4	*7	*6	*9	*8	*11
2	*2	*3	0	*1	*6	*7	*4	*5	*10	*11	*8
3	*3	*2	*1	0	*7	*6	*5	*4	*11	*10	*9
4	*4	*5	*6	*7	0	*1	*2	*3	*12	*13	*14
5	*5	*4	*7	*6	*1	0	*3	*2	*13	*12	*15
6	*6	*7	*4	*5	*2	*3	0	*1	*14	*15	*12
7	*7	*6	*5	*4	*3	*2	*1	0	*15	*14	*13
8	*8	*9	*10	*11	*12	*13	*14	*15	0	*1	*2
9	*9	*8	*11	*10	*13	*12	*15	*14	*1	0	*3
10	*10	*11	*8	*9	*14	*15	*12	*13	*2	*3	0

Figure 1.2 Equivalent Nim heaps $*n$ for positions of the Rook-move game.

A Queen-move game + a Nim heap

The following game is a game sum of the Queen-move game and a single Nim heap of size 4.

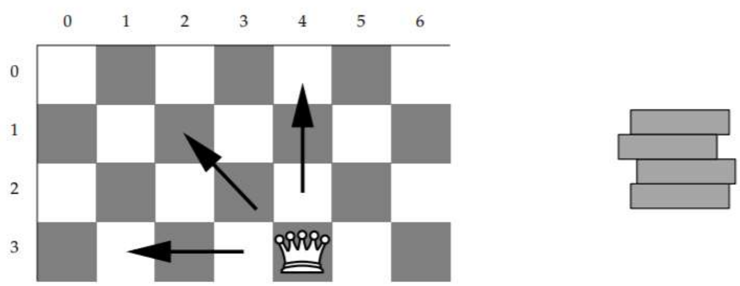


Figure 1.3 Game sum of a *Queen-move* game and a Nim heap. The player may *either* move the queen in the direction of the arrows *or* take some of the 4 tokens from the heap.

	0	1	2	3	4	5	6
0	0	*1	*2	*3	*4		
1	*1	*2	0	*4	*5		
2	*2	0	*1	*5	*3		
3	*3	*4	*5	*6	*2		

Figure 1.4 Equivalent Nim heaps for positions of the Queen-move game.

In this game, the Queen is at the position equivalent to *2. A winning move is to remove two tokens from the Nim heap, creating the game sum *2 + *2, and another is to move the Queen to 0,4 or 3,1, creating the game sum *4+*4.