Analyzing ELLIE - the Story of a Combinatorial Game

S. Heubach[1] P. Chinn[2] M. Dufour[3] G. E. Stevens[4]

[1]Dept. of Mathematics, California State Univ. Los Angeles

[2]Dept. of Mathematics, Humboldt State University

[3]Dept. of Mathematics, Université du Quebeq à Montréal

[4]Dept. of Mathematics, Hartwick College

October 23, 2007
Math Colloquium, Humboldt State University, Arcata

Overview

Overview

Overview

Overview

How ELLIE was conceived

- P. Chinn, R. Grimaldi, and S. Heubach, Tiling with Ls and Squares, Journal of Integer Sequences, Vol 10 (2007)
- Phyllis and Silvia talk to Gary - the idea of a game is born
- Matthieu joins in and brings background in combinatorial games

How ELLIE was conceived

- P. Chinn, R. Grimaldi, and S. Heubach, Tiling with Ls and Squares, Journal of Integer Sequences, Vol 10 (2007)
- Phyllis and Silvia talk to Gary - the idea of a game is born
- Matthieu joins in and brings background in combinatorial games

How ELLIE was conceived

- P. Chinn, R. Grimaldi, and S. Heubach, Tiling with Ls and Squares, Journal of Integer Sequences, Vol 10 (2007)
- Phyllis and Silvia talk to Gary - the idea of a game is born
- Matthieu joins in and brings background in combinatorial games

Description of ELLIE

ELLIE is played on a rectangular board of size m-by-n. Players alternately place L-shaped tiles of area 3. Last player to move wins (normal play).

Questions:

- For which values of n and m is there a winning strategy for player 1?
- What is the winning strategy?

Combinatorial Games

Definition

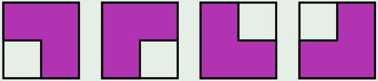
An*impartial combinatorial game* has the following properties:

- no randomness (dice, spinners) is involved, i.e., each player has complete information about the game and the potential moves
- each player has the same moves available at each point in the game (as opposed to chess, where there are white and black pieces). This condition makes the game impartial.

Working out small examples

Example (The 2×2 board)

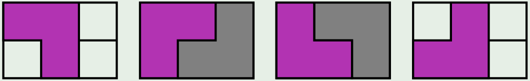
First player obviously wins, since only one L can be placed. In each case, the second player only finds one square left, which does not allow for placement of an L.



Working out small examples

Example (The 2×3 board)

First player's move is purple, second player's move is gray.

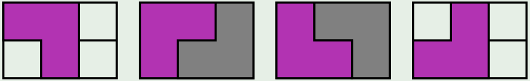


Note that for this board, the outcome (winning or losing) for the first player depends on that player's move. If s/he is smart, s/he makes the first or fourth move. This means that player I has a winning strategy.

Working out small examples

Example (The 2×3 board)

First player's move is purple, second player's move is gray.



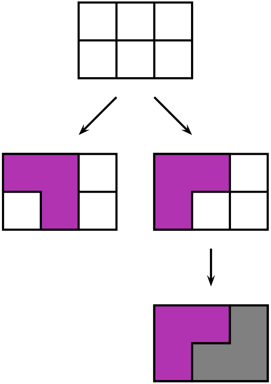
Note that for this board, the outcome (winning or losing) for the first player depends on that player's move. If s/he is smart, s/he makes the first or fourth move. This means that player I has a winning strategy.

Game trees

Definition

The game tree has the possible positions of the game as its nodes. The offspring of each position are the options, i.e., the positions that can arise from the next player's move. Usually we omit options that arise through symmetry.

Game tree for 2 × 3 board



Impartial Games

Definition

A position is a $\mathcal{P}$-position for the player about to make a move if the $\mathcal{P}$revious player can force a win (that is, the player about to make a move is in a losing position). The position is a $\mathcal{N}$-position if the $\mathcal{N}$ext player (the player about to make a move) can force a win.

For impartial games, there are only two outcome classes for any position, namely $\mathcal{N}$ (winning position) and $\mathcal{P}$ (losing position).

Recursive labeling

We label any node for which all options have been labeled as follows:

- Leafs of the game tree are always $\mathcal{P}$ positions.
- If a position has at least one option that is a $\mathcal{P}$ position then it should be labeled $\mathcal{N}$
- If all options of a position are labeled $\mathcal{N}$ then it should be labeled $\mathcal{P}$

The label of the empty board then tells whether Player I or Player II has a winning strategy.

Recursive labeling

We label any node for which all options have been labeled as follows:

- Leafs of the game tree are always $\mathcal{P}$ positions.
- If a position has at least one option that is a $\mathcal{P}$ position then it should be labeled $\mathcal{N}$
- If all options of a position are labeled $\mathcal{N}$ then it should be labeled $\mathcal{P}$

The label of the empty board then tells whether Player I or Player II has a winning strategy.

Recursive labeling

We label any node for which all options have been labeled as follows:

- Leafs of the game tree are always $\mathcal{P}$ positions.
- If a position has at least one option that is a $\mathcal{P}$ position then it should be labeled $\mathcal{N}$
- If all options of a position are labeled $\mathcal{N}$ then it should be labeled $\mathcal{P}$

The label of the empty board then tells whether Player I or Player II has a winning strategy.

Recursive labeling

We label any node for which all options have been labeled as follows:

- Leafs of the game tree are always $\mathcal{P}$ positions.
- If a position has at least one option that is a $\mathcal{P}$ position then it should be labeled $\mathcal{N}$
- If all options of a position are labeled $\mathcal{N}$ then it should be labeled $\mathcal{P}$

The label of the empty board then tells whether Player I or Player II has a winning strategy.

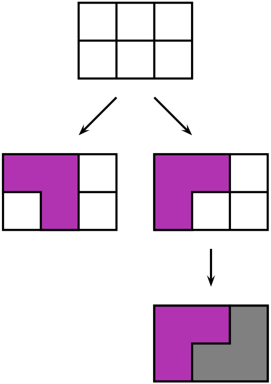
Recursive labeling

We label any node for which all options have been labeled as follows:

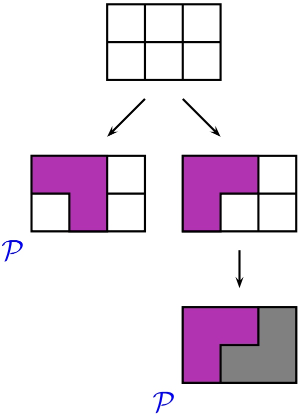
- Leafs of the game tree are always $\mathcal{P}$ positions.
- If a position has at least one option that is a $\mathcal{P}$ position then it should be labeled $\mathcal{N}$
- If all options of a position are labeled $\mathcal{N}$ then it should be labeled $\mathcal{P}$

The label of the empty board then tells whether Player I or Player II has a winning strategy.

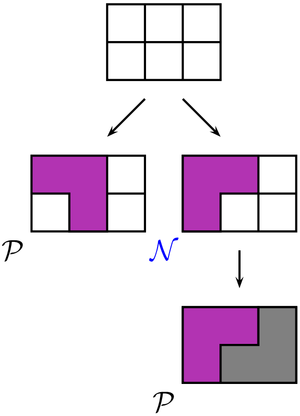
Labeling the game tree for 2 $\times$ 3 board



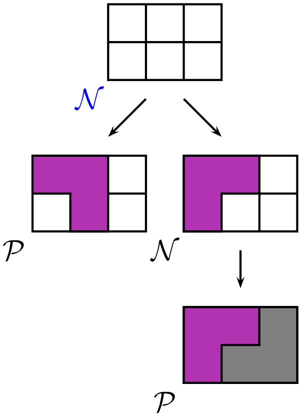
Labeling the game tree for 2 $\times$ 3 board



Labeling the game tree for 2 × 3 board



Labeling the game tree for 2 × 3 board



Nim

Definition

The game of Nim is played with heaps of counters. A move in the game of Nim consists of choosing one heap, and then removing any number of counters from that heap. A Nim game with heaps of size a, b, ..., k is denoted by $\text{Nim}(a, b, \ldots, k)$.

Why is this relevant?

Theorem

Every impartial game is just a bogus Nim-heap, i.e., the game can be translated into sum of Nim games.

Nim

Definition

The game of Nim is played with heaps of counters. A move in the game of Nim consists of choosing one heap, and then removing any number of counters from that heap. A Nim game with heaps of size a, b, ..., k is denoted by $\text{Nim}(a, b, \ldots, k)$.

Why is this relevant?

Theorem

Every impartial game is just a bogus Nim-heap, i.e., the game can be translated into sum of Nim games.

Nim-sum and Mex

Definition

The *nim-sum* of numbers a, b, ..., k, written as $a \oplus b \oplus \cdots \oplus k$ is obtained by adding the numbers in binary without carrying. This operation is also called *digital sum* or *exclusive or* (*xor* for short).

Definition

The *minimum excluded value* or *mex* of a set of non-negative integers is the least non-negative integer which does not occur in the set. This is denoted by $\mathrm{mex}\{a, b, c, \ldots, k\}$.

Nim-sum and mex

Example

The nim-sum $12 \oplus 13 \oplus 7$ equals 6:

12	1	1	0	0
13	1	1	0	1
7		1	1	1
	0	1	1	0

Example

$$\text{mex}\{1,4,5,7\} = 0$$
$$\text{mex}\{0,1,2,6\} = 3$$

Nim-sum and mex

Example

The nim-sum $12 \oplus 13 \oplus 7$ equals 6:

12	1	1	0	0
13	1	1	0	1
7		1	1	1
	0	1	1	0

Example

$$\text{mex}\{1,4,5,7\} = 0$$
$$\text{mex}\{0,1,2,6\} = 3$$

Nim-sum and mex

Example

The nim-sum $12 \oplus 13 \oplus 7$ equals 6:

12	1	1	0	0
13	1	1	0	1
7		1	1	1
	0	1	1	0

Example

$$\text{mex}\{1, 4, 5, 7\} = 0$$
$$\text{mex}\{0, 1, 2, 6\} = 3$$

Nim-sum and mex

Example

The nim-sum $12 \oplus 13 \oplus 7$ equals 6:

12	1	1	0	0
13	1	1	0	1
7		1	1	1
	0	1	1	0

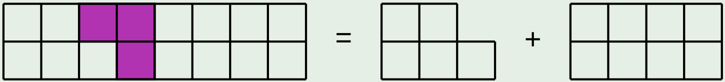
Example

$$\text{mex}\{1,4,5,7\} = 0$$
$$\text{mex}\{0,1,2,6\} = 3$$

Sums of Games

A game is the sum of games if the game board splits into the smaller sub-boards.

Example



How to compute the Grundy Function

Theorem

The Grundy-value or nim-value $\mathcal{G}(G)$ of a game G equals the size of the nim-heap to which the game G is equivalent. In particular, G is in the class $\mathcal{P}$ if and only if $\mathcal{G}(G) = 0$.

In particular,

Theorem

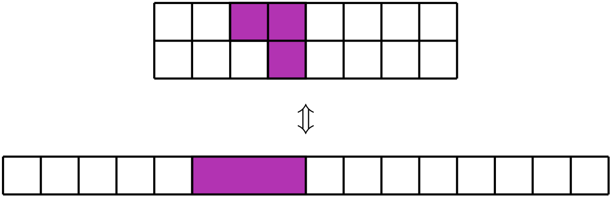
For any impartial games G, H, and J,

- $\mathcal{G}(G) = \mathrm{mex}\{\mathcal{G}(\mathrm{H}) | \mathrm{H}$ *is an option of* $\mathcal{G}\}$.
- $G = H + J$ *if and only if* $\mathcal{G}(G) = \mathcal{G}(H) \oplus \mathcal{G}(J)$.

What does this all mean?

- For any given game tree we can recursively label the positions with their Grundy value, then read off the value for the starting board.
- If we can translate the game into its equivalent Nim game, then we can actually produce a winning strategy (namely the one of the corresponding Nim game).
- We want to find a general rule explaining how a game breaks into smaller games so we can have a computer compute the Grundy function.

Nim-game equivalent of Ellie

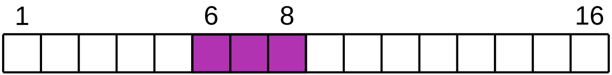


$2 \times n$ board for Ellie $\iff$ $1 \times (2n)$ Nim heap

Only the number of squares matters, not the geometry!

Recursion for Grundy function

- Play at position i splits a $1 \times n$ board into two sub-boards of sizes $1 \times (i-1)$ and $1 \times (n-i-2)$



- G_n denotes the $1 \times n$ board; $G(n, i)$ denotes the option of taking 3 counters at position i
- $\mathcal{G}(G(n,i)) = \mathcal{G}(G_{i-1}) \oplus \mathcal{G}(G_{n-i-2})$
- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(\mathrm{i}-1) \oplus \mathcal{G}(\mathrm{n}-\mathrm{i}-2) | 1 \leq \mathrm{i} \leq \lfloor \frac{\mathrm{n}}{2} \rfloor\}$

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(i-1) \oplus \mathcal{G}(n-i-2) | 1 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(i-1) \oplus \mathcal{G}(n-i-2) | 1 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(\mathrm{i}-1) \oplus \mathcal{G}(\mathrm{n}-\mathrm{i}-2) | 1 \leq \mathrm{i} \leq \lfloor \frac{\mathrm{n}}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(i-1) \oplus \mathcal{G}(n-i-2) | 1 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(\mathrm{i}-1) \oplus \mathcal{G}(\mathrm{n}-\mathrm{i}-2) | 1 \leq \mathrm{i} \leq \lfloor \frac{\mathrm{n}}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(i-1) \oplus \mathcal{G}(n-i-2) | 1 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(i-1) \oplus \mathcal{G}(n-i-2) | 1 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(i-1) \oplus \mathcal{G}(n-i-2) | 1 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Values for Grundy function

Let's compute the first 10 or so values of the Grundy function

- $\mathcal{G}(G_0) = \mathcal{G}(G_1) = \mathcal{G}(G_2) = 0$
- $\mathcal{G}(G_n) = \mathrm{mex}\{\mathcal{G}(i-1) \oplus \mathcal{G}(n-i-2) | 1 \leq i \leq \lfloor \frac{n}{2} \rfloor\}$

n	0	1	2	3	4	5	6	7	8	9	10
$\mathcal{G}(n) := \mathcal{G}(G_n)$	0	0	0	1	1	1	2	2	0	3	3

Mathematica Code

Questions to be answered:
Is the sequence of Grundy values $\mathcal{G}(G_n)$ periodic? Ultimately periodic?

- Need function to compute Digital Sum
- Need function to compute Mex
- Need function to compute Grundy value
- Need function to detect cycles

Values of $\mathcal{G}(n)$

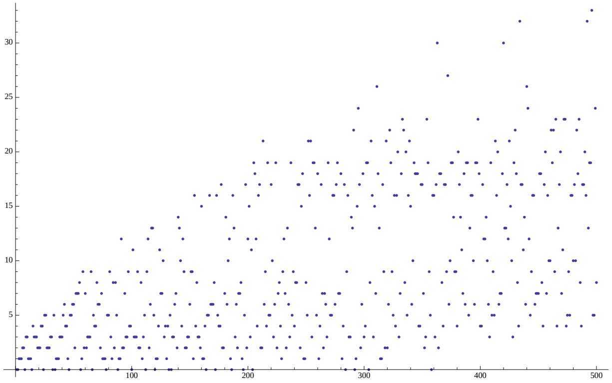


Figure: The first 500 values of $\mathcal{G}(n)$

Frequencies of $\mathcal{G}(n)$

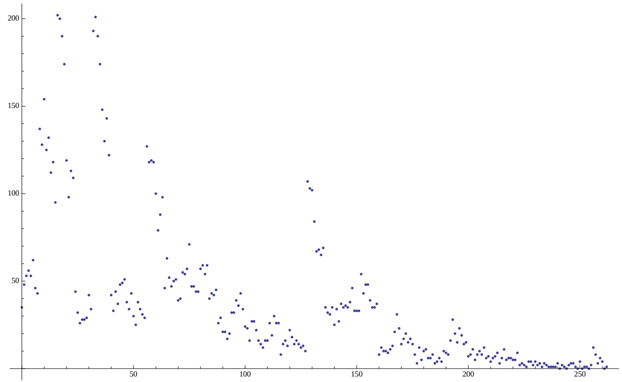


Figure: 10000 values of $\mathcal{G}(n)$; max val = 262; max freq = 202

Frequencies of $\mathcal{G}(n)$

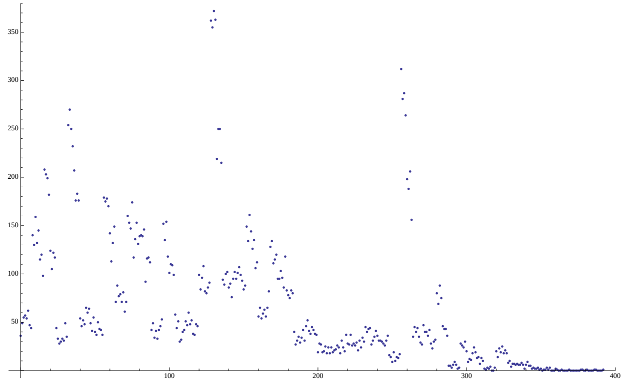


Figure: 25000 values of $\mathcal{G}(n)$; max val = 392; max freq = 372

Octal Games

Definition

An *octal game* is a 'take-and-break' game with Guy-Smith code $.d_1 d_2 d_3 \ldots$. A typical move consists of choosing one heap and removing i counters from the heap, then rearranging the remaining counters into some allowed number of new heaps. The code $.d_1 d_2 d_3 \ldots$ (whose digits range from 0 to 7) describes the allowed moves in the game:

- If $d_i \neq 0$, then an allowed move is to take i counters from a Nim heap.
- Writing $d_i \neq 0$ in base 2 then shows how the i counters may be taken: If $d_i = c_2 \cdot 2^2 + c_1 \cdot 2^1 + c_0 \cdot 2^0$, then removal of the i counters may ($c_j = 1$) or may not ($c_j = 0$) leave j heaps.

Octal Games

Example

The octal game **.17** allows us to take either 1 or 2 counters.

- $d_1 = 1 = 0 \cdot 2^2 + 0 \cdot 2^1 + \mathbf{1} \cdot 2^0$, therefore we are allowed to leave zero heaps when taking one counter, i.e., we can take away a heap that consists of a single counter.
- $d_2 = 7 = \mathbf{1} \cdot 2^2 + \mathbf{1} \cdot 2^1 + \mathbf{1} \cdot 2^0$, therefore we are allowed to leave either two, one or no heaps when taking two counters, i.e., we can take away a heap that consists of two counters, we can remove two counters from the top of a heap (leaving one heap), or can take two counters and split the remaining heap into two non-zero heaps.

Ellie = ?

Since we can only take three counters at a time, $d_i = 0$ for $i \neq 3$. When we place a tile, it can be

- at the end (leaving one heap),
- in the middle of the board (leaving two heaps), or
- covering the last three squares, leaving zero heaps.

$\Rightarrow$ **Ellie =**.007

Ellie = ?

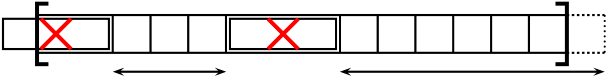
Since we can only take three counters at a time, $d_i = 0$ for $i \neq 3$. When we place a tile, it can be

- at the end (leaving one heap),
- in the middle of the board (leaving two heaps), or
- covering the last three squares, leaving zero heaps.

$\Rightarrow$ **Ellie =.007**

Treblecross = .007

- Treblecross is Tic-Tac-Toe played on a $1 \times n$ board in which both players use the same symbol, X. The first one to get three X's in a row wins.
- Don't want to place an X next to or next but one to an existing X, otherwise opponent wins immediately
- If only considering sensible moves, one can think of each X as also occupying its two neighbors



What is known about .007

- No complete analysis
- $\mathcal{G}(G_n)$ computed up to $n = 2^{21} = 2,097,152$
- Maximum nim-value in that range is $\mathcal{G}(1,683,655) = 1,314$
- Last new nim-value to occur is $\mathcal{G}(1,686,918) = 1,237$
- Most frequent value is 1024, which occurs 63,506 times
- Second most frequent value is 1026, which occurs 62,178 times
- 37 $\mathcal{P}$ positions: 0, 1, 2, 8, 14, 24, 32, 34, 46, 56, 66, 78, 88, 100, 112, 120, 132, 134, 164, 172, 186, 196, 204, 284, 292, 304, 358, 1048, 2504, 2754, 2914, 3054, 3078, 7252, 7358, 7868, 16170

What is known about .007

Table: Smallest value of n for which $\mathcal{G}(n) = m$

m	n
1	3
2	6
4	15
8	55
16	154
32	434
64	1320
128	3217
256	9168
512	35662
1024	109362

For Further Reading I

- Elwyn R. Berlekamp, John H. Conway and Richard K. Guy.
  Winning Ways for Your Mathematical Plays, Vol 1 & 2.
  Academic Press, London, 1982.
- Michael H. Albert, Richard J. Nowakowski, and David Wolfe.
  Lessons in Play.
  AK Peters, 2007.
- I. Caines, C. Gates, R.K. Guy, and R. J. Nowakowski.
  Periods in Taking and Splitting Games.
  American Mathematical Monthly, April:359–361, 1999.

For Further Reading II

A. Gangolli and T. Plambeck.
A Note on periodicity in Some Octal Games.
International Journal of Game Theory, 18:311–320, 1989.