

A New Take-Away Game

by

Jim Propp*

Department of Mathematics

University of California

Berkeley, California

1 Introduction

Two children take turns stealing cookies from a larder, each removing a single cookie every other day. Some of the cookies may go bad while on the shelf and become unsafe to eat, but fortunately each cookie has an expiration date written on it in frosting. The goal of each child is *not* to maximize the number of cookies he or she eats, but rather to have the spiteful pleasure of getting the last cookie.

If none of the cookies goes bad during the course of play, then of course the game will be a dull one, as the outcome depends only on whether the number of cookies present at the start was even or odd. But if the initial provisioning is such that some of the cookies might go bad during the course of play, then who gets the last cookie (thereby winning the game) may depend on who eats which cookie when.

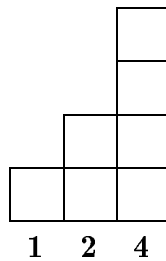
We may represent each cookie by a heap of n counters, where n is the number of days remaining before the cookie goes bad. A heap of size 0 represents a cookie whose time has expired; we assume that the two children are unwilling to risk food-poisoning and will not eat a possibly spoiled cookie.

*Partially supported by an NSF Postdoctoral Fellowship during the writing of this article.

On any given turn, a player removes *one* of the non-empty heaps and diminishes all the others by 1 counter; this corresponds to eating one of the cookies and noting that the rest have aged by a single day. When no counters remain, the player who just moved is deemed the winner. Let us represent each position schematically by a list of boldface numbers arranged in non-descending order (called the *components* of the position), each of which gives the size of a corresponding heap. Thus **1 2 4** signifies a three-heap position with heaps of size 1, 2, and 4. In this representation, a sample game would appear as:

1	2	4	
	↓		1st player removes the 2-heap;
0		3	
		↓	2nd player removes the 3-heap;
0			2nd player wins.

Though the heap-notation is probably the most convenient form in which to display cookie-game positions, there is another interesting representation: one can represent a heap of n counters graphically by a stack of n squares, and stick all the stacks together in a row, with the shorter stacks on the left and the taller stacks on the right. Thus, the position **1 2 4** would be depicted



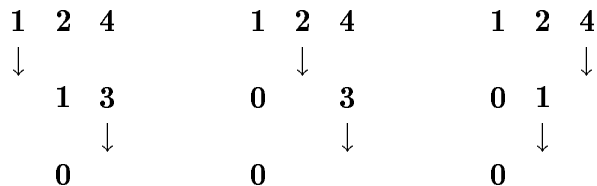
A legal move in this box-diagram representation is to cross out one of the columns and then to cross out the lowest row. Players continue in alternation until all the squares have been crossed out.

Jonathan King has made the interesting suggestion of regarding the *rows* of this diagram as the “heaps”, rather than the columns. Then we are effectively playing a heap-game in which a legal move consists of first removing a single counter from every heap that equals or exceeds a certain size (which may be chosen by the player at will, provided there is at least one heap of that size) and then removing the largest heap in its entirety (unless no heaps remain).

Lastly, John Conway has a way of viewing the game as a game of pure passing, in which there are no game pieces and no legal moves! Under normal play such a game would end as soon as it started, if not sooner, but in our modified scenario the players jointly own a supply of permits, each of which entitles the bearer to skip one turn. A permit can be used only once, and what is more each has an expiration date, so that if it is not used on or before that date it cannot be used at all. (Conway calls a permit that entitles the bearer to skip a turn anywhere up until the n th move of the game an “ n -day pass”.) Players take turns using these passes until no usable passes remain on the table. At this point, the player whose turn it is to move, no longer having the option of passing, loses the game.

2 The Reduction Theorems

Since a game that starts with n heaps must terminate in finitely many (indeed, at most n) moves, one of the two players must have a winning strategy. For instance, the position **1** is a clear win for the first player. On the other hand, note that from the initial position **1 2 4** the second player can win regardless of the first player’s opening move:



Following tradition [1], we call positions like **1**, in which the player about to move has a winning strategy, \mathcal{N} -positions (wins for the Next player). If a game is not an \mathcal{N} -position, then it is a win for the second player; such positions are called \mathcal{P} -positions (wins for the Previous player). We will call the \mathcal{N} -versus- \mathcal{P} character of the position the *type* of the position. Thus, **1 2 4** has type \mathcal{P} . Note that a position is of type \mathcal{N} if and only if there is a move from it to a position of type \mathcal{P} . If one could efficiently determine whether an arbitrary position was of type \mathcal{N} or type \mathcal{P} , one would have an effective winning strategy for one of the two players, since the rule “Always move to a \mathcal{P} -position if possible” can never lead a player astray.

If two positions G and H are of the same type (that is, if both are \mathcal{P} -positions or both are \mathcal{N} -positions), then we will write “ $G \sim H$ ”. For instance, it is easy to see in advance that

$$\mathbf{1\ 5\ 6\ 7} \sim \mathbf{1\ 4\ 4\ 4} \tag{1}$$

without even knowing whether both are \mathcal{P} -positions or both are \mathcal{N} -positions, because the exact size of a heap cannot matter if it exceeds the maximum possible length of the game, which is at most the number of heaps present at the start of the game. As a second example, one can see that

$$\mathbf{1\ 1\ 2} \sim \mathbf{1\ 2} , \tag{2}$$

because only k heaps of size k can be relevant strategically — extra heaps of that size may be ignored, since they will vanish before they can be used.

Relations (1) and (2) are special cases of two general theorems which greatly reduce the set of positions requiring analysis.

DEPLETION THEOREM: Let G be the position $\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \cdots \mathbf{a}_n$. Any component of size $> n$ may be depleted down to size n without affecting the type of the position.

DELETION THEOREM: Let G be the position $\mathbf{a}_1 \mathbf{a}_2 \mathbf{a}_3 \cdots \mathbf{a}_n$. If $a_k < k$ for some k , then the component \mathbf{a}_1 may be deleted without affecting the type of the position. Indeed, if we define the *surplus* of G as the smallest non-negative integer s such that $k - a_k \leq s$ for all $1 \leq k \leq n$, then whenever $s > 0$, the components $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_s$ may all be deleted.

Proofs of the two theorems will appear in a forthcoming article by myself and Dan Ullman.

It's worth emphasizing that heaps may have size 0; in practice, one ignores 0-components, since they can never be selected, but the theoretical analysis of the game is simpler if one permits them. A position without 0-components will be called *proper*.

We call a proper position $\mathbf{a}_1 \cdots \mathbf{a}_n$ *reduced* if $1 \leq a_k \leq n$ for all k and if the surplus of the position is 0, that is, if $a_k \geq k$ for all k .

As an application of these theorems, consider the position

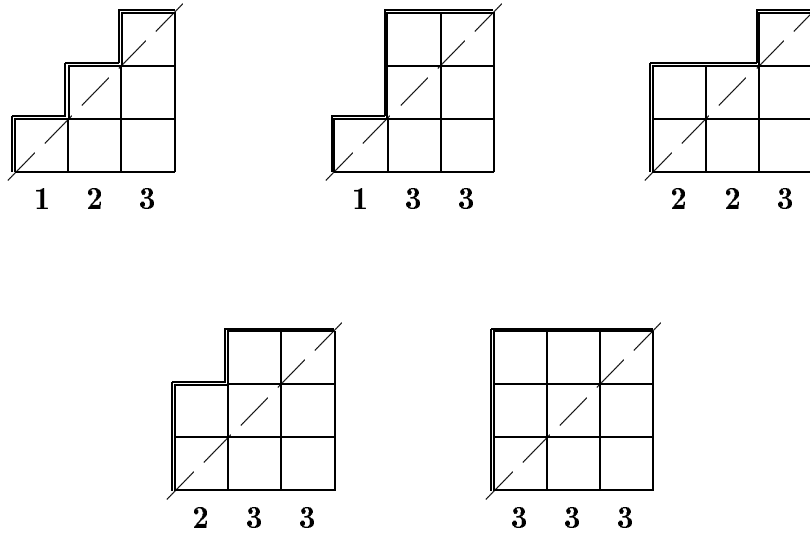
1 2 3 3 6.

By the depletion theorem, the position **1 2 3 3 5** has the same type. This position has surplus $4 - a_4 = 4 - 3 = 1$, and so is tantamount to **2 3 3 5**, by deletion. We may apply depletion once more to obtain the reduced position **2 3 3 4**.

The operations of deletion and depletion allow every position to be converted into a unique reduced form. To understand why, it is helpful to have an alternative way of conceiving of depletion: instead of replacing heaps of

size $> n$ by heaps of size n (where n is the number of heaps), replace all heaps of size $\geq n$ by heaps that are effectively infinite, to be denoted “*”; these correspond to cookies with an infinite shelf-life that remain edible indefinitely. This formal device makes it much easier to see what happens when one alternates the operations of deletion and depletion, because the process of “depletion” now leaves the surplus of a position alone. It becomes clear how one can determine the reduced form of a position: Perform deletion as many times as possible, and then depletion. For instance, given the position **1 2 3 3 6** \sim **1 2 3 3 ***, we should first delete the smallest component, yielding **2 3 3 6** \sim **2 3 3 ***; we should then deplete the largest component, yielding **2 3 3 4**. If the position we started with was proper, then “delete as many components as possible” means simply delete the s smallest components, where s is the surplus of the position.

The reduced positions with n heaps correspond to box-diagrams of width n and height n in which the bounding lattice-path that runs along the top of the diagram to connect the lower-left corner with the upper-right corner never goes below the diagonal line joining those two corners. Thus, the five reduced 3-heap positions may be represented as:



Such positions are in one-to-one correspondence with the lattice-paths that bound them, and these in turn are well known to be enumerated by the Catalan numbers (see for example [2], [3], and [4]). Thus the number of reduced positions with n components is $\binom{2n}{n} / (n + 1)$.

Table 1 shows all of the reduced positions involving 4 or fewer components; in the \mathcal{N} -positions, the winning components are circled. (We call a component *winning* if selecting that component leads to a \mathcal{P} -position.)

①	\mathcal{N}
1 ②	\mathcal{N}
2 2	\mathcal{P}
1 2 3	\mathcal{P}
① 3 3	\mathcal{N}
2 2 3	\mathcal{P}
② 3 3	\mathcal{N}
③ ③ ③	\mathcal{N}
① ② 3 4	\mathcal{N}
1 ② 4 4	\mathcal{N}
① ③ ③ ④	\mathcal{N}
1 ③ ④ ④	\mathcal{N}
1 ④ ④ ④	\mathcal{N}
② ② 3 4	\mathcal{N}
2 2 4 4	\mathcal{P}
② ③ ③ ④	\mathcal{N}
2 3 ④ ④	\mathcal{N}
2 4 4 4	\mathcal{P}
③ ③ ③ ④	\mathcal{N}
3 3 ④ ④	\mathcal{N}
3 4 4 4	\mathcal{P}
4 4 4 4	\mathcal{P}

Table 1

3 An Intriguing Pattern

If we tabulate positions of the form $\mathbf{1\ 2\ \cdots\ n}$, we notice a striking regularity:

①	\mathcal{N}
1 ②	\mathcal{N}
1 2 3	\mathcal{P}
① ② 3 4	\mathcal{N}
1 2 3 ④ ⑤	\mathcal{N}
1 2 3 4 5 6	\mathcal{P}
① ② ③ ④ 5 6 7	\mathcal{N}
1 2 3 4 5 ⑥ ⑦ ⑧	\mathcal{N}
1 2 3 4 5 6 7 8 9	\mathcal{P}
① ② ③ ④ ⑤ ⑥ 7 8 9 10	\mathcal{N}
1 2 3 4 5 6 7 ⑧ ⑨ ⑩ ⑪	\mathcal{N}
1 2 3 4 5 6 7 8 9 10 11 12	\mathcal{P}

Table 2

That is, with the exception of $n = 1$, we have:

For $n \equiv 0 \pmod{3}$, the position has type \mathcal{P} .

For $n \equiv 1 \pmod{3}$, the winning components are $\mathbf{1}, \dots, \frac{2}{3}(\mathbf{n} - \mathbf{1})$.

For $n \equiv 2 \pmod{3}$, the winning components are $\frac{2}{3}(\mathbf{n} + \mathbf{1}), \dots, \mathbf{n}$.

This pattern has been verified up to $n = 26$, but it has not been proved to hold for all n .

More generally, we can consider the two-dimensional family of positions of the form

$$\mathbf{a} \ \mathbf{a} + \mathbf{1} \ \mathbf{a} + \mathbf{2} \ \cdots \ \mathbf{a} + \mathbf{n} - \mathbf{1} .$$

Call them arithmetic progression positions, or AP-positions for short. Table 3 gives, for each value of a and n , the winning components of the corresponding AP-position. A blank entry means that the position has no winning components, i.e. no winning move is possible because the position has type \mathcal{P} .

		n									
		1	2	3	4	5	6	7	8	9	10
a	1	1	2		1-2	4-5		1-4	6-8		1-6
	2	2		2	4-5		2-4	6-8		2-6	8-11
	3	3		3-5		3-4	6-8		3-6	8-11	
	4	4		4-6		4-8		4-6	8-11		4-8
	5	5		5-7		5-9		5-11		5-8	10-14
	6	6		6-8		6-10		6-12		6-14	
	7	7		7-9		7-11		7-13		7-15	
	8	8		8-10		8-12		8-14		8-16	
	9	9		9-11		9-13		9-15		9-17	
	10	10		10-12		10-14		10-16		10-18	

Table 3

Lines are used to divide the behavior of the family of AP-positions into three distinct regimes. One regime (the “zeroeth”) consists of the single position $\mathbf{1}$ (with $a = n = 1$); the other two are infinite, and are determined by whether $n \leq 2a - 2$ or $n \geq 2a - 2$ (the boundary case $n = 2a - 2$ is properly considered as belonging to both).

In the regime $n \leq 2a - 2$, the type of the position is \mathcal{P} or \mathcal{N} according to whether n is even or odd. In fact, the same is true under the weaker condition $n \leq 2a - 1$. We prove this by considering the even case and odd case separately, and showing that one player can control the game by making sure that no heap ever goes down to zero. Suppose first that n is even, so that $n \leq 2a - 2$. At the start of the game there are n heaps, each of size at least a . If the second player always selects the smallest possible component, then after two turns (one turn by each player) there will be $n - 2$ heaps, each of size at least $a - 1$; after two more turns there will be $n - 4$ heaps, each of size at least $a - 2$; and so on. Finally, after $n - 2$ turns, there will be 2 heaps, each of size at least $a - (n - 2)/2 \geq 2$, which is a \mathcal{P} position. Hence the strategy of always removing the smallest heap guarantees a win for the second player. The argument for n odd is very similar, except that we conclude our analysis by looking at what happens after $n - 1$ turns: there will be 1 heap of size at least $a - (n - 1)/2 \geq 1$, which is an \mathcal{N} position.

The second infinite regime ($n \geq 2a - 2$) is more interesting: it contains among other things the special AP-positions $\mathbf{1 \ 2 \ \cdots \ n}$, and exhibits the same sort of period-3 behavior shown in Table 2. The patterns manifested in Table 3 have not been proven to hold in general, but they have been checked as far as $a = n = 20$. Also, the pattern can be shown to hold for the infinite diagonals $n = 2a - 1$ and $n = 2a$; the key idea is that games that begin in the second regime but very close to its boundary with the first regime can only get closer to the (already-understood) first regime, or move into it altogether, as the game progresses. The same method of analysis should in principle allow one to prove the pattern for any particular diagonal

$n = 2a + k$, but as k gets large and the starting position gets further from the boundary between the two regimes, the amount of detailed analysis required seems to grow without bound.

A simple rule for the opening move, apparently applicable to every AP-position (though for quite different reasons in the two regimes) is:

If the largest component is congruent to 0, 1, or 2 (mod 3), then select indifferently, select the smallest component, or select the largest component, respectively.

J.C. Kenyon has looked at the Grundy values of various positions in the cookie game. Following his lead, we can tabulate the Grundy values of the AP-positions. Once again, the regularity is striking:

		n									
		1	2	3	4	5	6	7	8	9	10
a	1	1	2	0	1	3	0	1	2	0	1
	2	1	0	1	3	0	1	2	0	1	3
	3	1	0	1	0	1	2	0	1	3	0
	4	1	0	1	0	1	0	1	3	0	1
	5	1	0	1	0	1	0	1	0	1	2
	6	1	0	1	0	1	0	1	0	1	0
	7	1	0	1	0	1	0	1	0	1	0
	8	1	0	1	0	1	0	1	0	1	0
	9	1	0	1	0	1	0	1	0	1	0
	10	1	0	1	0	1	0	1	0	1	0

Table 4

Notice that within the $n \geq 2a - 2$ regime, the Grundy value is constant on each northeast-to-southwest diagonal — that is, it depends only on $a + n$. Indeed, the Grundy value appears to be a periodic function of $a + n$ with period 6, given by the repeating sequence $0, 1, 2, 0, 1, 3, \dots$; this pattern has been observed as far out as $a = n = 20$. Can we somehow prove it in general by induction? If we try to devise such a proof, we almost immediately run afoul of the fact that the AP-positions do not form an isolated subset of “position-space”. Indeed, starting from an AP-position, either player can force his opponent into positions that look nothing at all like AP-positions. It follows that a winning strategy for a game whose first position is an AP-position must involve an understanding of many non-AP-positions, and it is correspondingly unlikely that one can prove facts about the class of AP-positions without having to consider the general behavior of the game. We can either find this fact of life vexing or encouraging: it is vexing because we are prevented from obtaining partial results about special classes of positions which by all evidence are orderly and understandable, yet it is encouraging because we are led to suspect that there is an over-arching pattern that governs not just the AP-positions but all positions, or at least all those in which the heap-sizes are distinct.

4 Strict Postions

Let us call a position $\mathbf{a}_1 \cdots \mathbf{a}_n$ *strict* if it is proper and if all its components are distinct, with the possible exception of components of size n , which may be duplicated. (The motivation for this exception comes from the Depletion Theorem: If the position **2 3 4** deserves to be called strict, then so does its reduced form **2 3 3**.) Clearly all AP-positions are strict, and it is left to the reader to check that the number of strict reduced positions with n heaps is 2^{n-1} .

The strict reduced positions seem to have some special characteristics.

GRUNDY VALUE CONJECTURE: Every strict position has Grundy value 0, 1, 2, or 3.

The evidence in favor of this conjecture is quite good. Strict positions with Grundy value 3 appear fairly early (starting with **2 3 4 4**), but exhaustive computer search reveals that there are no strict positions with Grundy value 4 having 20 or fewer components. In contrast, for non-strict positions, Grundy value 4 first appears for positions of size $n = 7$, and Grundy value 5 first appears for positions of size $n = 11$ (which is as far as the search went); so, for general positions, it would be rash indeed to conjecture that the Grundy values remain bounded.

WINNING INTERVAL CONJECTURE: If G is a strict position of type \mathcal{N} , then the winning components of G are exactly those whose sizes lie in a certain “winning interval” of natural numbers; that is, the winning components form a single uninterrupted subsequence $\mathbf{a}_i \mathbf{a}_{i+1} \cdots \mathbf{a}_j$.

This conjecture is *not* true for non-strict positions. For example, the position **1 2 2 3 6 6** does not have the winning interval property, since its winning components are the **2**'s and the **6**'s. On the other hand, the Winning Interval Conjecture has been checked for all strict reduced positions of length up to 20.

Shortly before this article went to press, Dan Ullman made the shrewd suggestion that every strict reduced position G has a certain innate duration, and that the way a player wins such a game is by controlling not merely the parity of the length of the game but the actual length of the game itself. This is indeed the case, and the observation turns out to be the key to the structure of the game's position-space. By the time this article appears in print, we hope to have a full theory for strict positions, and to have proofs of all the conjectures in this article.

REFERENCES

- [1] Elwyn R. Berlekamp, John H. Conway, and Richard K. Guy, *Winning Ways for Your Mathematical Plays*, Academic Press, 1982.
- [2] Martin Gardner, Catalan numbers: an integer sequence that materializes in unexpected places, *Scientific American* 234, No. 6 (June, 1976), Mathematical Games, pp. 120-125 and 132.
- [3] Fred S. Roberts, *Applied Combinatorics*, Prentice-Hall, 1984, pp. 234-243.
- [4] I. M. Yaglom and A. M. Yaglom, *Challenging Mathematical Problems with Elementary Solutions*, Holden-Day, 1964, volume 1, problems 53, 83.