Nim Games

1 One Pile Nim Games

Consider the following two-person game.

- There are initially *n* stones on the table.
- During a move a player can remove either one, two, or three stones.
- If a player cannot move then he loses (this only happens when there are 0 stones on the table).

Notation 1.1 We denote this game (1,2,3)-NIM.

Before reading on, think about how you should play this game to win starting with a pile of, say, 21 stones.

Strategy: It is clear that if there are only one, two, or three stones left (on your turn), you can win the game by taking all of them. If, however, there are exactly four stones you will lose, because no matter how many you take, you will leave one, two, or three and your opponent will win by taking the remainder. If there are five, six, or seven stones you can win by taking just enough to leave four stones. If there are eight stones you will again lose, because you must leave five, six, or seven. Etc.

Here is the win/loss information for take one-two-or-three for up to 21 stones.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
L W W W L W W W L W W W L W W W L W W W L W

In general, if there are a multiple of four stones you lose. Otherwise you win (by taking enough stones to leave a multiple of four for your opponent). So we see the pattern LWWW, which is repeated. Thus, 21 stones is a win: take one stone.

Def 1.2 Let $a_1, a_2, \ldots, a_k$ be k distinct natural numbers (nonzero). Then $(a_1, \ldots, a_k)$-NIM is the following game.

- There are initially n stones on the table.
- During a move a player can remove either a_1, a_2, a_3 or ... a_k stones.
- If a player cannot move then he loses. (This may happen even if there are a non-zero number of stones on the table. For example, if the game is (2,3)-NIM and there is 1 stone on the table, then the player cannot move.)

1.1 Modular Arithmetic

In this section we define terminology that will be useful in studying NIM games. This section will use modular arithmetic.

Def 1.3 a mod m is the remainder obtained when a is divided by m.

For example, 8 mod 3 = 2.

We are used to working with modular arithmetic in everyday life. For example, starting from the beginning of the year we could count the seconds, minutes, and hours exactly. In practice, this is too complicated so we take the seconds mod 60, the minutes mod 60, and the hours mod 12. (Sometimes we take hours mod 24 to distinguish a.m. and p.m.)

Mathematicians typically work with a slightly different definition of modular arithmetic. They say that that two numbers a and b are *equivalent* mod m if a and b differ by a multiple of m. This is written $a \equiv b \pmod{m}$.

Consider the first game from the last section. Using the modular arithmetic notation, we say that a pile of n stones is a win for the first Player if $n \equiv 1,2,3 \pmod{4}$ and a loss if $n \equiv 0 \pmod{4}$.

2 Generalized Take-One-Two-or-Three

Consider the following game:

- Initially the game begins with a pile of n stones.

- During a turn a player may either remove $a_1, a_2, \ldots, a_k$ stones. We call this $(a_1, \ldots, a_k)$-nim.
- The first player who cannot make a move loses.

Lets look at the game where we only allow one or four stones to be taken. In our notation this is the game (1,4)-NIM. Now it helps to think about the game more generally. Assume there are s stones left. If $s < 4$, you can take only one stone leaving $s - 1$ stones. If $s - 1$ is a loss for your opponent then s is a win for you, and if $s-1$ is a win for your opponent then s is a loss for you. If $s \geq 4$, you can remove one or four leaving either $s-4$ or $s-1$ stones. If either $s - 4$ or $s - 1$ is a loss for your opponent, then s is a win for you. Otherwise $s - 4$ and $s - 1$ are both wins for your opponent, so s must be a loss for you.

We can build a win/loss table by looking for each s at the information for $s-1$ and $s-4$ stones. Here it is for take one-or-four for up to 21 stones.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
L	W	L	W	W	L	W	L	W	W	L	W	L	W	W	L	W	L	W	W	L	W

We see that the pattern is LWLWWL. How do we confirm that this pattern repeats forever? We can do this by showing that once the pattern WLWWL exists it will keep repeating. The table clearly starts WLWWL (for 1 to 5 stones). After that the win/loss situation for any group of five stones depends only on the previous group. So once the pattern WLWWL continues into the next group of five stones (6 to 10) it has to repeat forever.

Theorem 2.1 *In the game* (1,4)-*NIM*

1. *If* $n \equiv 1,3,4 \pmod 5$ *then player I wins.*

2. *If* $n \equiv 0,2 \pmod 5$ *then player II wins.*

Proof: We are proving that the pattern WLWWL always repeats. Assume that the pattern holds for up to $5r$ stones for some natural number $r \geq 1$. Imagine that you are player I. This means that $5r - 4$ is a win, $5r - 3$ is a loss, $5r - 2$ and $5r - 1$ are wins, and $5r$ is a loss. Then for $5r + 1$ stones, you either move, to $5r-3$ or to $5r$, leaving your opponent in a losing position, so $5r+1$ is a win for you (player I); for $5r+2$ stones both moves, to $5r-2$ and to the newly discovered $5r + 1$, leaves your opponent in a winning position, so it is a loss for you (hence a win for player II); for $5r + 3$ stones taking one stone leaves the newly discovered $5r + 2$, which is a loss for your opponent, and thus a win for you (player I); for $5r + 4$ stones taking four stones leaves $5r$, which is a loss for your opponent, and thus a win for you (player I); and, for $5r + 5$ stones both moves to $5r + 1$ and $5r + 4$ leave our opponent in a winning positions, so it is a loss for you (hence a win for player I); We thereby reproduce the pattern WLWWL.∎

Note 2.2 The two sets {0,2} and {1,3,4} have the following properties:

(1) if $n \in \{1,3,4\} \pmod 5$ then some move will create $\{0,2\} \pmod 5$,

(2) if $n \in \{0,2\}$ then all moves will create $\{1,3,4\} \pmod 5$.

This kind of structure is common in proofs that certain values lead to player I/player II wins in NIM.

Def 2.3 If there is a pattern that repeats after some initial segment, the game is *periodic*. The length of a minimum repeating pattern is the *period*.

Using this notation (1,2,3)-NIM has period 4, and (1,4)-NIM has period 5.

Consider the game (2,4,7)-NIM (where you can remove 2, 4, or 7 stones). It has the following win/loss table.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
L	L	W	W	W	W	L	W	W	L	W	W	L	W	W	L	W	W	L	W	W	L

It has an initial segment LLWW. Then the pattern WWL repeats forever, so the game is periodic with period 3.

Theorem 2.4 *For any $h_1,\ldots,h_n$ the games $(h_1,\ldots,h_n)$-NIM is periodic.*

Aside Let us consider how many ways we can fill n positions with W and L's. If $n = 1$ there are only 2 ways to do this (either its a W or an L). For $n = 2$ there are 4 ways to arrange them: WW WL LW and LL. If $n = 3$, notice that we can take each unique pattern from the $n = 2$ case, and either put a W or and L at the end. So if there are 4 ways to arrange them when $n = 2$, there are $2 \cdot 4 = 8$ ways to arrange them for $n = 3$. Similarly there are $2 \cdot 8 = 16$ ways to arrange them for $n = 4$. In general then, we see there are 2^n ways to arrange them when there are n positions.

Proof: Let m be the maximum of the h_k. Consider the first $m(2^m + 1)$ entries in the win/loss table. Group them into 2^m+1 groups of m contiguous entries. There are only 2^m possible patterns for each group. Why? Since there are $2^m + 1$ groups, by the pigeon hole principle, two groups must be the same. Whatever pattern occurs between those two groups must repeat after that forever. So the period must be at most $m2^m$.▮

3 Easy Two-Pile Nim

Consider the following NIM-type game:

- Initially there are two piles of stones. We denote a position (a,b).
- During a move a player removes as many stones as he wants from one of the piles.
- If a player cannot move then he loses. (This only happens when the position is (0,0).)

Here is an example of a play of the game.

1. Starting position is (20,14).
2. Player I removes 6 from pile 1. Position is now (14,14).
3. Player II removes 4 from pile 1. Position is now (10,14).
4. Player I removes 4 from pile 2. Position is now (10,10).

5. Player II removes 8 from pile 2. Position is now (10,2).

6. Player I removes 8 from pile 1. Position is now (2,2).

7. Player II removes 2 from pile 1. Position is now (0,2).

8. Player I removes 2 from pile 2. Position is now (0,0).

9. Player II cannot move, so he loses.

Note that Player I's strategy was to always even out the piles. We prove that this works. Let *ONE* be the set of ordered pairs where I wins, and let *TWO* be the set of ordered pairs where II wins.

Theorem 3.1 *$(a,b) \in ONE$ iff $a \neq b$.*

Proof: We prove this informally. Notice that if $a \neq b$, player I can always make them equal. Since player II must remove some stones, he must then make it such that $a \neq b$ again. So when the game begins with $a \neq b$ on player I's first turn, player I can assure that this remains the case for each successive turn. Eventually player II will be forced to remove the last stone from one of the piles (because he is required to take some stone), and player I will win. On the other hand, if $a = b$ to begin the game, the table is exactly reversed. Whatever play I makes, he will have to leave the piles uneven. Suppose after player I's first move it is now $a \neq b'$. Now we can view player II as player I on a new game where $a \neq b'$, and the proof that he wins is the same as above. ▮

4 General Two-pile Nim

Consider the following game.

1. Initially there are two piles of stones. We denote a position by (a,b).

2. During a turn a player can do one of the following.

    - Remove 1,2, or 3 stones from pile 1.

- Remove 1,3, or 4 stones from pile 2.

3. If you cannot move, you lose. (This will only happen if the position is (0,0).

More generally, consider the following game.

Def 4.1 Let G_1 and G_2 be 1-pile NIM games. $G = G_1 \oplus G_2$ is defined as follows.

1. Initially there are two piles of stones. We denote a position by (a,b).
2. During a turn a player can do one of the following.
   - Remove what is allowed on pile 1 by G_1.
   - Remove what is allowed on pile 2 by G_2.
3. If you cannot move, you lose. (This may happen even if the position isnot (0,0). For example, if G_1 is (2,3)-NIM and G_2 is (1,2)-NIM then there is no move from position (1,0).)

Note the following philosophies from the previous games studied.

In (1,2,3)-NIM you try to make your opponent look at position a where $a \equiv 0 \pmod 4$, while never looking at such a position yourself. Note that the losing position 0 has the property that $0 \equiv 0 \pmod 4$. Hence you are trying to make your opponent look at a position that has a property also shared by the losing position, while never looking at a position with that property yourself. Since the positions numeric value keeps decreasing, and you never look at a position with that property, you must win.

In two pile-NIM you try to make your opponent look at position (a,b) where $a = b$, while never looking at such a position yourself. Note that the losing position (0,0) has the property that $a = b$. Hence you are trying to make your opponent look at a position that has a property also shared by the losing position, while never looking at a position with that property yourself. Since the positions numeric value keeps decreasing, and you never look at a position with that property, you must win.

So, we need to find some property of the position (0,0) and make sure our opponent always sees a position with that property.

4.1 Grundy numbers

Before analyzing two-pile Nim games, we need to analyze 1-pile Nim games in more depth. We will assign numbers to all the positions in a one-pile nim game. These numbers would not be needed if all we wanted to do was win the 1-pile game; however, they are needed if we want to study many-pile games.

Def 4.2 A game is *Impartial (or nonpartisan)* if from any position exactly the same moves are available to both Players.

All the games of the form $(a_1,...,a_k)$-NIM are impartial. Chess is not impartial because one Player can only move the white pieces and the other Player the black pieces. Similarly, checkers and go are not impartial.

We will see that we can "solve" all impartial games where the last Player to move wins. We give a recursive definition for the Grundy number of a position in an impartial game.

Def 4.3 Let G be a game. The *Grundy number* of a position P in G is

- If P is a final position, it has Grundy number 0. (It is a loss for the Player whose move it is.)

- Otherwise the Grundy number is the minimum natural number that is *not* a Grundy number of any position that is attained from making one move on P.

Lemma 4.4 *Let G be a 1-pile NIM game and g be the Grundy function of G. Let* a,a^0 *be positions in G such that you can get from a to* a^0 *in one move. Then* $g(a)$ 6= $g(a^0)$.

Proof: Assume that from position a you can get to, in one move, the positions $c_1,c_2,...,c_k$. Then $g(a)$ is the least number that is not in the set $\{g(c_1),...,g(c_k)\}$. In particular $g(a)$ cannot equal any number in $\{g(c_1),...,g(c_k)\}$. Since a^0 is one of the c_i, $g(a)$ 6= $g(a^0)$.▮

Example 1: Here are Grundy numbers for (1,2,3)-NIM for up to 21 stones.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

0 1 2 3 0 1 2 3 0 1 2 3 0 1 2 3 0 1 2 3 0 1

It is periodic with sequence 0,1,2,3.

We write the Grundy function as

$$g(n) = \begin{cases} 0 & \text{if } n \equiv 0 \pmod 4; \\ 1 & \text{if } n \equiv 1 \pmod 4; \\ 2 & \text{if } n \equiv 2 \pmod 4; \\ 3 & \text{if } n \equiv 3 \pmod 4. \end{cases}$$

Example 2: Here are Grundy numbers for (1,3,4)-NIM for up to 21 stones.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

0 1 0 1 2 3 2 0 1 0 1 2 3 2 0 1 0 1 2 3 2 0

It is periodic with sequence 0,1,0,1,2,3,2.

We write the Grundy function as

$$g(n) = \begin{cases} 0 & \text{if } n \equiv 0,2 \pmod 7; \\ 1 & \text{if } n \equiv 1,3 \pmod 7; \\ 2 & \text{if } n \equiv 4,6 \pmod 7; \\ 3 & \text{if } n \equiv 5 \pmod 7. \end{cases}$$

Example 3: For example, here are the Grundy numbers for (2,4,7)-NIM up to 21 stones.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

0 0 1 1 2 2 0 3 1 0 2 1 0 2 1 0 2 1 0 2 1 0

It is eventually periodic with sequence 1,0,2. Notice that it has an initial sequence of length 8, as compared to length 4 when we looked only at win/loss information.

We write this function as

$$g(n) = \begin{cases} 0 & \text{if } n = 0,1,6 \text{ or } n \geq 8 \wedge n \equiv 0 \pmod 3; \\ 1 & \text{if } n = 2,3 \text{ or } n \geq 8 \wedge n \equiv 1 \pmod 3; \end{cases}$$

2 if $n = 4,5$ or $n \geq 8 \wedge n \equiv 2 \pmod 3$; 3 if $n = 7$.

4.2 Using Grundy Numbers for Two-pile NIM Games

We can now state the philosophy of two-pile NIM games. Let G_1 be a 1-pile NIM game with Grundy function g_1. Let G_2 be a 1-pile NIM game with Grundy function g_2. Let $G = G_1 \oplus G_2$. The position (0,0) has the property that $g_1(0) = g_2(0)$. To win we make our opponent always look at (a,b) with $g_1(a) = g_2(b)$.

Theorem 4.5 *Let G_1 be a 1-pile NIM game with Grundy function g_1. Let G_2 be a 1-pile NIM game with Grundy function g_2. Let $G = G_1 \oplus G_2$. Let ONE be the initial positions (a,b) from which I wins. Let TWO be the initial positions (a,b) from which II wins. Then $(a,b) \in ONE$ iff $(g_1(a)$ 6= $g_2(b))$.*

Proof: We prove this in a way similar to the way we proved it for simple 2pile games. Suppose that to begin the game, $g(a)$ 6= $g(b)$. Player I can move to make it so that $g(a) = g(b^0)$. This follows trivially from the definition of grundy numbers: without loss of generality, suppose $g(a) < g(b)$. By the definition of grundy numbers, we know that there is some number of stones b^0 reachable from position b on pile 2 such that $g(b) = g(a)$. This is because for every natural number less than $g(b)$ there is some reachable position from b with that grundy number. On the other hand, we notice that if $g(a) = g(b)$, there is nothing II can do to keep them equal after his turn. This is because all reachable positions from either pile must result in a lower grundy number. Eventually II will have to move such that $g(b) = 0$ and there are no more legal moves on that pile. Player I can then assure that $g(a) = 0$ after each of his moves, until there is no legal move on that pile either. Player I then wins. To see that player I loses when the game begins with $g(a) = g(b)$ we simply reverse the table.▮

We end this section where we began- with a particular 2-pile NIM game.

Example 4.6 Let G_1 be (1,2,3)-NIM game with Grundy function g_1. Let G_2 be (1,3,4)-NIM game with Grundy function g_2. Let $G = G_1 \oplus G_2$. We know that

$$g_1(n) = \begin{cases} 0 & \text{if } n \equiv 0 \pmod 4; \\ 1 & \text{if } n \equiv 1 \pmod 4; \\ 2 & \text{if } n \equiv 2 \pmod 4; \\ 3 & \text{if } n \equiv 3 \pmod 4. \end{cases}$$

We know that

$$g_2(n) = \begin{cases} 0 & \text{if } n \equiv 0,2 \pmod 7; \\ 1 & \text{if } n \equiv 1,3 \pmod 7; \\ 2 & \text{if } n \equiv 4,6 \pmod 7; \\ 3 & \text{if } n \equiv 5 \pmod 7. \end{cases}$$

Hence we have that $(a,b) \in TWO$ iff

- $a \equiv 0 \pmod 4$ and $b \equiv 0,2 \pmod 7$.
- $a \equiv 1 \pmod 4$ and $b \equiv 1,3 \pmod 7$.
- $a \equiv 2 \pmod 4$ and $b \equiv 4,6 \pmod 7$.
- $a \equiv 3 \pmod 4$ and $b \equiv 5 \pmod 7$.

5 Many Pile Nim Games

We define many-pile NIM games.

Def 5.1 Let $G_1,G_2,\ldots,G_k$ be 1-pile NIM games. $G = \oplus_{i=1}^{k} G_i$ is defined as follows.

1. Initially there are k piles of stones. We denote a position by $(a_1,a_2,\ldots,a_k)$.
2. During a turn a player can do one of the following.
   - Remove what is allowed on pile 1 by G_1.
   - Remove what is allowed on pile 2 by G_2.
   - $\vdots$
   - Remove what is allowed on pile k by G_k.
3. If you cannot move, you lose. (This may happen even if the positionis not (0,0,0,...,0). For example, if G_1 is (2,3)-NIM and all the rest are (1,2)-NIM then from position (1,0,0,...,0) there is no move.)

Aside The numbers we use every day are in base 10 (aka decimal). Sometimes its useful to use other bases: we will quickly learn about base 2 (also called binary). In base 10, the number 357 is actually shorthand for $3 \cdot 10^2 + 5 \cdot 10^1 + 7 \cdot 10^0$. Another way of saying this is that there is a ”hundreds” place, a ”tens” place and a ”ones” place. Also notice that we use 10 digits (0-9). In binary, we use powers of 2 rather than powers of 10, and there are only 2 digits (0 and 1). For example, the number 101 in binary is the same as the number 5 in decimal: $1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 5$.

We need the following definition to just state our theorem.

Def 5.2 Let $G_1, G_2, \ldots, G_k$ be 1-pile NIM games. $G = \oplus_{i=1}^{k} G_i$. Let $(a_1, \ldots, a_k)$ be a position in G. Write $g_1(a_1)$, ..., $g_k(a_k)$ in base 2. Add zeros to the left of the numbers so that all the numbers have the same length. Write the numbers down in a table. This table is called $T(G, a_1, \ldots, a_k)$.

Example 5.3 Let G_1 be (1,2,3,4,5)-NIM. Let G_2 be (1,3,4)-NIM. Let G_3 be (2,4,7)-NIM. Let $G = G_1 \oplus G_2 \oplus G_3$. Let g_1, g_2, g_3 be the Grundy functions of G_1, G_2, G_3. We write the Grundy Functions of G_1, G_2, and G_3 in both base 10 and base 2 of length 5, since that is the largest length. The Grundy function of G_1 is

$$g_1(n) = \begin{cases} 0 = (000)_2 & \text{if } n \equiv 0 \pmod 5; \\ 1 = (001)_2 & \text{if } n \equiv 1 \pmod 5; \\ 2 = (010)_2 & \text{if } n \equiv 2 \pmod 5; \\ 3 = (011)_2 & \text{if } n \equiv 3 \pmod 5; \\ 4 = (100)_2 & \text{if } n \equiv 4 \pmod 5. \end{cases}$$

The Grundy function of G_2 is

$$g_2(n) = \begin{cases} 0 = (000)_2 & \text{if } n \equiv 0,2 \pmod 7; \\ 1 = (001)_2 & \text{if } n \equiv 1,3 \pmod 7; \\ 2 = (010)_2 & \text{if } n \equiv 4,6 \pmod 7; \\ 3 = (011)_2 & \text{if } n \equiv 5 \pmod 7. \end{cases}$$

The Grundy function of G_3 is

$$g_3(n) = \begin{cases} 0 = (000)_2 & \text{if } n = 0,1,6 \text{ or } n \geq 8 \wedge n \equiv 0 \pmod 3; \\ 1 = (001)_2 & \text{if } n = 2,3 \text{ or } n \geq 8 \wedge n \equiv 1 \pmod 3; \\ 2 = (010)_2 & \text{if } n = 4,5 \text{ or } n \geq 8 \wedge n \equiv 2 \pmod 3; \\ 3 = (011)_2 & \text{if } n = 7. \end{cases}$$

Consider the position (24,2,0). The Grundy numbers are (100,010,000). We pad out 0's on the left to make the numbers the same length to obtain (00100,00010,00000). Hence $T(G,24,2,0)$ is

100
010
000

Theorem 5.4 *Let $G_1,...,G_k$ be 1-pile NIM games. Let $g_1,...,g_k$ be the associated Grundy functions. Let $G = \oplus_{i=1}^{k} G_i$, Let $(a_1,...,a_k)$ be a position in G. $(a_1,...,a_k) \in$ ONE iff some column of $T(G,a_1,...,a_k)$ has an odd number of 1's.*

Proof: Again similar to before. Notice that if some column has an odd number of 1's in it, then I can make sure that all columns have an even number of 1's after his move. He simply chooses the pile with the greatest grundy number. From that pile, all other smaller grundy numbers are available. He chooses the one that evens out the 1's in each column. Of course player II is now forced to make (at least) one column have an odd number: otherwise he wouldn't have taken any sticks (do you see why this is true?). This logic again holds until the game ends.▮

Example 5.5 We revisit this example from before. Let G_1 be (1,2,3,4,5)NIM. Let G_2 be (1,3,4)-NIM. Let G_3 be (2,4,7)-NIM. Let $G = G_1 \oplus G_2 \oplus G_3$. Let g_1,g_2,g_3 be the Grundy functions of G_1,G_2,G_3. We write the Grundy Functions of G_1,G_2, and G_3 in both base 10 and base 2 of length 5, since that is the largest length. The Grundy function of G_1 is

$$0 = (000)_2 \quad \text{if } n \equiv 0 \pmod 5;$$
$$\text{if } n \equiv 1 \pmod 5;$$
$$\text{if } n \equiv 2 \pmod 5;$$

$$g_1(n) = \begin{cases} 1 = (001)_2 \\ 2 = (010)_2 \\ 3 = (011)_2 & \text{if } n \equiv 3 \pmod 5; \\ 4 = (100)_2 & \text{if } n \equiv 4 \pmod 5. \end{cases}$$

The Grundy function of G_2 is

$$g_2(n) = \begin{cases} 0 = (000)_2 & \text{if } n \equiv 0,2 \pmod 7; \\ 1 = (001)_2 & \text{if } n \equiv 1,3 \pmod 7; \\ 2 = (010)_2 & \text{if } n \equiv 4,6 \pmod 7; \\ 3 = (011)_2 & \text{if } n \equiv 5 \pmod 7. \end{cases}$$

The Grundy function of G_3 is

$$g_3(n) = \begin{cases} 0 = (000)_2 & \text{if } n = 0,1,6 \text{ or } n \geq 8 \land n \equiv 0 \pmod 3; \\ 1 = (001)_2 & \text{if } n = 2,3 \text{ or } n \geq 8 \land n \equiv 1 \pmod 3; \\ 2 = (010)_2 & \text{if } n = 4,5 \text{ or } n \geq 8 \land n \equiv 2 \pmod 3; \\ 3 = (011)_2 & \text{if } n = 7. \end{cases}$$

It is possible to write down a statement like

$(a_1,a_2,a_3) \in TWO$ iff XXX

however it would be quite complicated involving many cases. We give a few examples, both for ONE and TWO.

1. If $a_1 \equiv 0 \pmod 5$, $a_2 \equiv 4,6 \pmod 7$ and $a_3 \geq 8$ and $a_3 \equiv 2 \pmod 3$ then $g_1(a_1) = 0 = (00)_2$, $g_2(a_2) = 2 = (10)_2$, and $g_3(a_3) = 2 = (10)_2$. $T(G,a_1,a_2,a_3)$ is

   00
   10
   10

   Notice that every column has an even number of 1's. Hence $(a_1,a_2,a_3) \in TWO$.

2. If $a_1 \equiv 3 \pmod 5$, $a_2 \equiv 1,3 \pmod 7$ and $a_3 \geq 8$ and $a_3 \equiv 1 \pmod 3$ then $g_1(a_1) = 3 = (11)_2$, $g_2(a_2) = 1 = (01)_2$, and $g_3(a_3) = 2 = (01)_2$. $T(G,a_1,a_2,a_3)$ is

11
01
01

Notice that the last column has an odd number of 1's. Hence $(a_1,a_2,a_3) \in ONE$.

6 Misere Version

What if the goal to **not** take the last stone. This is the misere version.

- There is initially a pile of n stones.
- Players alternate turns removing 1,2, or 3 stones.
- The player who makes the last move loses.

Think about the strategy for misere take one-two-or-three before continuing.

Strategy: It is clear that if there are only one stone left you lose,

Strategy: It is clear that if there are only one stone left you lose, because you must take it. If there are two, three, or four stones left, you will win by taking enough stones to leave exactly one. If there are exactly five stones you will again lose, because you will have to leave two, three, or four and your opponent will take enough to leave one stone. If there are six, seven, or eight stones you will win by taking enough to leave five stones. Etc.

Here is the win/loss information for misere take one-two-or-three for up to 21 stones.

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
W L W W W L W W W L W W W L W W W L W W W L

In general, if there are a multiple of four stones plus one you lose. Otherwise you win by taking enough stones to leave a multiple of four plus one. The pattern is WLWW, which is repeated. Using the modular arithmetic notation, we say that a pile of n stones is a win for the first Player if $n \equiv 0,2,3 \pmod 4$ and a loss if $n \equiv 1 \pmod 4$. In this version, 21 stones is a loss for the first Player.

7 Dynamic Programming

Consider the following game

- Initially there are 3 piles of stones. We denote a position by (a,b,c).
- A player may remove a prime from the first pile, or a square from the second pile, or a Fibonacci number from the third pile.
- If a player cannot move then they lose.

This is a rather complicated game it is doubtful that it has a succinct statement about when player I wins. However, we can still (with a computer program) calculate who wins which positions rather easily. The key is that we never try to calculate who wins (a_1,a_2,a_3) until we know ALL of the lower positions.

$$W(a,b,c) = \begin{cases} I & \text{if player I wins when the game starts with } (a,b,c); \\ II & \text{if player II wins when the game starts with } (a,b,c). \end{cases}$$

Let PR be the set of primes, SQ be the set of squares and FIB be the set of Fibonacci numbers. Then the following holds:

$W(0,0,0) = II$

$$W(a,b,c) = \begin{cases} I & (\exists p \in PR)[a \geq p \wedge W(a-p,b,c) = II] \text{ OR} \\ & (\exists s \in SQ)[b \geq s \wedge W(a,b-s,c) = II] \text{ OR} \\ & (\exists f \in FIB)[c \geq f \wedge W(a,b,c-f) = II]; \end{cases}$$

II otherwise.

If by the time we are looking at (a,b,c) we have already computed W of all (a^0,b^0,c^0) with $a^0+b^0+c^0 < a+b+c$ then we can carry out this calculation easily. In fact, the problem we are really solve here is not "What is $W(a,b,c)$?' but instead "What is $W(a,b,c)$ for all (a,b,c) with $a + b + c \leq n$?"

Here is psuedocode for the problem. It is not very efficient; however, it can be made alot more efficient.

```
Input(n).
W(0,0,0)=II. (This sets W(0,0,O) to the value II.) for i=1 to n for
a=0 to i for b=0 to i-a c=i-a-b (So now (a+b+c=i)
W(a,b,c)=II (We will set this to I if we find a good move.) LOOKING = YES
(We are looking for the good move.) for p=1 to a
if (p is prime) and (W(a-p,b,c)=II) and (LOOKING=YES) then
W(a,b,c)=I
LOOKING = NO If
LOOKING = YES then for
s=1 to b
if (s is square) and (W(a,b-s,c)=II) and (LOOKING=YES) then
W(a,b,c)=I
LOOKING = NO If
LOOKING = YES then for
f=1 to c
if (f is a Fibonacci) and (W(a,b,c-f)=II) and (LOOKING=YES) then
W(a,b,c)=I
LOOKING = NO
END
```

Even though this is not a nice formula, it is an efficient calculation (or can be made into one.)