Theory of Impartial Games

February 3, 2009

Introduction

Kinds of Games We'll Discuss

Much of the game theory we will talk about will be on ***combinatorial games*** which have the following properties:

- There are two players.
- There is a finite set of positions available in the game (only on rare occasions will we mention games with infinite sets of positions).
- Rules specify which game positions each player can move to.
- Players alternate moving.
- The game ends when a player can't make a move.
- The game eventually ends (it's not infinite).

Today we'll mostly talk about ***impartial games***. In this type of game, the set of allowable moves depends only on the position of the game and not on which of the two players is moving. For example, Nim, sprouts, and green hackenbush are impartial, while games like GO and chess are not (they are called ***partisan***).

The Game of Nim

We first look at the simple game of Nim, which led to some of the biggest advances in the field of combinatorial game theory. There are many versions of this game, but we will look at one of the most common.

How To Play

There are three piles, or *nim-heaps*, of stones. Players 1 and 2 alternate taking off any number of stones from a pile until there are no stones left. There are two possible versions of this game and two corresponding winning strategies that we will see. Note that these definitions extend beyond the game of Nim and can be used to talk about impartial games in general.

- ***Normal Play*** The player to take the last stone (or in general to make the last move in a game) wins. This is called normal play since most impartial games are played this way, although Nim usually is not.
- ***Misere Play*** The player that is forced to take the last stone loses.

An example normal play game is shown below:

```
Sizes of heaps  Moves
A B C

3 4 5           I take 2 from A
1 4 5           You take 3 from C
1 4 2           I take 1 from B
1 3 2           You take 1 from B
1 2 2           I take entire A heap leaving two 2's.
0 2 2           You take 1 from B
0 1 2           I take 1 from C leaving two 1's.
0 1 1           You take 1 from B
0 0 1           I take entire C heap and win.
```

What is your winning strategy? Luckily, we can find one. Nim has been solved (we use the term solved loosely here, but there are several categories of "solutions" to games) for all starting positions and for any number of heaps.

First we'll look at different types of game positions, then we'll do some work with "nimbers" (yes, that really is a word) and then apply them to finding a solution to Nim.

Types of impartial game positions

- A game is in a **P-position** if it secures a win for the Previous player (the one who just moved).
- A game is in a **N-position** if it secures a win for the Next player.

So in normal play Nim with three heaps, (0,0,1) is an N-position and (1,1,0) is a P-position. We call the position from which no possible moves are left a **terminal position**.

To find whether a Nim position is N or P, we work backwards from the end of the game to the beginning in a process called **backwards induction** :

1. Label every terminal position as P.
2. Label every position that can reach a P position as N.
3. For positions that only move to N positions, label P.
4. At this point either all positions are labeled or return to step 2 and repeat the process until all positions are labeled.

For misere play, just invert step 1: every terminal position is N.

Applying these rules to Nim, we first set the only terminal position (in other games there could be many) 0,0,0, to P. It is obvious that any position (0,0,n) is an N position, since the next player can just take the last heap in one turn.

Practice With N and P positions

Consider the subtraction game in which you start with a pile of chips and players alternate taking away any number s_i from the set $S = \{1, 3, 4\}$ of chips from the heap. The player to take the last chip loses.

We can see that 1, 3 and 4 must be N-positions, since the next player can just take all of the chips. 0 must be a P-position of course, since the player that moved to 0 wins. 2 must be a P position since the only legal move is to an N-position. Then 5 and 6 must be N since they can be moved to 2. If we continue analyzing the game in this manner we get the following sequence of N and P positions:

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
pos	P	N	P	N	N	N	N	P	N	P	N	N	N	N	P

This period sequence of Ns and Ps (PNPNNNN) continues forever. In fact, almost all subtraction games have such periodic sequences of N and P values.

Nimber Arithmetic

The key operation in the solution to Nim is binary addition without carrying. To add two numbers in this manner, first write out their binary expansions, and then take the exclusive or (XOR) of the two numbers bit by bit. The following is an example:

```
 3     011
+5     100
---------
 7     111
```

In the XOR operation, 1+1= 0 = 0+0, 1+0=1=1+0. Another way to look at it is that if you are adding an odd number of ones the answer is 1, an even number of ones gives 0. We will write this kind of addition of two numbers x and y as $x \oplus y$.

Below is an addition table for nimbers:

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

The Solution

Now comes the big moment... the solution! Notice that if we take the sum of all the nim-heaps, at the end the nimsum of all the heaps is equal to 0 (since adding 0 together any number of times gives 0). But there are other times that the nim-sum can be 0. Note that any $x \oplus x = 0$, since any number XORed with itself is 0. We will now prove the following theorem about Nim:

Theorem The winning strategy in normal play Nim is to finish every move with a Nim-sum of 0.

To prove this we will use the following two lemmas:

Lemma 1 If the Nim-sum is 0 after a player's turn, then the next move must change it.

To prove this, let the number of stones in the heaps be $x_1, x_2, \ldots x_n$, and s be the nim-sum, $s = x_1 \oplus x_2 \oplus x_3 \oplus \ldots \oplus x_n$. Let t be the sum of the heaps

y_i after the move, $t = y_1 \oplus y_2 \oplus y_3 \oplus \ldots \oplus y_n$. Then if $s = 0$, the next move causes some $x_k \neq y_k$ and the rest of the $x_i = y_i$ for $i \neq k$, since only one pile of stones is changed. Then:

$t = 0 \oplus t$
$= s \oplus s \oplus t$
$= s \oplus (x_1 \oplus x_2 \oplus \ldots \oplus x_n) \oplus (y_1 \oplus y_2 \oplus \ldots \oplus y_n)$
$= s \oplus (x_1 \oplus y_1) \oplus (x_2 \oplus y_2) \oplus \ldots \oplus (x_k \oplus y_k)$
$= s \oplus x_k \oplus y_k$
If s is 0, then t must be nonzero, since $x_k \oplus y_k$ will never be 0. Therefore, if you make the nim-sum 0 on your turn, your opponent must make it nonzero. Think of the games as being *balanced* when the nim-sum is 0. From a balanced position it is only possible to unbalance it by moving it, but from an unbalanced position it is possible to move to either another unbalanced position or a balanced one. We prove this below:

Lemma 2 It is always possible to make the nim-sum 0 on your turn if it wasn't already 0 at the beginning of your turn.

Let d be the position of the most significant bit in s (defined above). Now choose a heap x_k such that it's most significant bit is also in position d (one must always exist, the most significant bit of s must come from the most significant bit of any of the nim heaps). Now choose to make the new value of the heap $y_k = s \oplus x_k$ by removing $x_k - y_k$ stones from the heap. Now the new nim-sum is:

$t = s \oplus x_k \oplus y_k$(from above)
$= s \oplus x_k \oplus x_k \oplus s$
$= s \oplus s \oplus x_k \oplus x_k$
$= 0$

Proof Now we prove the original theorem. If you start off by making your first move so that the Nim-sum is 0, then on each turn your opponent will disturb the sum, and you will in turn set it back to 0. By lemma 1, the opponent has no choice but to disturb the sum, and by lemma 2 you can always set it back to 0. Eventually on your turn there will be no stones left with a nim-sum of 0, meaning that you are the winner! Of course, if the nim-sum starts off at 0 and you go first, then you must hope for your opponent to make a mistake, since he will have the winning strategy.

There is only a slight modification of this strategy for misere play. Follow the same strategy as above until there are only heaps of size 1 left. In normal play the strategy would be to leave an even number of heaps of size 1, but in misere play just be sure to leave an odd number of heaps of size 1 so that your opponent is stuck with the very last one.

From our analysis above, we can see that any nim position in which the nim-sum of the heaps is 0 is a P-position, else the position is an N position.

Practical Strategy

Here are a couple things to keep in mind while playing:

- Whenever possible, reduce the heaps to two non-zero heaps containing the same number of coins each. This obviously has a nim-sum of 0. Now just mimic your opponent's move each time on the opposite heap to keep the two heaps equal until you are able to take the final coin.

- Since doing binary addition is kind of hard to do in your head for large numbers, a more feasible strategy is often needed. An easy way to think about making the nim-sum 0 is to always leave even subpiles of the powers of 2, starting with the largest power possible, where a subpile is a pile group of coins within a nim-heap. So for example, leave an even number of subpiles of 2, 4, 8, 16, etc. Any time there are an even number of piles of each power of 2, the nim-sum must be 0.

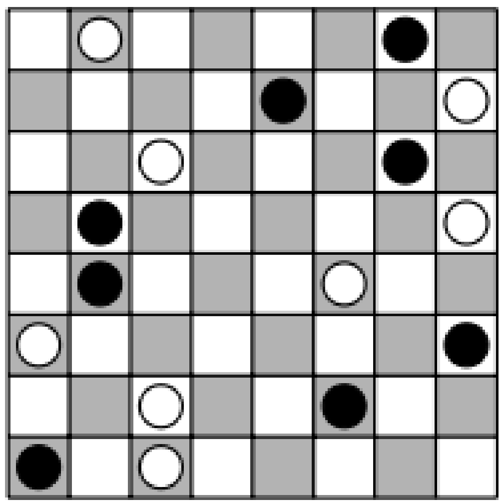
Poker Nim

This game is played the same as regular Nim, but a player can now have the option on his turn of either adding more chips to a heap or subtracting chips from a heap. We call the heaps in such a game ***bogus nim heaps***. What is the winning strategy in this game?

A closer look reveals that this is just the same game as regular Nim. Any time a player adds chips to a pile, the next player can exactly reverse the move and return the game to its original position. In this way, adding chips is a *reversible* move. So just play regular Nim, but any time your opponent adds chips, just remove the same amount on your turn.

Try Northcott's Game below:

A position in Northcotts game is a checkerboard with one black and one white checker on each row. "White" moves the white checkers and "Black" moves the black checkers. A checker may move any number of squares along its row, but may not jump over or onto the other checker. Players move alternately and the last to move wins.



Note that this game is neither impartial nor must it end in finite time, but knowin the strategy for Nim will allow you to win the game. Try the game before looking at the solution at the end of these notes.

Below we'll show that any impartial game is the same as a bogus Nim heap.

Sprague-Grundy Theorem

Now we'll use Nim to help us derive the fundamental theorem of impartial games.

From Games to Graphs

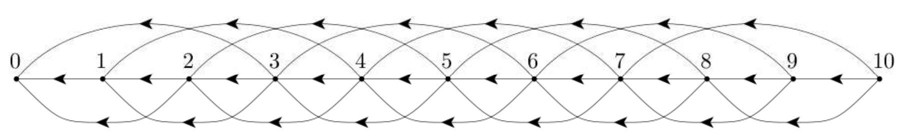
To further analyze impartial games we will put the games on a graph as follows:

A game consists of a graph $G = (X, F)$ where

- X is the set of all possible game positions
- F is a function that gives for each $x \in X$ a subset of possible x's to move to, called **followers**. If $F(x)$ is empty, the position x is terminal.
- The start position is $x_0 \in X$. So player 1 moves first from x_0.
- Players alternate moves. At position x, the player chooses from $y \in F(x)$.
- The player confronted with the empty set $F(x)$ loses.

We will only look at graphs that are **progressively bounded** , meaning that from every start position x_0, every path has finite length. In other words, the graph is finite and has no cycles.

The following is an example of the game graph for the "Subtraction Game." In this game you start with a set number (in this example, 10) and take away any number of coins up to k (in this example, 3). The person that makes the total number of coins 0 on their turn wins.



The Sprague-Grundy Function

The ***Sprague-Grundy function*** of a graph $G = (X, F)$ is a function g defined on X that takes only non-negative integer values and is computed as follows:

$$g(x) = min\{n \geq 0 : n \neq g(y) \text{ for } y \in F(x)\}$$

Uh... English please? In words, the Sprague-Grundy (which from now on I'll abbreviate as SG) is the smallest non-negative value not found among the

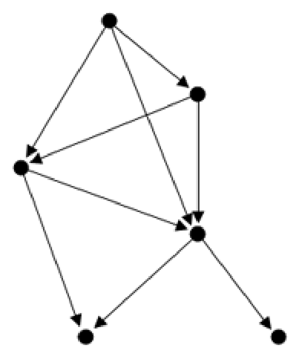
SG values of the followers of x. This is known formally as the **mex** function, meaning Minimum Excluded Value. Below are some practice examples:

```
mex({2,4,5,6}) = 0
mex({0,1,2,6}) = 3
```

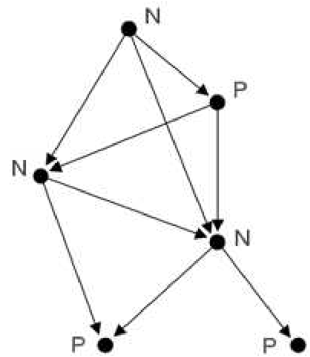
So the SG function can be rewritten as follows:

$$g(x) = mex\{g(y) : y \in F(x)\}$$

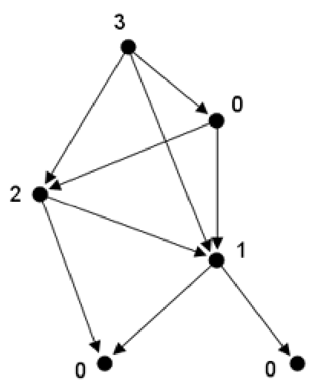
Notice this function is defined recursively. That is, the definition of $g(x)$ uses g itself. So we'll need some base cases. Set all terminal nodes x to have $g(x) = 0$. Then any nodes that have only terminal nodes as followers have $g(x) = 1$. In this way we can work our way through the graph until all nodes are assigned an SG value. Try it on the graph below:



We'll first start by letter each position as being either "N" or "P". Label every node with no outgoing edges as P, since it is an endgame and by definition a P position. Every node pointing to a P node must be N, so go ahead and label those too. Now label all the positions that only go to N positions as P. You should end up with the following labels:



Now we can label the Sprague-Grundy values of each node. First wet all the terminal positions (all of which should have P's right now) to 0 as the base cases. Since the SG value of nodes is just the mex function defined above, any node pointing *only* to terminal nodes must be 1. So label the one node pointing to the two terminal positions as 1. The node pointing only to 0 and 1 valued nodes can be labeled with a 2, since it is the minimum integer not in the set $\{0, 1\}$. But the node pointing to 2 and 1 can be labeled 0 since 0 is the smallest integer not in the set of its followers, 2 and 1. Finish off the numbering by labeling the top node with 3 to obtain the following numbering:



Notice that all vertices that have an SG value of 0 are P positions! All others are N. Sound familiar? So it looks like a good strategy on a game that can be represented in such a graph would be to move to a vertex with $g(x) = 0$.

Let's look at another example game, called the 21 subtraction game. You and a friend start with 21 coins. You take turns taking up to 3 coins away at a time. The person to take the last coin wins.

We can see that the position with 4 coins left is a P position. The next player must reduce to the pile to some numer within the range 1-3, and so when the player after him moves he can take all the coins. Below are the SG numbers for each position:

```
x    0  1  2  3  4  5  6 ...
g(x) 0  1  2  3  0  1  2 ...
```

In other words, the SG function of this game is $g(x) = x \bmod 4$.

For the game of Nim with one heap, the function is just $g(x) = x$.

Adding Games (Graphs)

One advantage of using the SG function is that we can break up games into smaller parts and then *add* their graphs together to make one big game. We call this addition the ***disjunctive sum*** of two games. So to take the disjunctive sum of games G and H, players can move on their turn in either the game G or the game H, and the entire game is over when both G and H are at terminal positions. We can add any n games G_i together as follows:

To sum the games $G_1 = (X_1, F_1), G_2 = (X_2, F_2), \ldots G_n = (X_n, F_N)$,
$G(X, F) = G_1 + G_2 + \ldots + G_n$ where:

- $X = X_1 \times X_2 \times X_3 \ldots \times X_n$, or the set of all n-tuples such that $x_i \in X_i \forall i$
- The maximum number of moves is the sum of the maximum number of moves of each component game

For example, 3 pile Nim is just the sum of 3 games of 1 pile Nim.

Sprague-Grundy Theorem

Now we are actually ready to state the theorem, which says that the SG function for a sum of games on a graph is just the Nim sum of the SG functions of its components.

If g_i is the Sprague-Grundy function of G_i , $i = 1 \ldots n$, then $G = G_1 + \ldots + G_n$ has Sprague-Grundy function $g(x_1 \ldots x_n) = g_1(x_1) \oplus g_n(x_n)$.

Proof: (replicated from the reading) Let $x = (x_1 \ldots x_n)$ be an arbitrary point of X. Let $b = g_1(x_1) \oplus \ldots \oplus g_n(x_n)$. We are to show two things for the function $g(x_1 \ldots x_n)$:

1. For every non-negative integer $a < b$, there is a follower of $(x_1 \ldots x_n)$ that has g-value a.
2. No follower of $(x_1 \ldots x_n)$ has g-value b.

Then the SG-value of x, being the smallest SG-value not assumed by one of its followers, must be b.

To show (1), let $d = a \oplus b$, and k be the number of digits in the binary expansion of d, so that $2^{k-1} \leq d < 2^k$ and d has a 1 in the kth position (from the right). Since $a < b, b$ has a 1 in the kth position and a has a 0 there. Since $b = g_1(x_1) \oplus \ldots \oplus g_n(x_n)$, there is at least one x_i such that the binary expansion of $g_i(x_i)$ has a 1 in the kth position. Suppose for simplicity that $i = 1$. Then $d \oplus g_1(x_1) < g_1(x_1)$ so that there is a move from x_1 to Some x_1' with $g_1(x_1') = d \oplus g_1(x_1)$. Then the move from $(x_1, x_2 \ldots x_n)$ to $(x_1', x_2 \ldots x_n)$ is a legal move in the sum, G, and $g_1(x_1') \oplus g_2(x_2) \oplus \ldots \oplus g_n(x_n) = d \oplus g_1(x_1) \oplus g_2(x_2) \oplus \ldots \oplus g_n(x_n) = d \oplus b = a$.

Finally, to show (2), suppose to the contrary that $(x_1 \ldots x_n)$ has a follower with the same g-value, and suppose without loss of generality that this involves a move in the first game. That is, we suppose that $(x_1', x_2 \ldots x_n)$ is a follower of $(x_1, x_2 \ldots x_n)$ and that $g_1(x_1') \oplus g_2(x_2) \oplus \ldots \oplus g_n(x_n) = g_1(x_1) \oplus g_2(x_2) \oplus \ldots \oplus g_n(x_n)$. By the cancellation law, $g_1(x_1') = g_1(x_1)$. But this is a contradiction since no position can have a follower of the same SG-value.

So if we think of the game of 3-heap nim as a sum of 3 individual games, where the SG value of a game of nim is just the size of the heap, then the SG value of the game is indeed the nim-sum of the individual heap-sizes, or SG-values, as we discovered in our strategy for playing Nim.

Example problems: see 4.3, 4.4 in the week's readings. We will discuss several of these examples in class.

More Impartial Games

Try the following games, most of which can be analyzed as being exactly equivalent to games of Nim or sums of games of Nim.

21 Takeaway

Start with a pile of 21 coins. Each player alternates taking anywhere from one to three coins away from the pile. The player to take the last coin wins.

Turning Turtles

A horizontal line of n coins is laid out randomly with some coins showing heads and some tails. A move consists of turning over one of the coins from heads to tails, and in addition, if desired, turning over one other coin to the left of it (from heads to tails or tails to heads). For example consider the sequence of n = 13 coins:

```
T H T T H T T T H H  T  H  T
1 2 3 4 5 6 7 8 9 10 11 12 13
```

One possible move in this position is to turn the coin in place 9 from heads to tails, and also the coin in place 4 from tails to heads.

The player to turn the last coin from heads to tails wins.

Nimble

Nimble is played on a game board consisting of a line of squares labeled: 0, 1, 2, 3, ... A finite number of coins is placed on the squares with possibly more than one coin on a single square. A move consists of taking one of the coins and moving it to any square to the left, possibly moving over some of the coins, and possibly onto a square already containing one or more coins. The players alternate moves and the game ends when all coins are on the square labeled 0. The last player to move wins.

Silver Dollar Game

From John Conway's "On Numbers and Games": This game is played on a semi-infinite strip of squares, with a finite number of coins, no one of which is a Silver Dollar. Each coin is placed on a separate square, and the legal move is to move some coin leftwards (i.e. towards the finite end of the strip), not passing over any other coin, onto any unoccupied square. The game ends when some player has no legal move, because the coins are in a traffic jam at the end of the strip.

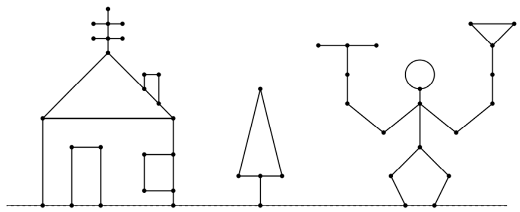
2D Nim

This game is played exactly like Nimbles, except on a 2-dimensional checkerboard. Start with 4 coins placed anywhere on the board. On each turn a player moves a coin either any number of squares downward or any number of squares left. The player to move the last coin to the bottom left corner square wins.

Green Hackenbush

Two players take turns cutting edges on a connected rooted graph or a collection of connected rooted graphs. For our purposes, there will only be finitely many edges on the board, we will associate all the roots with "the ground" and we will call the edges "branches". When a player cuts a branch, the branch dissapears along with any branches that are no longer connected to the ground. The player who cuts the last branch wins.

The folloing is an example game of hackenbush:



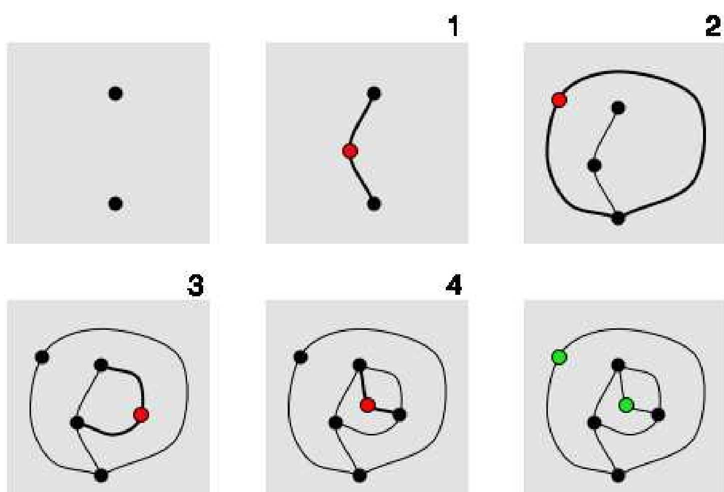
Sprouts

(This game is not so easily related back to Nim, and has not even been complete solved, but nevertheless has some interesting theory behind it so we present it here.)

In the game of sprouts two players start with some number of dots and alternate making moves. A move consists of connecting two dots (called spots) with a curve and marking a new dot anywhere on the curve. The segments of curves connecting two dots are called edges. subject to the following rules:

1. The curves do not intersect (other curves or themselves).
2. No more than three edges emanate from any one spot.
3. A curve may connect a spot to itself.

The player who draws the last curve wins. Below is an example game:



Solutions to Some of the Games

These hints will help you solve the games once you know the secret to Nim.

Northcott's Game

The the number of spaces between the two tokens on each row are the sizes of the Nim heaps. This is the same as Poker Nim. If your opponent increases the number of spaces between two tokens, just decrease it on your next move. Else, play the game of Nim and make the Nim-sum of the number of spaces between the tokens on each row be 0.

21 Takeaway

Always make the size of the heap 0 mod 4 (so 20, 16, 12, etc.)

In general, if players can take at most k coins away, always try to make the size of the pile 0 mod $k + 1$.

Turning Turtles

This game is essentially the same as Nim. If the turtles are numbered 1, ..., n from left to right, then a Turning Turtles position where the coins with heads showing are the ones with numbers a, b, ..., z is equivalent to a Nim position with heaps of size a, b, ..., z. The correspondence between moves is as follows:

1. A Nim move which removes a heap of size a corresponds to flipping over the coin numbered a from heads to tails.
2. A Nim move which decreases a heap from size a to size b, where there was no heap of size b already on the board, corresponds to flipping over the coin numbered a from heads to tails, and flipping over the coin numbered b from tails to heads.

3. A Nim move which decreases a heap from size a to size b, where there was a heap of size b already on the board, corresponds to flipping over the coin numbered a from heads to tails, and flipping over the coin numbered b from heads to tails.

In the last case, the Nim position arrived at, as compared to the Turning Turtles position arrived at, has an extra pair of heaps of size b. However, the presence of a pair of equal heaps, or any number of pairs of equal heaps, never makes any difference to the outcome of a Nim game. This is because whoever had the win before the pairs of heaps were added can continue to play this strategy on the heaps originally present, and when the other player plays on one of the pairs of equal heaps, he can make the same play on the heap equal to the heap the other player moved on. In this way he is guaranteed the last move in the pairs of equal heaps, and as he also has the last move on the heaps originally present, he has the last move in the overall game.

Nimble

A coin on square n is the same as a nim-heap of size n.

Silver Dollar Nim

This one is kind of tricky. Starting from the rightmost coin, count the number of squares in alternate spaces between the coins, and let these numbers be the sizes of Nim-heaps.

2D Nim

This is the sum of 2 Nimble (Nim) games: one in the vertical direction and one in the horizontal direction. So a square at (x,y) is a heap of size x in the horizontal game and size y in the vertical game. By the Sprague-Grundy theorem, you can calculate the SG value of the x and y games as $SG(x) + SG(y)$, where $SG(x)$ is just the num sum of all the x coordinates.

Green Hackenbush

(Much more about this game next week)
When played only with “bamboo stalks” (see pictures) each stalk with n

segments is the same as a nim heap of size n.

When played with a forest of "trees" (**Colon Principle**): When branches come together at a vertex, one may replace the branches by a non-branching stalk of length equal to their nim sum.

For general graphs: We fuse two neighboring vertices by bringing them together into a single vertex and bending the edge joining them into a loop. Each loop can now seen as a nim-heap of size 1.

If we have time we'll go over the proof of the Colon Principle. Proof for the fusion principle can be found in volume 1 of *Winning Ways For Your Mathematical Plays*

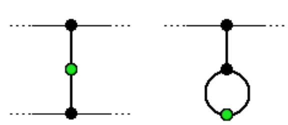
Sprouts

The analysis of this game hasn't actually been solved for arbitrarily large cases. What is known is below:

Suppose that a game starts with n spots and lasts for exactly m moves.

Each spot starts with three lives (opportunities to connect a line) and each move reduces the total number of lives in the game by one (two lives are lost at the ends of the line, but the new spot has one life). So at the end of the game there are 3nm remaining lives. Each surviving spot has only one life (otherwise there would be another move joining that spot to itself), so there are exactly 3nm survivors. There must be at least one survivor, namely the spot added in the final move. So 3nm $\geq$ 1; hence a game can last no more than 3n1 moves.

At the end of the game each survivor has exactly two dead neighbors, in a technical sense of "neighbor"; see the diagram below.



No dead spot can be the neighbor of two different survivors, for otherwise there would be a move joining the survivors. All other dead spots (not neighbors of a survivor) are called pharisees (from the Hebrew for "separated ones"). Suppose there are p pharisees. Then

$$n + m = 3n - m + 2(3n - m) + p$$

since initial spots + moves = total spots at end of game = survivors + neighbors + pharisees. Rearranging gives:

$$m = 2n + p/4$$

So a game lasts for at least 2n moves, and the number of pharisees is divisible by 4.

Real games seem to turn into a battle over whether the number of moves will be m or m+1 with other possibilities being quite unlikely. One player tries to create enclosed regions containing survivors (thus reducing the total number of moves that will be played) and the other tries to create pharisees (thus increasing the number of moves that will be played).

** Code Examples **

For those interested, I will try to include some sort of coding example each week. These sections will be denoted by **'s in the coursenotes.

The problem is to code the normal play game of Nim while giving each player what strategy they should use. Attempt it yourself before looking at the solution posted online.

Also of interest: hackenbush program:
http://cgt.calculusfairy.com/Software/VisualHackenbush/

This program evaluates hackenbush positions. Try making some positions, trying to evaluate their SG value, and then checking with the program. (The program also includes red-blue hackenbush which we'll cover next week with surreal numbers).