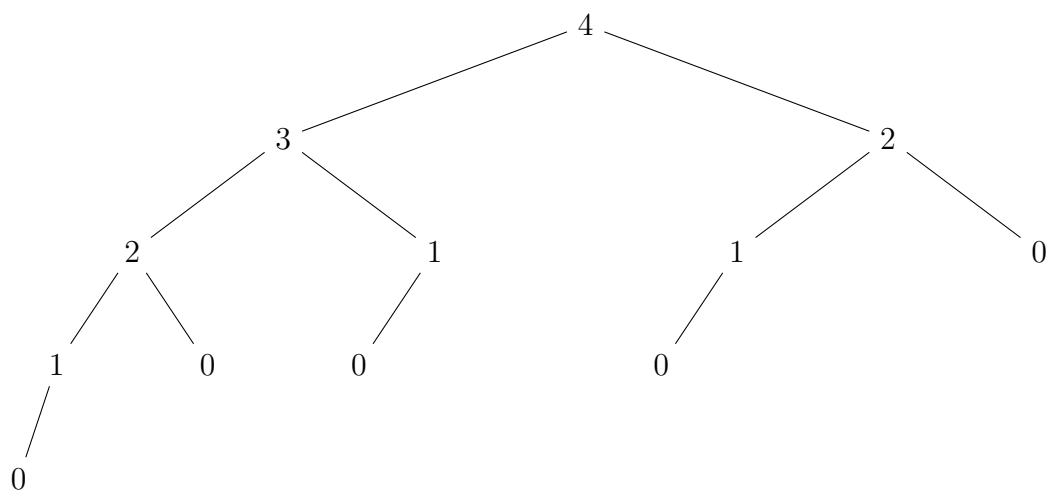
MATH CIRCLES MARCH 4 - NOTES

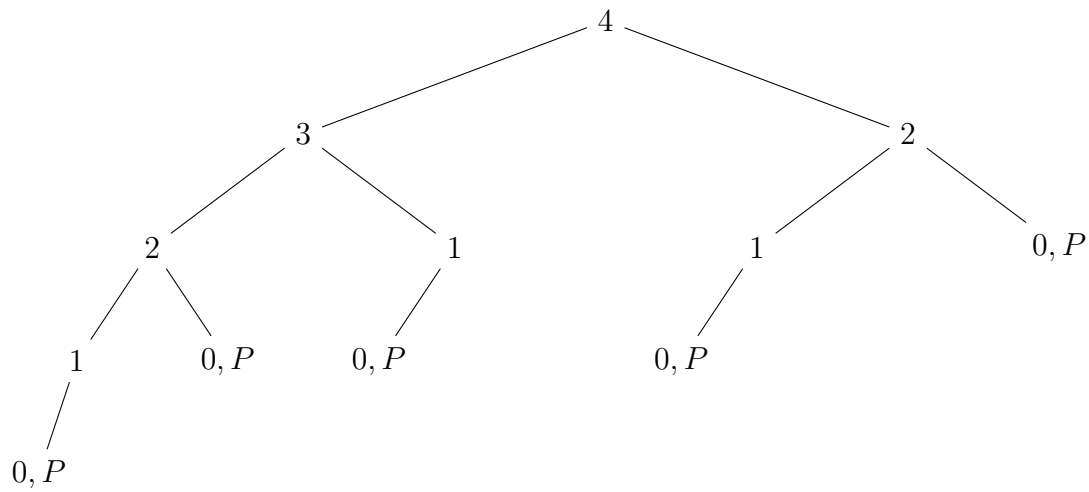
IAN PAYNE

1. Game Trees, P, and N

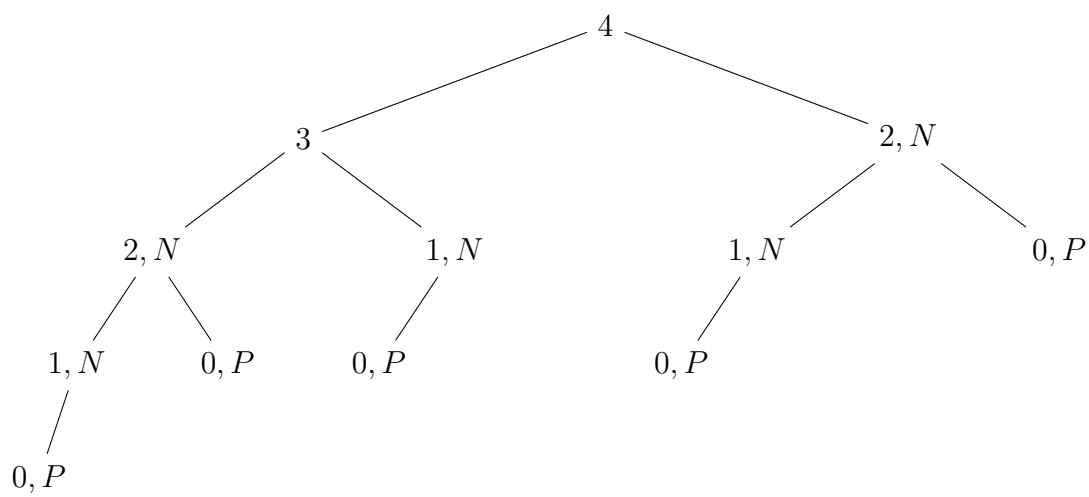
A game tree is a diagram that is useful for analyzing game strategy. Basically, it is a flow chart that follows through all possible moves. It doesn't take long to notice that they get very large very fast, so sometimes we will use a partial game tree. That is, we will stop branching out when we hit a situation that is already understood. The best way to understand a game tree is by example. In fact, we already drew a game tree in the Feb. 25 notes. The game was that there was a pile of stones, and players were allowed to remove one or two. The game tree is assumed to start at the top, and branch downward. Here it is:



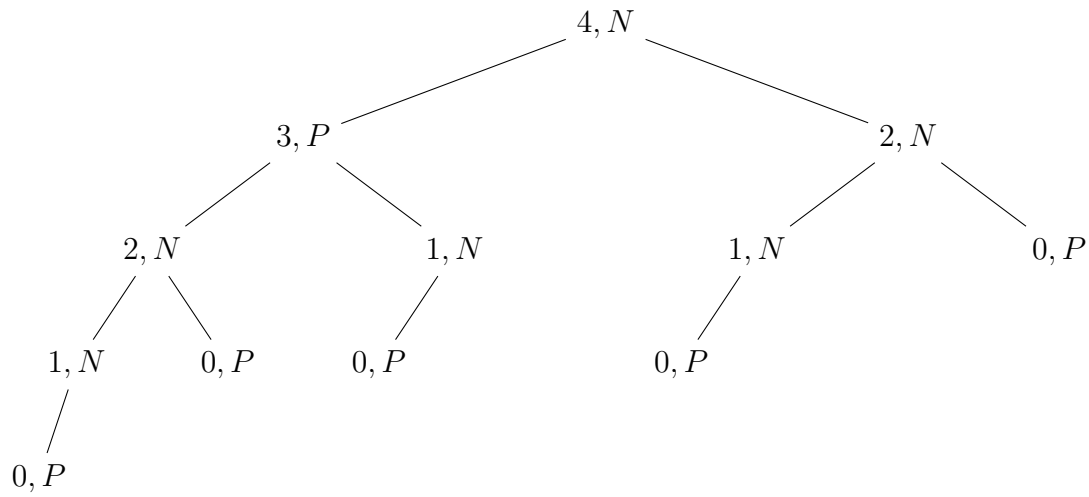
This diagram illustrates every possible sequence of moves that can happen. The idea is to deduce a winning strategy by building up from smaller examples. First, we need some notation. We call each number a state, and move is a change of state. For example, perhaps the state is 4, then the legal moves are to states 3 and 2. We label a state by N if the next person to move has a winning strategy, and P if the person who moved it to that state has a winning strategy. The P stands for previous. Right away, this gives us that the 0 state must be labelled by a P, since the person who moved it to that state wins. In other words, the 0 state is a "previous player win". Let's do this:



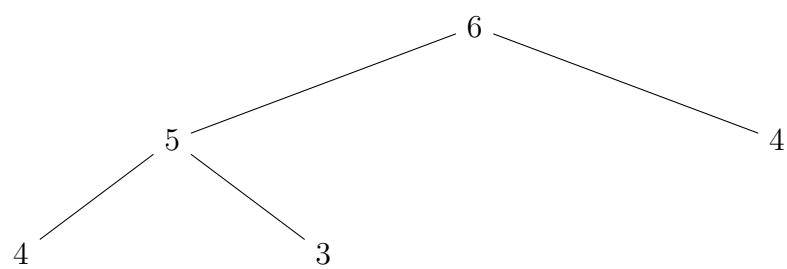
We need to figure out how to label up the branches and build a strategy. The question is, when should a state be labelled by an N, and when should it be labelled by a P. Imagine if a state has a legal move to a state labelled by a P. Remember that this means that they win. Therefore, the NEXT player that moves has a winning strategy. Therefore, if there is **any** legal move from a state to a P labelled state, then it should be labelled by N. On the other hand if there are no moves to a P labelled state, then every move is to an N labelled state. This means that no matter what move is made, the next person will win (the next person after the move). Since there are only two players, the person after the next person is the same as the previous person (read that again, slowly). Therefore, if there are only moves to N labelled states, it must be a previous player win, so labelled with a P. Going back to the diagram, we can immediately label the 1 and 2 states by N since they have a move to a P.



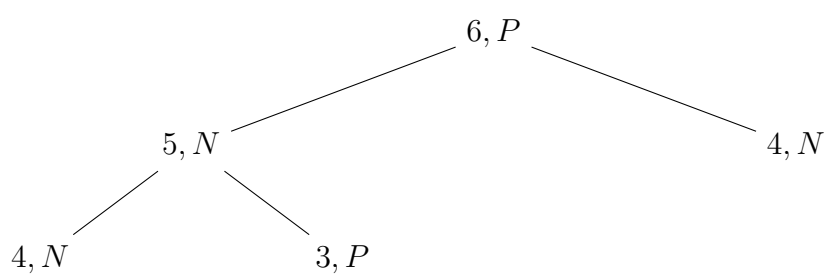
Now we see that 3 has only moves to N states, so it must be a P state. Therefore, 4 has a move to a P labelled state, so it is an N state. Here is the completely labelled tree:



Observe that this technique doesn't just tell us that the first player wins if the game starts with 4 stones, it tells us exactly what the correct move is at each stage. That is, the strategy is to always move to a P state when you can. The way the states were labelled makes this work. If you are faced with an N labelled state, you can move to a P labelled state. Then you opponent is faced with a P labelled state, which leaves them no choice but to move to an N labelled state, so you are again faced with an N labelled state. The process continues and you continue to move to P states, eventually hitting the 0 state. As mentioned before, these diagrams get big fast, and it is unreasonable to draw them. For this, we introduce partial game trees. That is, we only branch until we see a state that we already understand. For example, since we now know the label for 0,1,2,3, and 4, we can draw a partial game tree for 6:



We can label the 3 and the two 4s, and then complete the partial game tree as follows:



This tells us that 6 is a previous player win, which means the second player to move can win. However, it does not tell us the whole strategy. This is the downfall of partial game trees.

2. Binary numbers

Most of the time, we view numbers in base 10. That is, we write them as a sum of powers of 10. For example, the number 347 is really encoding $7 + 4 \cdot 10 + 3 \cdot 10^2$. Similarly, the number 9703 means $3 + 0 \cdot 10 + 7 \cdot 10^2 + 9 \cdot 10^3$. What we are doing is writing a number k as a sum of powers of 10, where we are allowed at most 9 of any given power. In other words, for some k, we write

$$k = a_0 + a_1 10 + a_2 10^2 + \cdots + a_n 10^n$$

where each $0 \leq a_i \leq 9 (= 10 - 1)$. In fact, with this requirement, there is only one way to do this. The decimal or base 10 expansion of k in this case is $a_n a_{n-1} \ldots a_1 a_0$. The binary expansion works in the same way. We want some coefficients $b_0, \ldots, b_m$ for some m such that

$$k = a_0 + a_1 2 + a_2 2^2 + \cdots a_m 2^m.$$

If we insist that, like the base 10 case that $0 \leq a_i \leq 1 = (2 - 1)$, the expansion is again unique. For example,

$$347 = 1 + 1 \cdot 2 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^4 + 0 \cdot 2^5 + 1 \cdot 2^6 + 0 \cdot 2^7 + 1 \cdot 2^8.$$

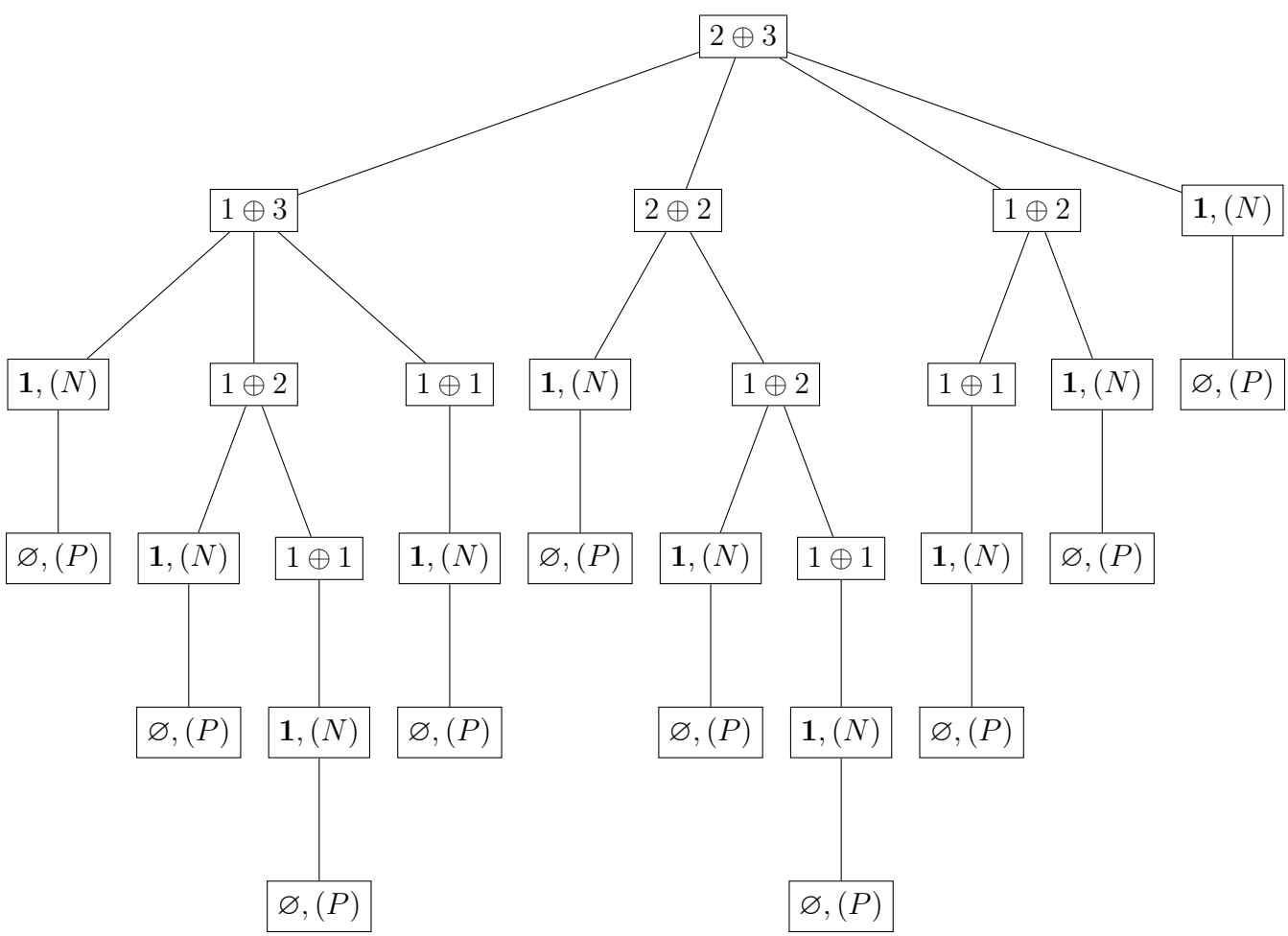
We record these coefficients and say that $347 = 101011011_2$. When it is clear from context that it is base 2, we omit the subscript. To find the binary expansion, first, find the largest power of 2 that is less than or equal to 347. This is 256 or 2^8. Then, subtract 256 from 347 to get 91. Repeat the process and find the largest power of 2 less than or equal to 91. This is 64, or 2^6. Subtracting this from 91 gives 27. Next we find that 16 is the largest power of 2 less than or equal to 27, so we subtract 16 from 27 to get 11. The largest power of 2 less than or equal to 11 is 8, and subtracting it leaves 3. The largest power of 2 less than or equal to 3 is 2, and subtracting gives 1. Finally, the largest power of 2 less than or equal to 1 is 1, and subtracting gives 0. This process will always terminate with a 0, at which point you check to see which powers of 2 were found. They were: 256,64,16,8,2, and 1. The coefficients of these will be 1, and the coefficients of the others will be 0. To work backwards, the number 1110001010_2 is

$$0 \cdot 2^0 + 1 \cdot 2^1 + 0 \times 2^2 + 1 \times 2^3 + 0 \times 2^4 + 0 \times 2^5 + 0 \times 2^6 + 1 \times 2^7 + 1 \times 2^8 + 1 \times 2^9,$$

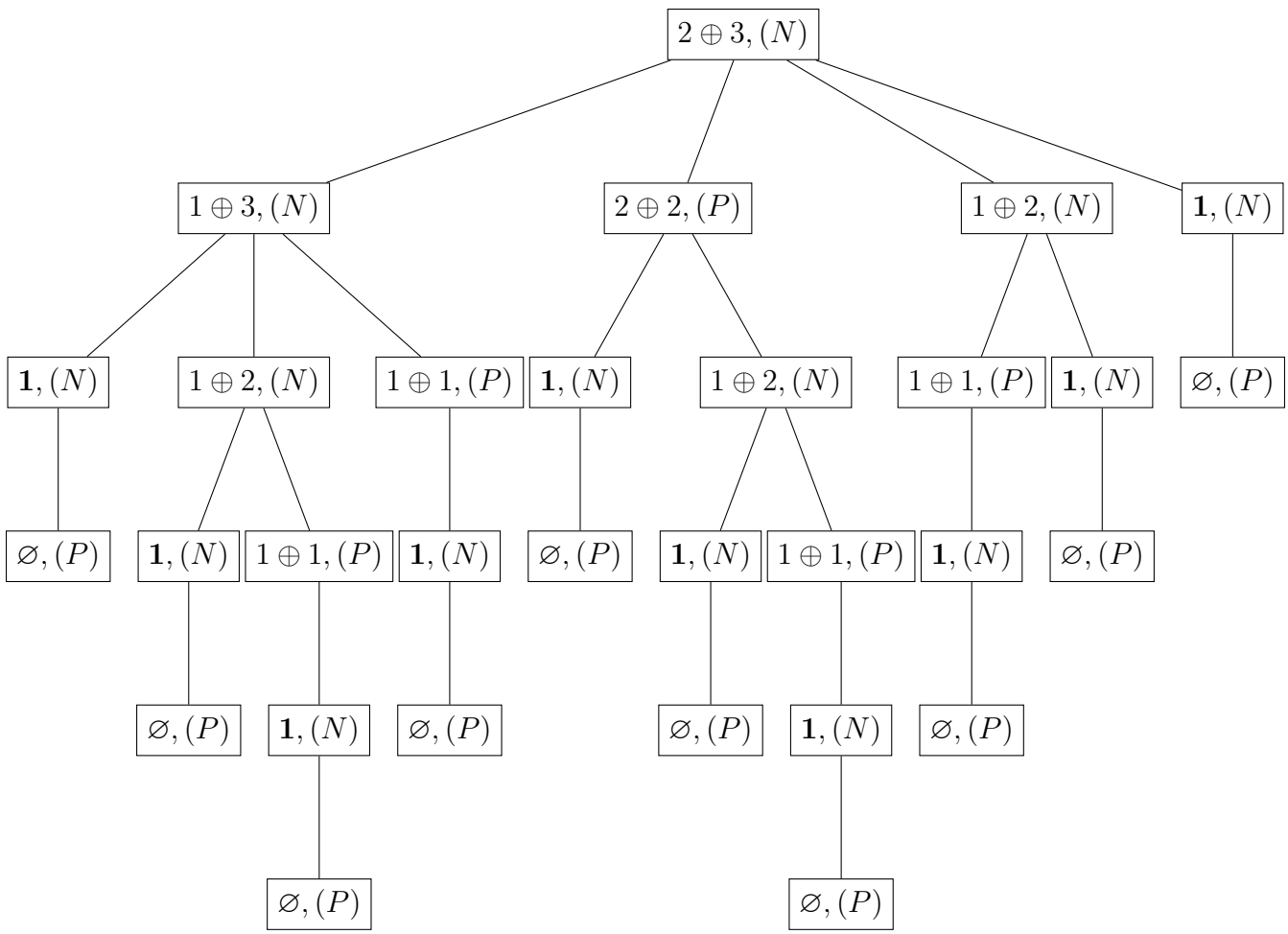
which is better known as 906.

3. Nim

Nim is a simple game that is related to the subtraction games from earlier sections. There are several piles of stones, and two players alternate turns. A legal move is to remove some of the stones from one of the piles. A player must remove at least one stone. The first player that can not move loses. First, we will apply the idea of game trees to get a feel for some small cases. A state will be denoted by $m_1 \oplus m_2 \oplus \cdots \oplus m_n$ and means the game where there are n piles, and they have $m_1, m_2, \ldots, m_n$ stones. For example, $2 \oplus 3$ denotes the game with a pile of 2 stones and a pile of 3 stones. The game with no stones will be denoted by $\varnothing$. It is a previous player win (see the section Game Trees, P, and N for the notation), and it is very important for the same reasons that the empty game was important in other games. One last bit of notation is that we will denote any single pile game by $\mathbf{1}$. This is because these games are all essentially the same. If a player is left with a single pile, they can win in one move. Therefore, $\mathbf{1}$ represents any single pile game, and is a next player win. Here is the start of the labelled game tree for $2 \oplus 3$.



Now we can continue to label the game tree. We see that $1 \oplus 1$ has only a move to $\mathbf{1}$, which is an N state. Therefore, $1 \oplus 1$ is a P state. This makes sense because the only move is to take one of the stones, which means your opponent wins, making it a previous player win. Continuing to label the game tree, we get



We see that this game tree shows that $2 \oplus 3$ is a next player win, and the game tree actually gives an explicit strategy. Using partial game trees and some logical reasoning, you should be able to deduce the following:

- If there are an even number of piles with an even number of piles of each size (eg. $2 \oplus 2 \oplus 5 \oplus 5$,$1 \oplus 1 \oplus 7 \oplus 7 \oplus 7 \oplus 7 \oplus 28 \oplus 28$), it is a previous player win.
- If there are two piles of unequal size, it is a next player win (There is a move to a P state like the ones described in the previous item.)
- If every pile has the same size, it is a previous player win if there are an even number of piles, and next player win if there are an even number of piles.

Next we outline a general strategy that doesn't require cumbersome game trees.

We would like to find a general strategy so that we don't need to draw game trees to figure out what the correct move is. To find it, first consider the following known P states:

- $10 \oplus 10$
- $1 \oplus 2 \oplus 3$
- $1 \oplus 4 \oplus 5$
- If you think about the argument in (e) you should be able to figure out that $3 \oplus 3 \oplus 7 \oplus 7$ is also a P state.

Writing the involved numbers in binary and lining them up gives

$$\begin{array}{c} 10 \oplus 10 \\ \hline 1\ 0\ 1\ 0 \\ 1\ 0\ 1\ 0 \end{array} \qquad \begin{array}{c} 1 \oplus 2 \oplus 3 \\ \hline 0\ 1 \\ 1\ 0 \\ 1\ 1 \end{array}$$

$$\begin{array}{c} 1 \oplus 4 \oplus 5 \\ \hline 0\ 0\ 1 \\ 1\ 0\ 0 \\ 1\ 0\ 1 \end{array} \qquad \begin{array}{c} 3 \oplus 3 \oplus 7 \oplus 7 \\ \hline 0\ 1\ 1 \\ 0\ 1\ 1 \\ 1\ 1\ 1 \\ 1\ 1\ 1 \end{array}$$

If we do the same thing for these known N states

- $4 \oplus 11$
- $1 \oplus 2 \oplus 4$
- $1 \oplus 3 \oplus 4$, and
- $1 \oplus 1 \oplus 6 \oplus 6 \oplus 12$

we get

$$\begin{array}{c} 4 \oplus 11 \\ \hline 0\ 1\ 0\ 0 \\ 1\ 0\ 1\ 1 \end{array} \qquad \begin{array}{c} 1 \oplus 2 \oplus 4 \\ \hline 0\ 0\ 1 \\ 0\ 1\ 0 \\ 1\ 0\ 0 \end{array}$$

$$\begin{array}{c} 1 \oplus 3 \oplus 4 \\ \hline 0\ 0\ 1 \\ 0\ 1\ 1 \\ 1\ 0\ 0 \end{array} \qquad \begin{array}{c} 1 \oplus 1 \oplus 6 \oplus 6 \oplus 12 \\ \hline 0\ 0\ 0\ 1 \\ 0\ 0\ 0\ 1 \\ 0\ 1\ 1\ 0 \\ 0\ 1\ 1\ 0 \\ 1\ 1\ 0\ 0 \end{array}$$

The thing to notice is that in the P cases, every column has an even number of 1s, and in the N cases, at least one column has an odd number of 1s. This works in general, and gives a strategy. Here it is:

- For the state $m_1 \oplus m_2 \oplus \cdots \oplus m_n$, write the numbers $m_1, m_2, \ldots, m_n$ in binary and right justify them all.
- Count the number of ones in each column. If the number is even for each column, it is a P state. Otherwise, it is an N state.
- If it is an N state, find the rightmost column with an odd number of 1s. Choose a pile with a 1 in that position and remove enough stones from that pile so that every column has an even number of 1s.
- If it is already a P state, there is no good move. Any move will change exactly one pile (this corresponds to changing the digits in exactly one row), and since it is a different number, at least one digit must change. The column in which this change occurs will now have an odd number of 1s, making it an N state.

Here are a few examples of this strategy in action.

(1) $2 \oplus 6 \oplus 17 \oplus 13 \oplus 5$ lined up and written in binary looks like

$$\begin{array}{ccccc} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1. \end{array}$$

We see that there is an odd number of ones in at least one column, so this is an N state and the first player can win. The leftmost column with an odd number of ones is in fact the leftmost column. There is only one row with a one in the leftmost column, so the move has to be made on the 17 pile. In order to make it so that there are an even number of ones in each column, we must replace this row by 01111. This number represents 15, so the correct move is to remove 2 stones from the pile of 17.

(2) For $5 \oplus 5 \oplus 5 \oplus 21 \oplus 16$ we get

$$\begin{array}{ccccc} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0. \end{array}$$

We see that every column has an even number of ones, so it is a P position. It is a second player win, so there is no good move for the first player.

(3) From what was discussed earlier, this is the case where there are an odd number of equal piles. This is an N state, so the correct move is to remove one of the piles entirely. Indeed, if we use the general technique, you will find that it tells you the same answer. Try it!