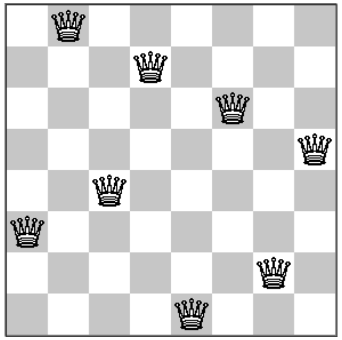
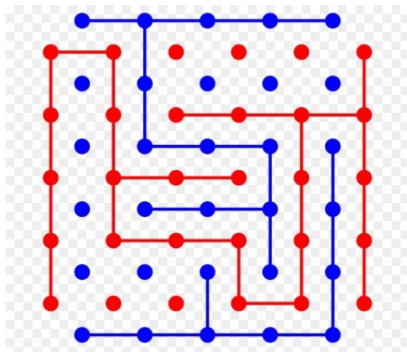
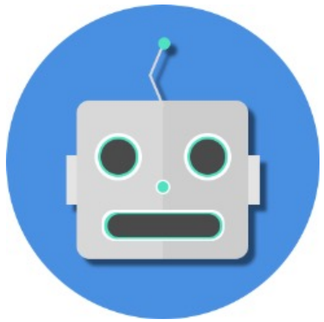
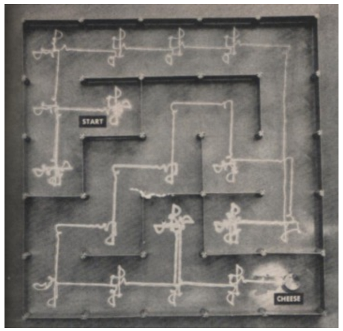
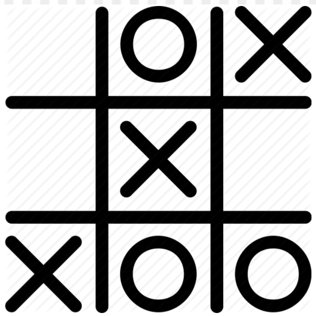
Artificial Intelligence:

Past, Present and Future



Chee Wei Tan

tive feedback system for hunting for particular solutions of the logical equations without an exhaustive search through all possible combinations. This is achieved by elements which sense whether or not a particular logical relation is satisfied. If not, the truth variables involved in this relation are caused to oscillate between their two possible values. Thus, variables appearing in unsatisfied relations are continually changing, while those appearing only in satisfied relations do not change. If ever all relations are simultaneously satisfied the machine stops at that particular solution. Changing only the variables in unsatisfied relations tends, in a general way, to lead to a solution more rapidly than methodical exhaustion of all cases, but, as is usually the case when feedback is introduced, leads to the possibility of continual oscillation. McCallum and Smith point out the desirability of making the changes of the variables due to the feedback unbalance as random as possible, to enable the machine to escape from periodic paths through various states of the relays.

Game Playing Machines

The problem of designing game-playing machines is fascinating and has received a good deal of attention. The rules of a game provide a sharply limited environment in which a machine may operate, with a clearly defined goal for its activities. The discrete nature of most games matches well the digital computing techniques available without the cumbersome analog-digital conversion necessary in translating our physical environment in the case of manipulating and sensing machines.

Game playing machines may be roughly classified into types in order of increasing sophistication:

1. Dictionary-type machines. Here the proper move of the machine is decided in advance for each possible situation that may arise in the game and listed in a "dictionary" or function table. When a particular position arises, the machine merely looks up the move in the dictionary. Because of the extravagant memory requirements, this rather uninteresting method is only feasible for exceptionally simple games, e.g., tic-tac-toe.
2. Machines using rigorously correct playing formulas. In some games, such as Nim, a complete mathematical theory is known, whereby it is possible to compute by a relatively simple formula, in any position that can be won, a suitable winning move. A mechanization of this formula provides a perfect game player for such games.
3. Machines applying general principles of approximate validity. In most games of interest to humans, no simple exact solution is known, but there are various general principles of play which hold in the majority of positions. This is true of such games as checkers, chess, bridge, poker and the like. Machines may be designed applying such the principles are not infallible, neither are the machines, as indeed, neither are humans.
4. Learning machines. Here the machine is given only the rules of the game and perhaps an elementary strategy of play, together with some method of improving this strategy through experience. Among the many methods that have been suggested for incorporation of learning we have:
   a) trial-and-error with retention of successful and elimination of unsuccessful possibilities;
   b) imitation of a more successful opponent;
   c) "teaching" by approval or disapproval, or by informing the machine of the nature of its mistakes; and finally
   d) self-analysis by the machine of its mistakes in an attempt to devise general principles.

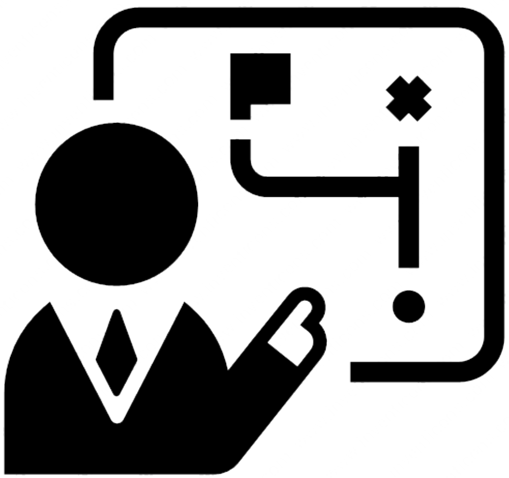
Many examples of the first two types have been constructed and a few of the third. The fourth type, learning game-players, is reminiscent of Mark Twain's comment on the weather. Here is a real challenge for the programmer and machine designer.

Two examples of the third category, machines applying general principles, may be of interest. The first of these is a machine designed by E. F. Moore and the writer for playing a commercial board game known as Hex. This game is played on a board laid out in a regular hexagon pattern, the two players alternately placing black and white pieces in unoccupied hexagons. The entire board forms a rhombus and Black's goal is to connect the top and bottom of this rhombus with a continuous chain of black pieces. White's goal is to connect the two sides of the rhombus with a chain of white pieces. After a study of this game, it was conjectured that a reasonably good move could be made by the following process. A two-dimensional potential field is set up corresponding to the playing board, with white pieces as positive charges and black pieces as negative charges. The top and bottom of the board are negative and the two sides positive. The move to be made corresponds to a certain specified saddle point in this field. To test this strategy, an analog device was constructed, consisting of a resistance network and gadgetry to locate the saddle points. The general principle, with some improvements suggested by experience, proved to be reasonably sound. With first move, the machine won about seventy per cent of its games against human opponents. It frequently surprised its designers by choosing odd-looking moves which, on analysis, proved sound. We normally think of computers as expert at long involved calculations and poor in generalized value judgments. Paradoxically, the positional judgment of this machine was good; its chief weakness was in end-game combinatorial play. It is also curious that the Hex-player reversed the usual computing procedure in that it solved a basically digital problem by an anlog machine.

The game of checkers has recently been programmed into a general-purpose computer, using a "general prin-

C. E. Shannon, Computers and Automata, Proc. of IRE, vol. 41, pp. 1234-1241, 1953

AI Games: 2-Player Game of Strategy



- This lecture looks at two-player games.
- Some games are all about luck. Your winning chance depends on the roll of a die or the cards you've been dealt. But there are other games that require strategy: if you play cleverly, you're guaranteed to win.
- What are some examples of games of strategy?
- A game of strategy thus requires you to inspect the state of the game and design a winning strategy for one of the two players. If one player is a computer, you have AI games.
- This lecture will inform you that a winning strategy can be designed based on a suitable representation of data and knowledge, and can be devised based on simple arithmetic.

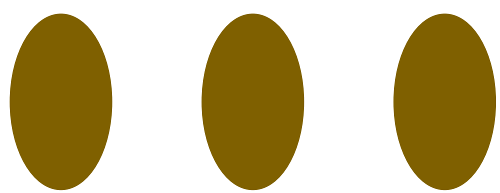
Let’s Play 1-2 NIM Game

We have a heap of beans.

Two players take turns to remove 1 or 2 beans from the heap.

The player who removes the last bean wins.

Let’s start with a heap of 3 beans:



What is your strategy to win?

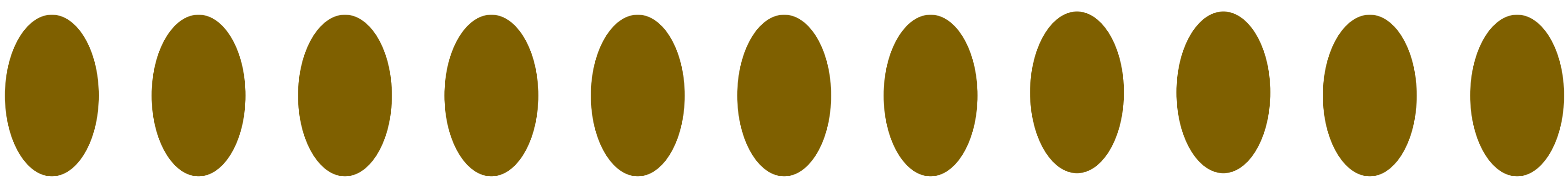
Do you want to be the first or second player?

Let’s Play 1-2 NIM Game

What if there are 11 beans?

Would you choose to take first?

What is your winning strategy?



NIM Game and AI

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
A.I. LABORATORY

January 1970

Artificial Intelligence
Memo No. 254

LOGO
Memo No. 5

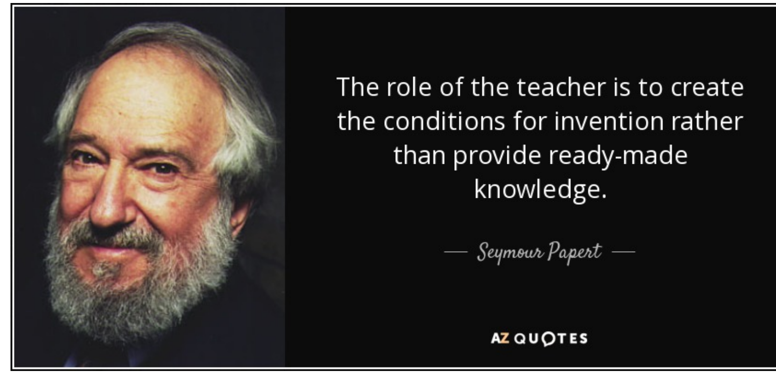
NIM: A Game-Playing Program

Seymour Papert

and

Cynthia Solomon

This work was supported by the National Science Foundation under grant number GJ-1049 and conducted at the Artificial Intelligence Laboratory, a Massachusetts Institute of Technology research program supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored by the Office of Naval Research under Contract Number N00014-70-A-0362-0002.



Seymour Papert (1928 –2016) was a South African-born American mathematician, computer scientist, and educator, who spent most of his career teaching and researching at MIT. He was one of the pioneers of artificial intelligence, and of the constructionist movement in education.

NIM Game and AI

NIM: A Game-Playing Program

by

Seymour Papert and Cynthia Solomon

1.0 Introduction

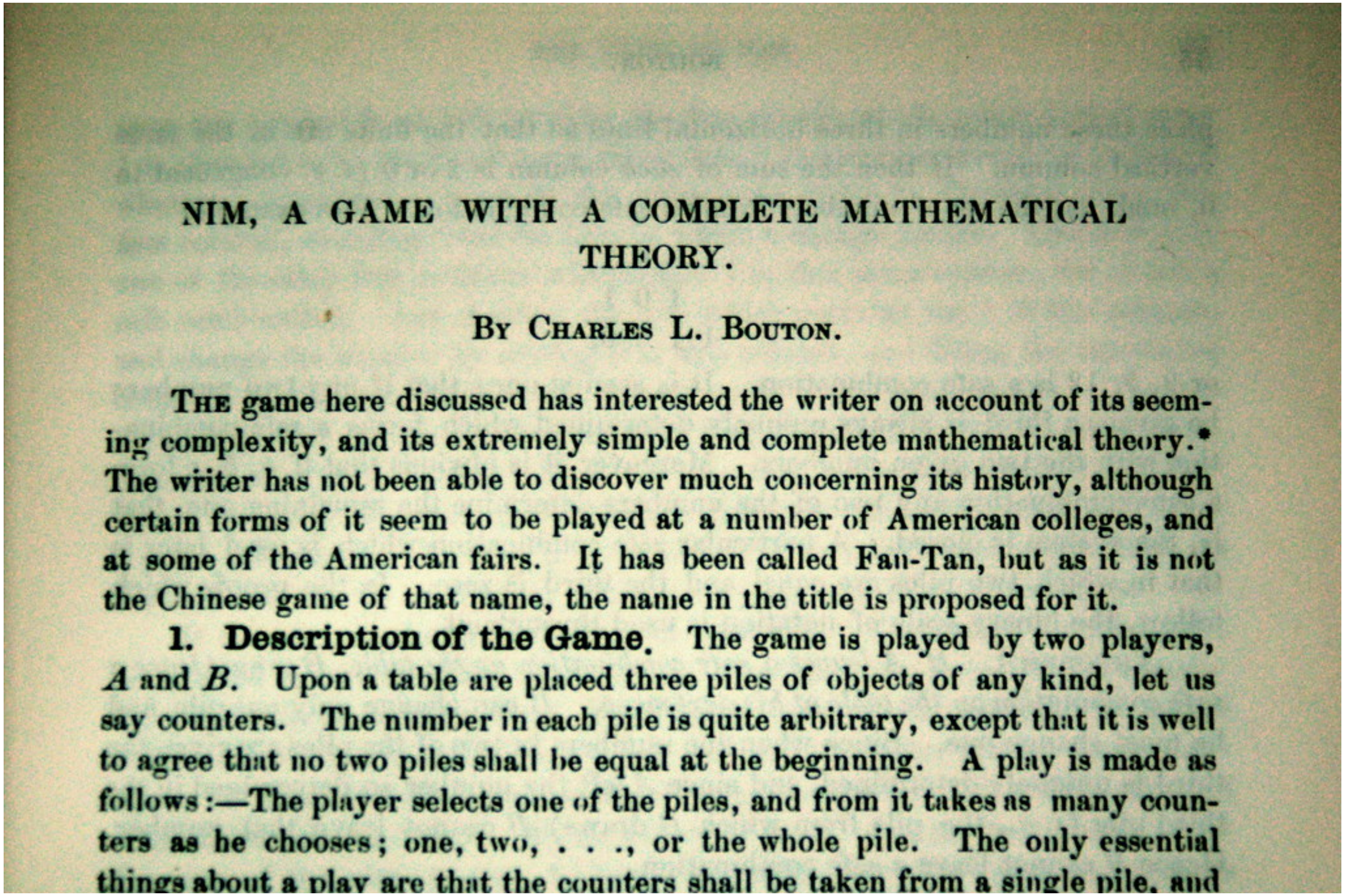
This note illustrates some ideas about how to initiate beginning students into the art of planning and writing a program complex enough to be considered a project rather than an exercise on using the language or simple programming ideas. The project is to write a program to play a simple game ("one-pile NIM" or "21") as invincibly as possible. We developed the project for a class of seventh grade children we taught in 1968-69 at the Muzzey Junior High School in Lexington, Mass.* This was the longest programming project these children had encountered, and our intention was to give them a model of how to go about working under these conditions. To achieve this purpose we ourselves worked very hard to develop a clear organization of sub-goals which we explained to the class at the beginning of the 3 - week period devoted to this particular program. One would not expect beginners to find as clear a subgoal structure as this one; but once they have seen a good example, they are more likely to do so in the future for other problems. Thus our primary teaching purpose was

Example of a Script

```
THE NUMBER OF STICKS IS 8
JON TO PLAY.  WHAT'S YOUR MOVE?
<2

THE NUMBER OF STICKS IS 6
BILL TO PLAY.  WHAT'S YOUR MOVE?
<3

THE NUMBER OF STICKS IS 3
JON TO PLAY.  WHAT'S YOUR MOVE?
<3

JON IS THE WINNER.
```

Later in the project we insist that children consider what happens when a player replies to "WHAT'S YOUR MOVE?" by "5" or "COW". In the beginning we would discourage all but the most competent children from worrying about "funny" answers before getting the program to work with normal answers.

NIM Game and AI



NIM, A GAME WITH A COMPLETE MATHEMATICAL THEORY.

BY CHARLES L. BOUTON.

THE game here discussed has interested the writer on account of its seeming complexity, and its extremely simple and complete mathematical theory.* The writer has not been able to discover much concerning its history, although certain forms of it seem to be played at a number of American colleges, and at some of the American fairs. It has been called Fan-Tan, but as it is not the Chinese game of that name, the name in the title is proposed for it.

1. Description of the Game. The game is played by two players, *A* and *B*. Upon a table are placed three piles of objects of any kind, let us say counters. The number in each pile is quite arbitrary, except that it is well to agree that no two piles shall be equal at the beginning. A play is made as follows:—The player selects one of the piles, and from it takes as many counters as he chooses; one, two, . . ., or the whole pile. The only essential things about a play are that the counters shall be taken from a single pile, and

Bouton, C. L. (1901), "Nim, a game with a complete mathematical theory", *Annals of Mathematics*, 2, **3** (1/4): 35–39, doi:10.2307/1967631

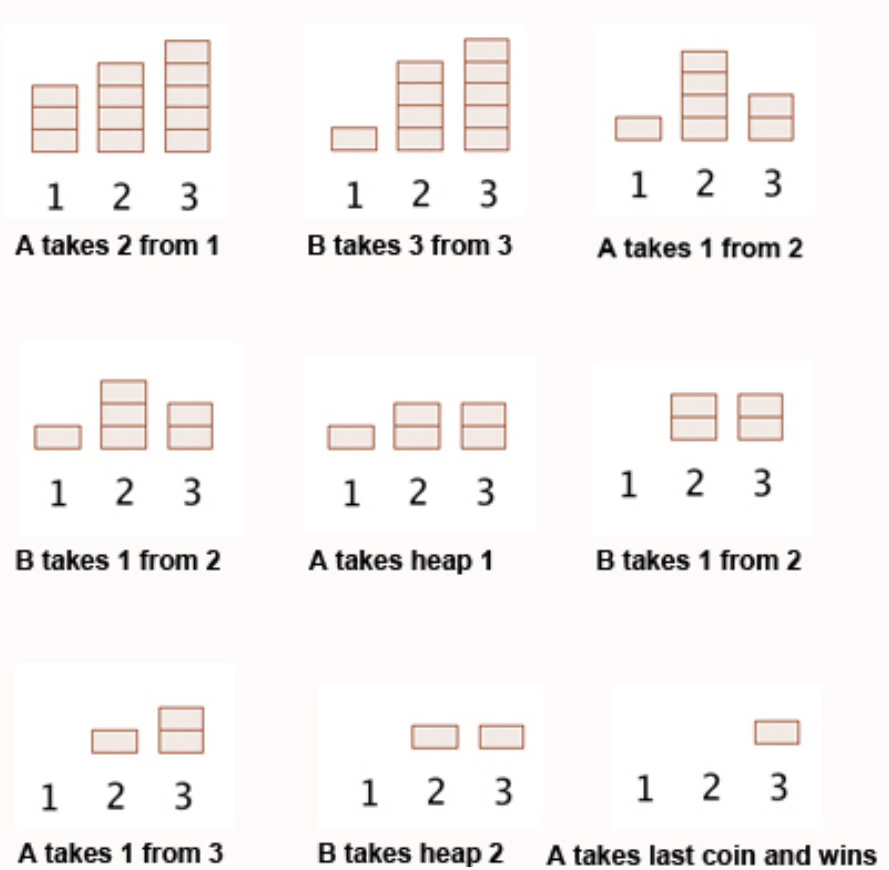
NIM Game and AI

The rules of Nim (multiple rows)

- The Nim is played with a number of coins arranged in heaps: the number of coins and heaps is up to you. There are two players.
- When it's a player's move he or she can take any number of coins from a single heap. They have to take at least one coin, though, and they can't take coins from more than one heap.
- The winner is the player who makes the last move, so there are no coins left after that move.

NIM Game and AI: Example

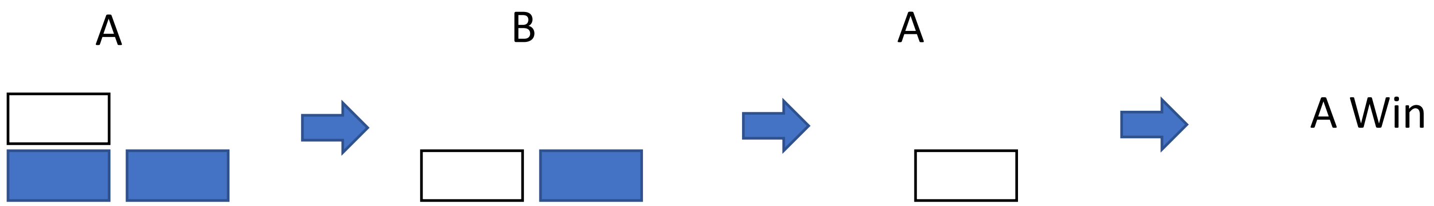
Suppose there are three heaps with three, four and five coins respectively. Here is how the game could develop:



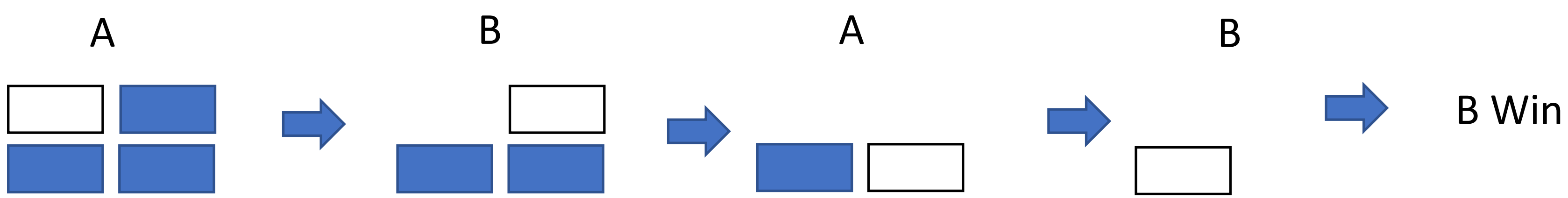
The question of interests is: given a particular configuration of heaps, is there a *winning strategy* for one of the players (either Player A or Player B)? That is, this player is guaranteed to win if he or she *plays the right moves* regardless of the other player's moves?

NIM Game and AI: Example (cont'd)

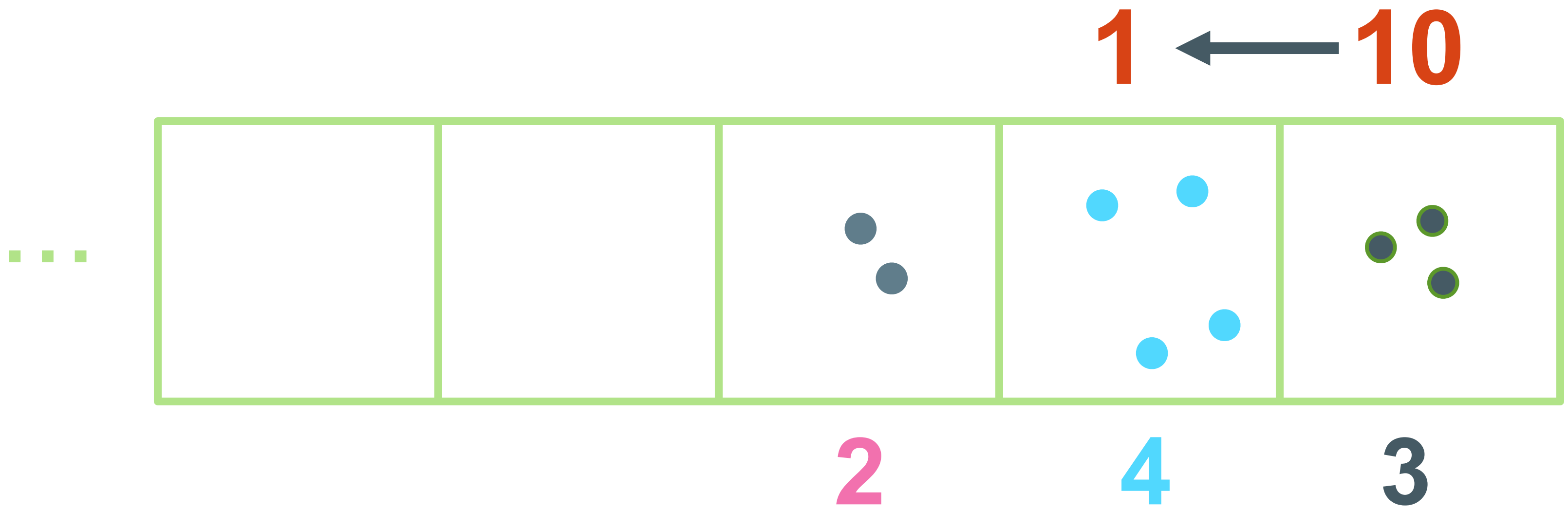
Now suppose that there are two heaps, one of which contains two coins and the other one. Now player A has a winning strategy: take one of the coins in the two-coin heap. This leaves two heaps with a coin each and B to go next. And as we saw in the previous example, this means that A will win.



Let's do one more: suppose that there are two heaps with two coins each. Now player B has a winning strategy. If A takes an entire heap, then B should take the remaining heap and win. If A takes only one coin of one of the heaps, then we are in the same situation as in the previous example, with B to go first. Therefore, B is guaranteed to win if she takes one coin from the two-coin heap.



Speaking Our Language



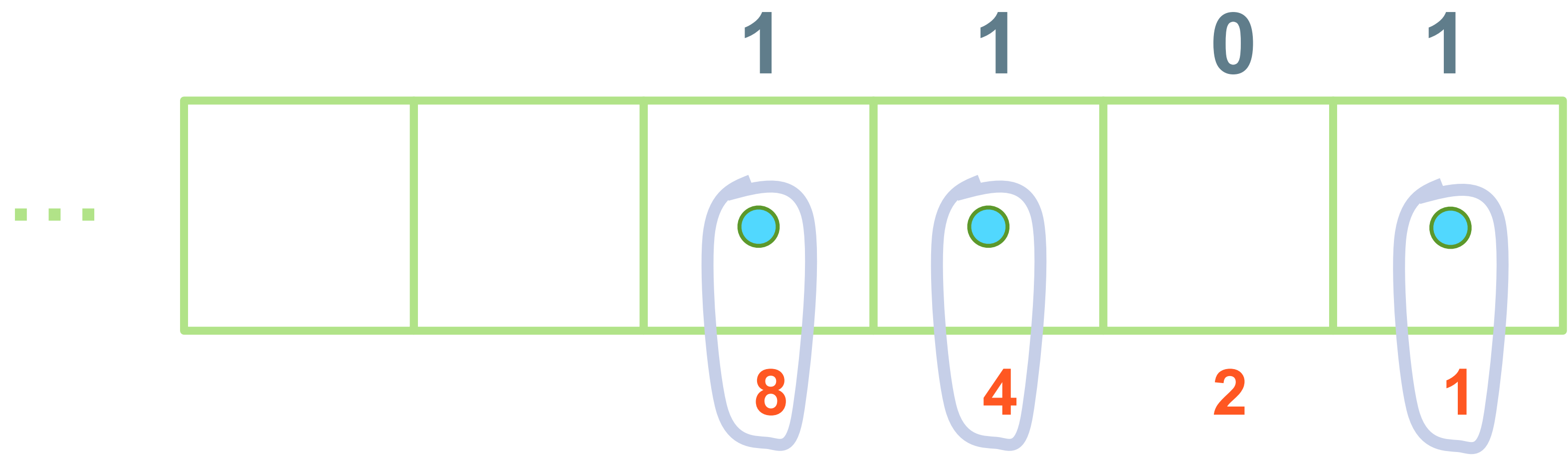
In our language, 243 means:

Two HUNDRED(S)

Forty (four TENS)

Three (ONES)

number: machine code 1 ← 2



$$1 \times 8 = 8$$
$$1 \times 4 = 4$$
$$0 \times 2 = 0$$
$$1 \times 1 = 1$$
$$\overline{13}$$

13 : 1101

NIM Game and AI

- Fact 1: Suppose it's your turn and the Nim-sum of the number of coins in the heaps is equal to 0. Then whatever you do, the Nim-sum of the number of coins after your move will not be equal to 0.

- Fact 2: Suppose it's your turn and the Nim-sum of the number of coins in the heap is not equal to 0. Then there is a move which ensures that the Nim-sum of the number of coins in the heaps after your move is equal to 0.

Tic-Tac-Toe Rule

1. The game is played on a grid that's 3 squares by 3 squares.

2. You are **X**, your friend (or the opponent in this case) is **O**. Players take turns putting their marks in empty squares.

3. The first player to get 3 of her marks in a row (up, down, across, or diagonally) is the winner.

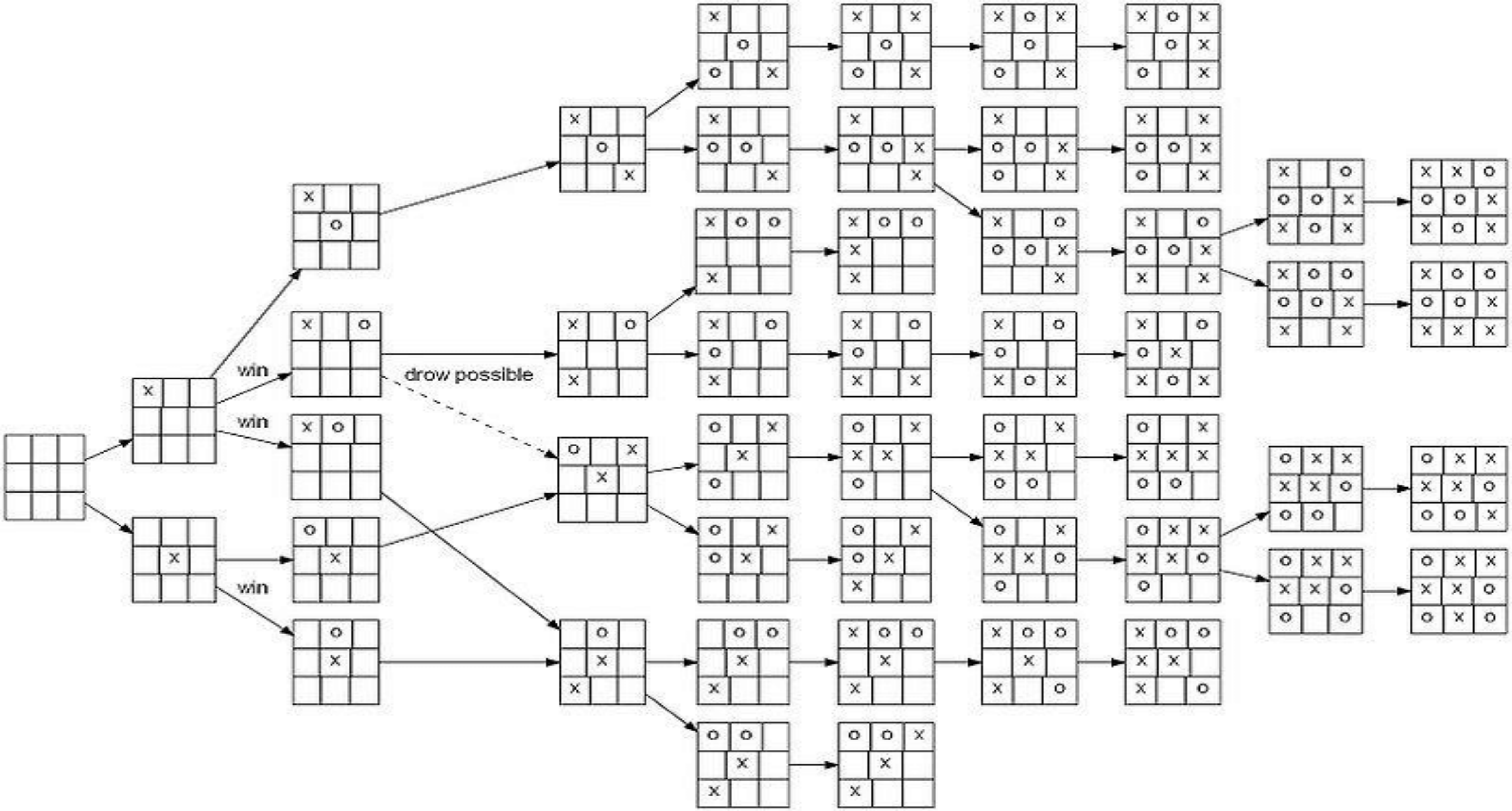
4. When all 9 squares are full, the game is over. If no player has 3 marks in a row, the game ends in a tie or draw.

How to Win Tic-Tac-Toe?

Tic-Tac-Toe is a game of strategy. There are some strategies that can enhance the chances of victory. Often, the first move decides the destiny of the game. The following points briefly sketches how its winning strategies can be designed:

1. The first move of the first player should be in one of the four corners. This is because the opponent (if thinking strategically) can only force a draw.
2. Avoid the edges as a first move (edges are the four boxes that are neither the center nor the corner)
3. Placing the first move at the center will restrict victory from other positions

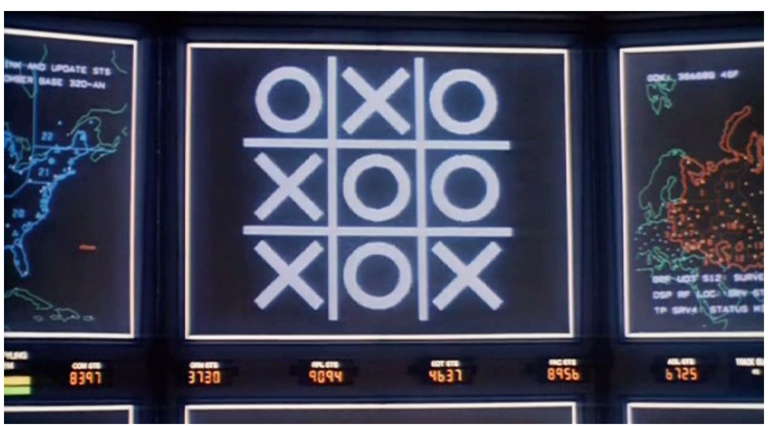
Tic-Tac-Toe Game Tree



The game tree starts with the initial position and consecutive nodes contain all possible distinct ways that the game can be played from each position. Overall a complete game tree for Tic-Tac-Toe has 255,168 leaf nodes.

Summary of AI Games

- Arithmetic plays a key role in AI game winning strategies
- Representation of two-player games of strategy
  - How to design winning strategies depends on how you/computer represent a problem
  - Winning strategies require a clever and principled way of visualizing problem
- Curation of AI games (Nim, Tic-Tae-Toe) and their human-computer interface (HCI)
  - Human aspect of an AI game is important!



WarGames (1983) https://en.wikipedia.org/wiki/WarGames

https://wearethemutants.com/2016/12/15/falkens-maze-game-theory-computer-science-and-the-real-life-cold-war-inspirations-for-wargames