Plan:

1. Introduction

2. Game of NIM

3. Minimax
**Introduction**

The search problems we have studied so far assume that the situation is not going to change. For example, searching a tree or a string. In game playing this is not the case. If you make a move then your opponent is going to move next, so you cannot be certain what the board state will be after your opponents move.

Game playing has a long history within computer science research.

**Poker**

A research history of poker started with von Neumann (1944) in his work on economics (in which game theory was developed). It was recognized that bluffing is a part of any automated poker player. Findler (1977) studied automated poker, over a 20 year period. He concluded that dynamic and adaptive algorithms are required for successful play and static mathematical models were unsuccessful and easily beaten.

Poker is interesting, from a research point of view, as it is a game of imperfect information. Unlike chess and checkers, games such as poker have some information that is unseen. Poker also contains other unknowns such as the playing styles of the other players who may use bluffing (and double bluffing) during the course of the game. Unlike complete information games where the techniques to solve the games (computational power allowing) have been known and understood for a long time (such as mini-max search and alpha-beta pruning), games of imperfect information have not received the same sort of analysis and, doing so, could prove relevant to many other areas such as economics, on-line auctions and negotiating.

Nowadays the best automated poker is called Loki, developed at the University of Alberta. It cannot yet compete in the World Series of Poker.

Here at CMU we also have people (J. Bertomeu) working in this area.
Chess

In 1949, Claude Shannon (a scientist at Bell Laboratories) delivered a paper where he presented his ideas as to how a computer could play chess. He noted that chess has $10^{120}$ possible moves (or unique chess boards).

The bottom estimate. Each player has 16 pieces. There are at least 20 possible moves (8 pawns moving 1 or 2 positions plus two knights). On average, a chess game has 40 moves. Therefore, the total number of moves is $10^{104}$.

The upper estimate $10^{120}$

In 1957, artificial intelligence pioneers Herbert Simon and Allen Newell (they are from CMU) predicted that a computer would beat a human at chess within 10 years. The prediction was part of an attempt "to give some feeling for what computers would mean" to society. And it was lumped with three others -- making music, proving a math theorem, and expressing a psychological theory in computer code.

In the mid-1980s, scientists at CMU started the work that was to become the Deep Blue project. The project moved to IBM in 1989. On May 11, 1997, Kasparov lost a six game match against deep blue with a score of 2.5 - 3.5.
**Game of NIM**

Nim is a two player game in which players alternate making moves. Initially the players are given several piles of sticks. Each pile can have any number of sticks in it. On each turn a player can remove any number of sticks from one pile. The player must remove a minimum of one stick from the pile or may remove the whole pile. The player who picks up the last stick loses (or wins, it depends on variation). The NIM game (1, 2, 2) is presented below (credit to McGill Univ. for the pix)

Each level represents either the MAX score for the player (assuming that he/she make a first move) or the MIN score for the opponent.

We also assign 0s and 1s to each node. A loss is presented by "0" and a win is presented by "1".

We start assigning score from the leaves going up to the root: the scores are assigned to "0" if the leaves are at MAX nodes and the scores are assigned to "1" if the leaves are MIN.

When we back up we choose

the MAXIMUM score of the children if we are on the MAX level;

the MINIMUM score of the children if we are on the MIN level;
In this example, we get 1 for the root, which means that there is a winning strategy. To find that strategy we should follow 1's nodes.

- **Minimax**

  This method was originally published by John von Neumann in 1944. In order to implement minimax you need some way of measuring how good (or bad) your position is.

  In a two player game, the first player moves with MAX score and the second player moves with MIN score. A minimax search is used to determine all possible continuations of the game up to a desired level. A score is originally assigned to the leaf, then by evaluating each possible set of moves, a score is assigned to the upper level by the minimax algorithm. The minimax algorithm performs a preorder traversal and computes the scores on the fly. The same would be obtained by a simple recursive algorithm. The rule is as follows:

  A true minimax search tree is very expensive, since each node must be visited. For games of any size we cannot calculate the entire game tree.

  **Exercise.** Draw a game tree for the tic-tac-toe game. How many nodes in that tree?