Mancala Game

Qian Song
songqian@pdx.edu

OSS course project summer 2016
Github: Qian82
Under MIT License

Outline

- *Zero-Sum-Game Introduction*

- *Mancala game rules*

- *Applied Algorithm*

- *Implementation and Display of game*

What is a zero-sum game?

Zero-sum game is a situation in game theory in which one person's gain is equivalent to another's loss, so the net change in benefit is zero. A zero-sum game may have as few as two players.

Introduction to Mancala

- *The Mancala 'board' is made up of two rows of six pits, each.*

- *Four pieces -- marbles or stones -- are placed in each of the 12 holes. The color of the pieces is irrelevant.*

- *Each player has a 'store' to the right side of the Mancala board.*

Introduction to Mancala

- *The game begins with one player pick up all stones in any one of the pits on his side.*

- *Moving counter-clockwise, the player deposits one of the stones in each pit until the stones run out.*

- *If you run into your own store, deposit it, if you run into your opponent's store, skip it.*

- *If the last piece you drop is in your own store, you get a free turn.*

Introduction to Mancala

- *If the last piece you drop is in an empty pit on your side, you capture the piece and any piece directly opposite.*

- *Always keep captured pieces in your store.*

- *The game ends when all pits on any one side of the mandala board are empty.*

- *The player who still has pieces on his side of board when the game ends captures all of those pieces.*

- *The winner has most pieces in his/her store.*

Introduction to Mancala

- *Still not quite clear on how to play?*

- *Watch this video:*

- *https://www.youtube.com/watch?v=jGM_yntoNsE*

Program and algorithm

Two arrays to represent two rows of the board(each row includes one player's pits), the elements of the arrays represent the number of pieces in each pit.

Program and algorithm

- **_Minimax decision_**

- _An algorithm used to determine the score in a zero-sum game after a certain number of moves, with best play according to an evaluation function. It is a kind of tree-searching algorithm._

- _Evaluation function is a function used by game-playing programs to estimate the value or goodness of a position in the minimax and related algorithm._

Minimax decision
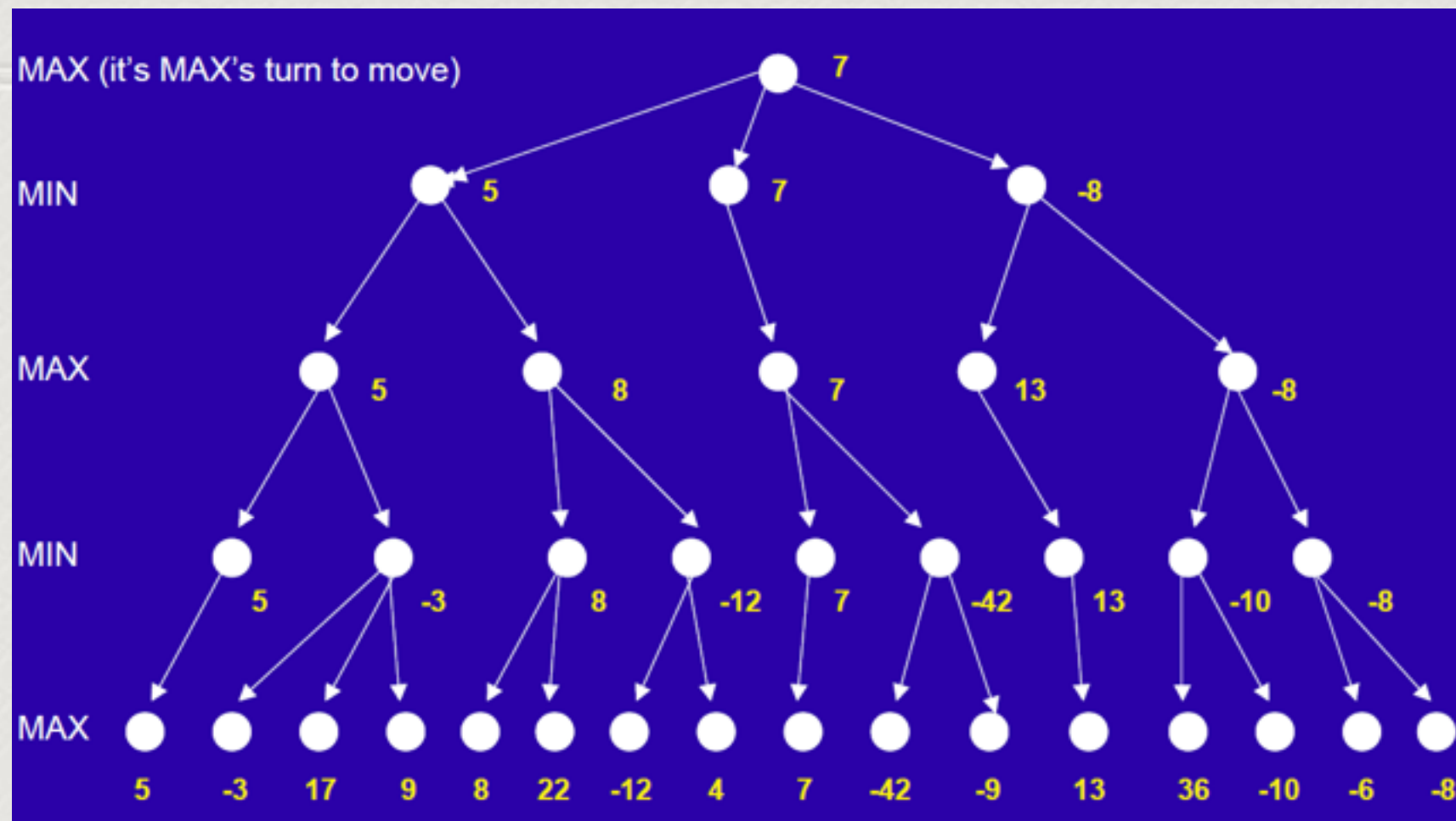
"OK, if I make this move, then my opponent can only make these several moves, and each of those would let me win. So this is the right move to make."

Minimax decision

- *My evaluation function is: evaluation = A[1] - B[8]*

- *In my program, I choose depth = 5*

Implementation and Display

* Python version: terminal interface

* Javascript version: html interface

What have been gained

- *Programming experience in python*

- *programming algorithm (minimax decision)*

- *Javascript, html, CSS, and JQuery*

- *More experience on using github*

Thank You!