ARTIFICIAL INTELLIGENCE

GAME THEORY

By -

Gurpreet Singh
(TEAM.EXE)

CONTENTS

- INTRODUCTION
- APPLICATIONS OF AI
- GAME THEORY
- MINIMAX ALGORITHM
- ALPHA-BETA PRUNING
- A* SEARCH
- HEURISTICS
- CONCLUSION

INTRODUCTION

- Artificial intelligence (AI) is intelligence exhibited by machines.
- In computer science, the field of AI research defines itself as the study of "intelligent agents".
- An agent is any device that perceives its environment and takes actions that maximize its chance of success at some goal.

APPLICATIONS OF AI

1. Speech Recognition - Siri (Apple), Cortana (Microsoft)
2. Handwritting Recognition
3. Robotics
4. Recommendation System
5. Email (Spam Filter)
6. Face Detection
7. Autonomous Driving
8. Route Finding (GPS)
9. Web Search Engines
10. Neural Networks
11. Natural Language Processing
12. Game Theory (Our Main Focus) .... and many more

What is Game Theory ?

- Game theory studies settings where multiple parties (agents) each have
  - different preferences (utility functions),
  - different actions that they can take.
- Each agent's utility depends on all agents's actions
  - What is optimal for one agent depends on what other agents do.
- In many cases, the computer abilities must be toned down to give human players a sense of fairness. Eg.- Racing, Shooting etc.
- Approx. 50% of game project time is spent on building AI.
- Others exmaples of games being Chess, Checkers, Go, BackGammon (involves probability) etc.

TURING TEST

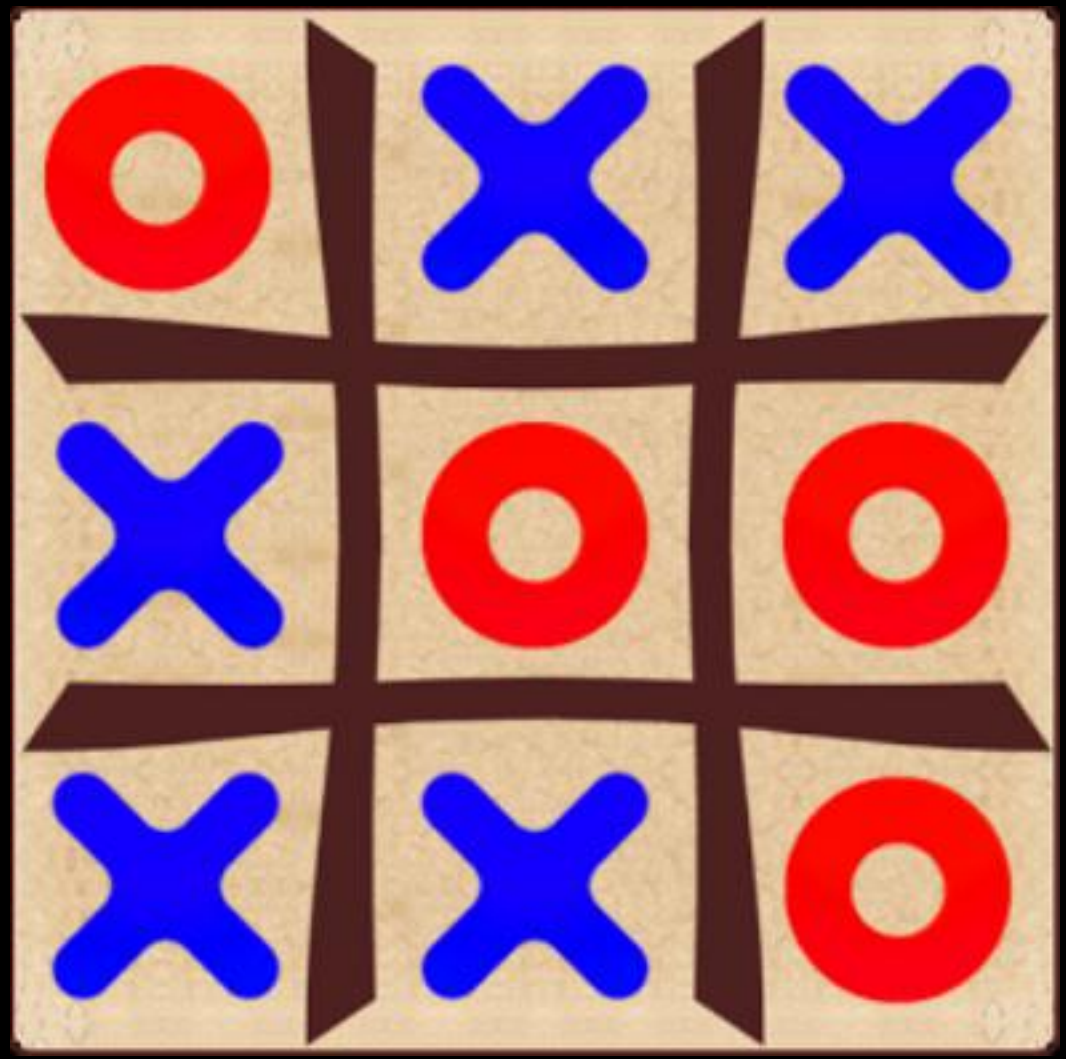
Types of Games

	Deterministic	Chance
Perfect Information	Chess, Checkers, Go	Backgammon, Monopoly
Imperfect Information	Battleships	Bridge, Poker, Scrabble

WHAT WILL WE DO IN THIS WORKSHOP?

- A TIC TAC TOE game bot.
- If time permits : N - Puzzle AutoSolver

TIC TAC TOE



BATTLE OF BOTS
HACKER EARTH

DIFFERENCES IN C and C++

- iostream
- standard namespace
- variable declaration
- variable initialization within loops
- cin and cout instead of scanf and printf
- STL (Standard Template Library)
- Classes (Not required for this workshop)

RECURSION

TREE

TREE

TREE

- A tree is a data structure made up of nodes or vertices and edges without having any cycle.
- Important Terms-
  - Node
  - Root
  - Parent
  - Child
  - Depth
  - Back to Recursion
  - A little bit about Graph

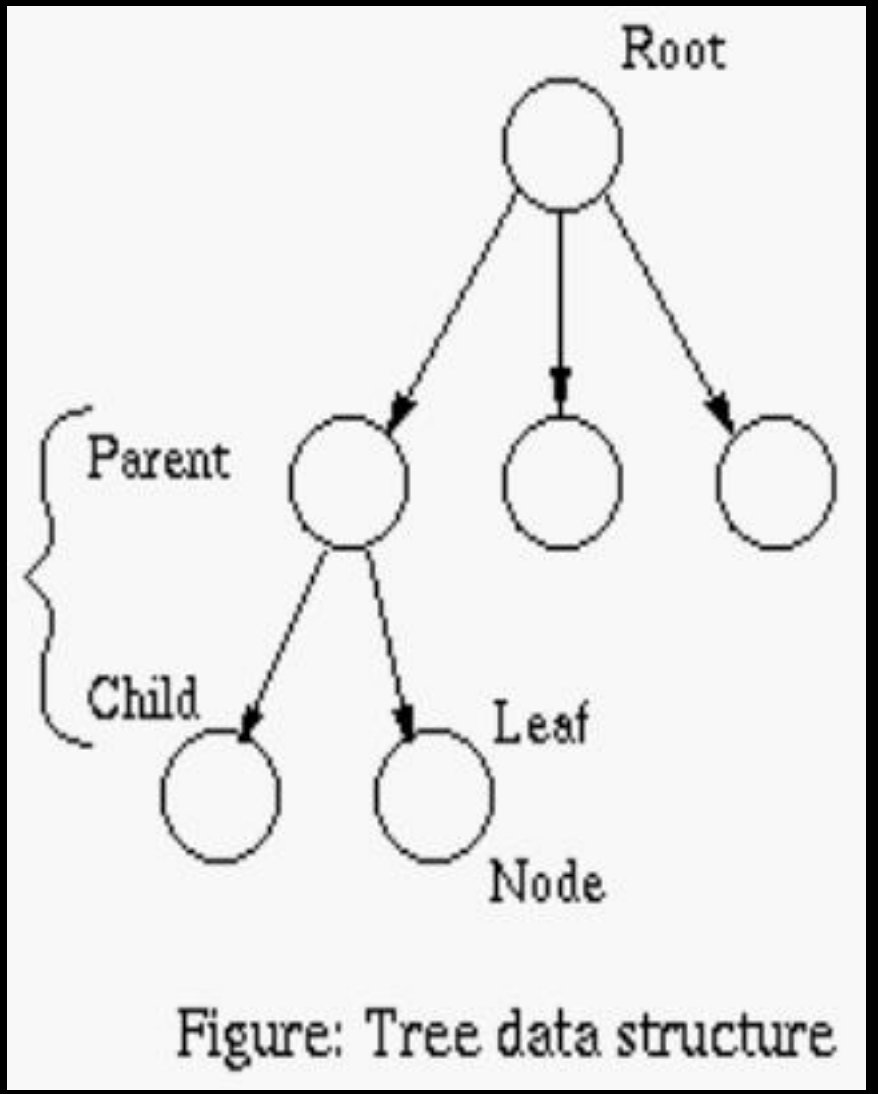


Figure: Tree data structure

TREE TRAVERSAL

- In computer science, tree traversal (also known as tree search) is a form of graph traversal and refers to the process of visiting (checking and/or updating) each node in a tree data structure, exactly once.

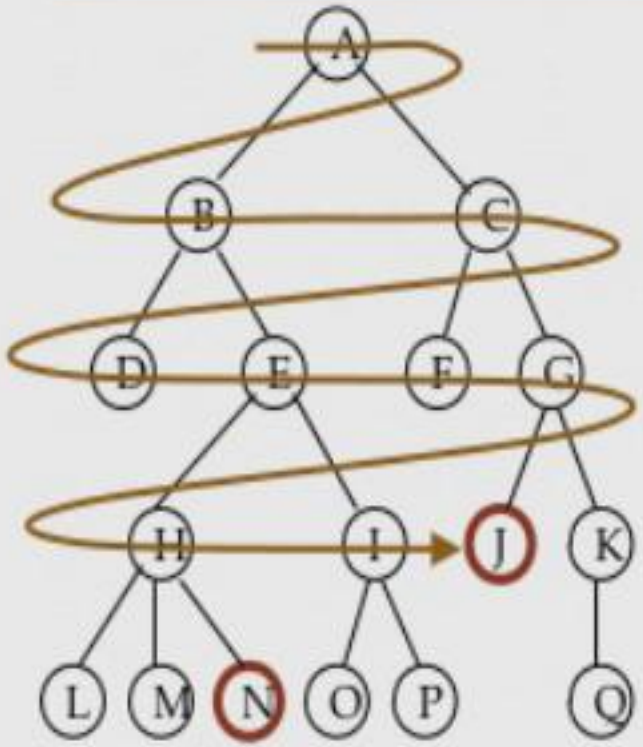
- Array Traversal.
- Meaning of Traversal.

- Methods of Tree Traversal.

BFS

Data Structure used - Queue

Breadth First Searching

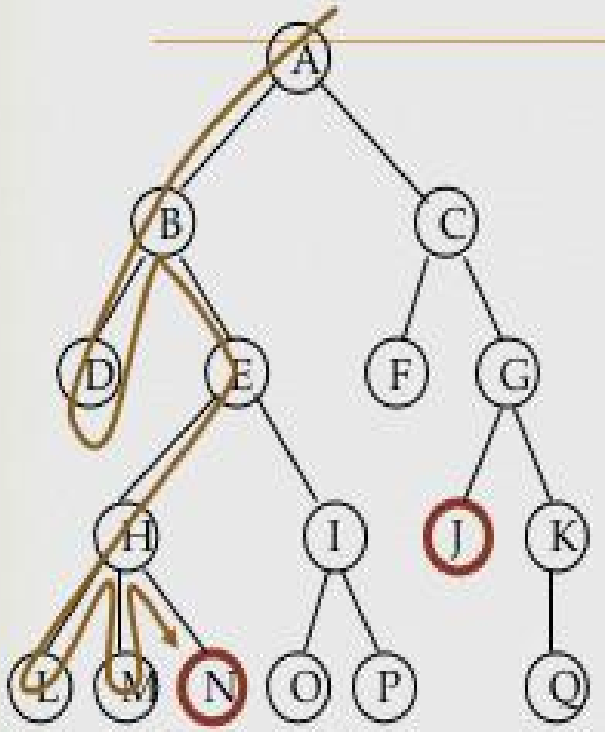


- A breadth-first search (BFS) explores nodes nearest the root before exploring nodes further away.
- For example, after searching A, then B, then C, the search proceeds with D, E, F, G.
- Node are explored in the order A B C D E F G H I J K L M N O P Q.
- J will be found before N.

DFS

Data Structure used - Stack.

Depth-first searching



- A depth-first search (DFS) explores a path all the way to a leaf before backtracking and exploring another path.
- For example, after searching A, then B, then D, the search backtracks and tries another path from B.
- Node are explored in the order A B D E H L M N I O P C F G J K Q.
- N will be found before J.

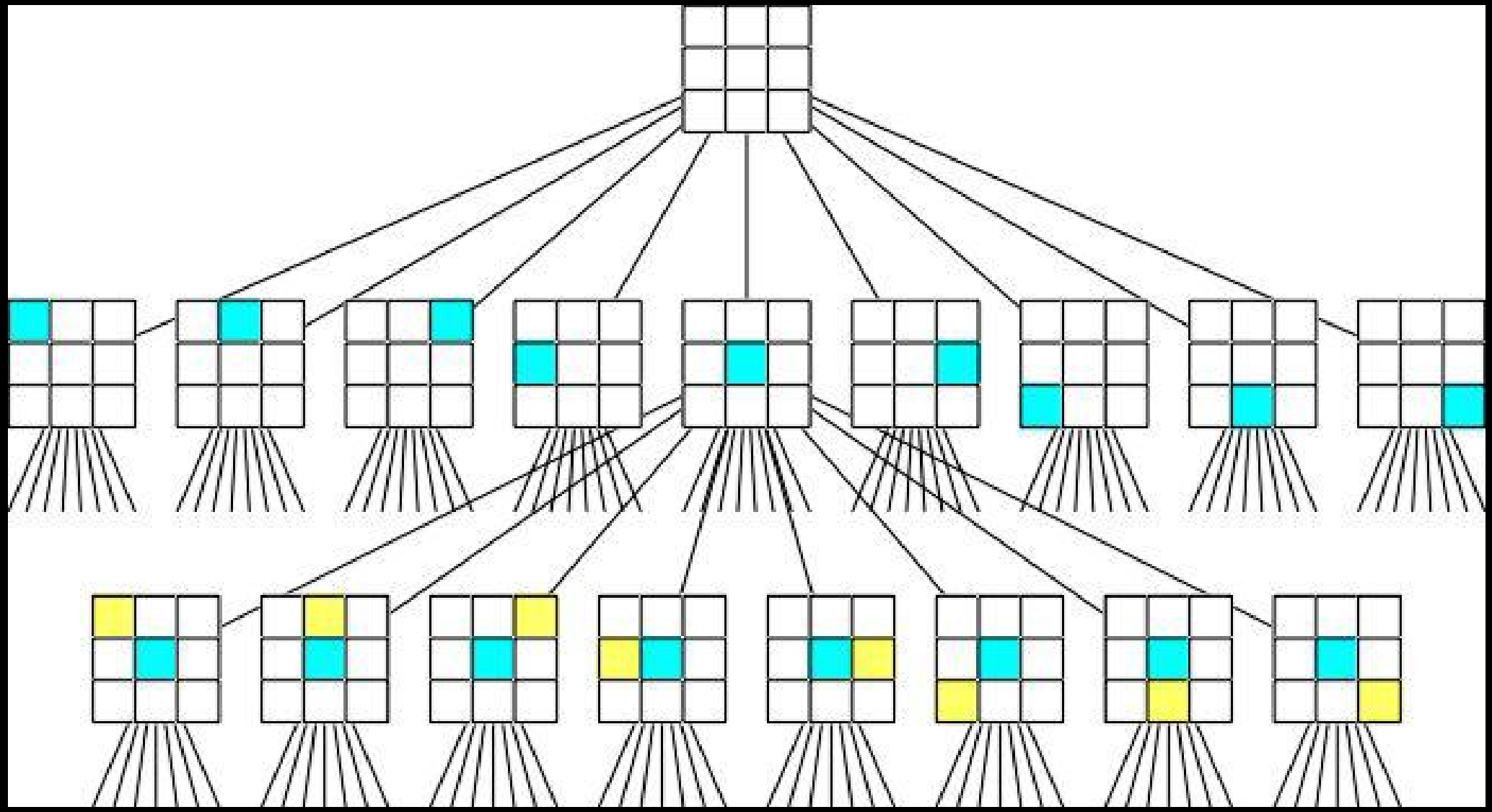
GAME SETUP

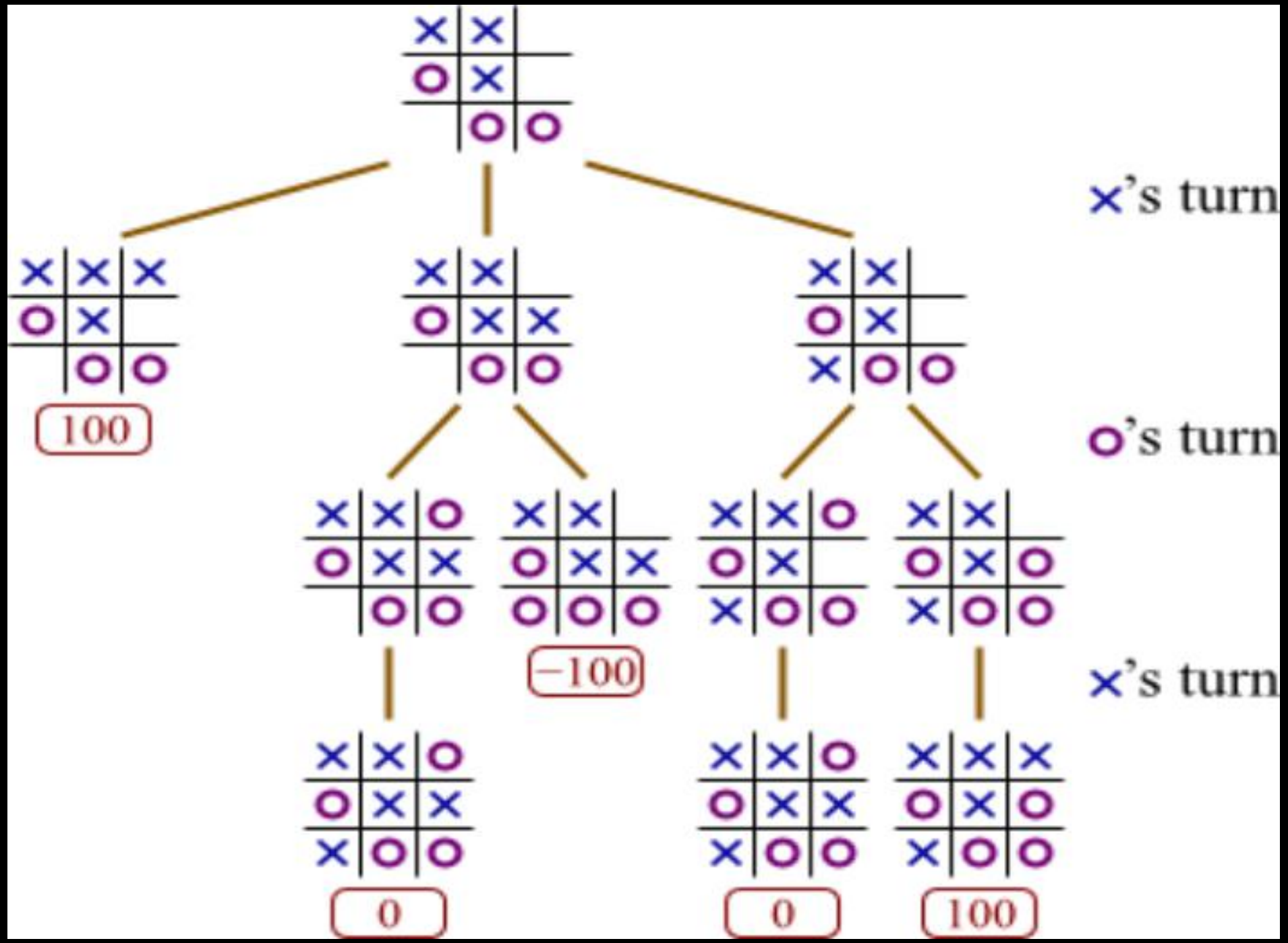
- Two Players : A and B.
- A moves first and they take turns until the game is over.
- Functionality of Game Search:
  - Initial state. Eg.- board configuration.
  - Successor Function: list of (move, state) pairs specifying legal moves.
  - Terminal Test: Is the game finished?
  - Utility Function: Gives numerical values to terminal states. Eg.- Win(+1), Lose(-1), Draw(0).
- A player uses search tree to determine next move.

PLAYING GAME BY SEARCHING

- General Scheme
  - Consider all legal moves, each of which will lead to some new state of the environment('board position').
  - Evaluate each possible resulting board position.
  - Pick the move which leads to the best board position/state.
  - Wait for your opponent's move, the repeat.
- Key Problems
  - Representing the state/board.
  - Representing legal next states.
  - Evaluating states.
  - Looking ahead.

TIC TAC TOE SEARCH TREE



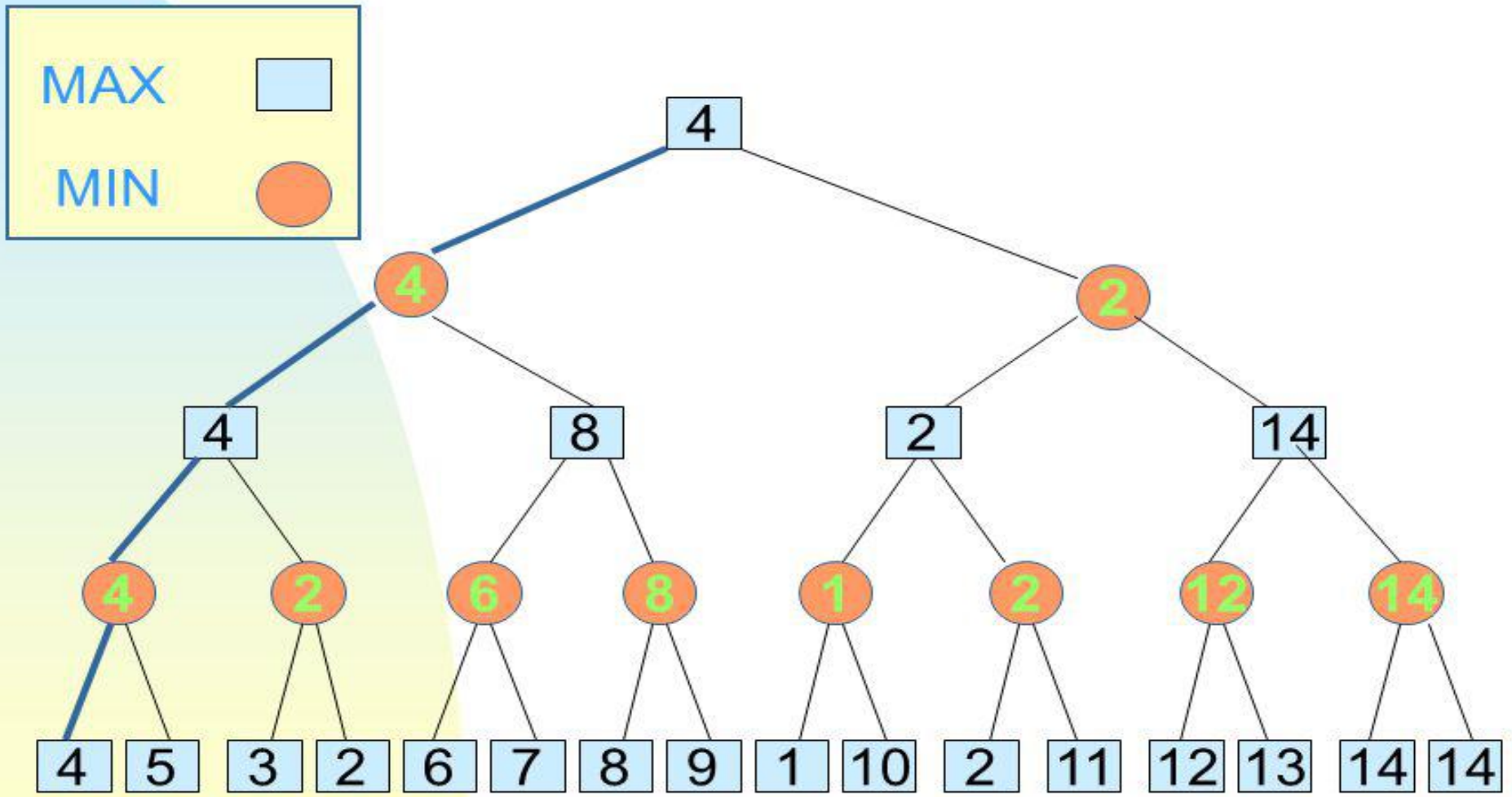

x's turn
o's turn
x's turn
100
−100
0
0
100

MINIMAX ALGORITHM

- Decision making in Multi-agent System.
- Perfect play for deterministic, perfect-information games.
- Logic: make the move for player MAX which has the most benefit assuming that MIN makes the best move for MIN in response.
- This is computed by a recursive process -
  - The backed-up value of each node in the tree is determined by the values of its children.
  - For a MAX node, the backed-up value is the maximum of the values of its children.
  - For a MIN node, the backed-up value is the minimum of the values of its children.

Minimax search

example Minimax Tree



PROPERTIES OF MINIMAX

- Complete? Yes (if the tree is finite)
- Optimal? Yes (against an optimal opponent)
- Time complexity? $O(b^m)$
- Space complexity? O(bm) (depth-first exploration)

Where b - average branching factor.

m - no.of moves it takes till the game ends.

- For a chess game, b = 35, m = 100 (approx.) for reasonable games, so exact solution is completely infeasible.
- So, the search tree has 35^{100} nodes. Even eliminating the duplicates, there are 10^{40} unique legal states. Assuming successors can be generated in 0.33 nanoseconds, it would take 10^{22} centuries to do a complete search.

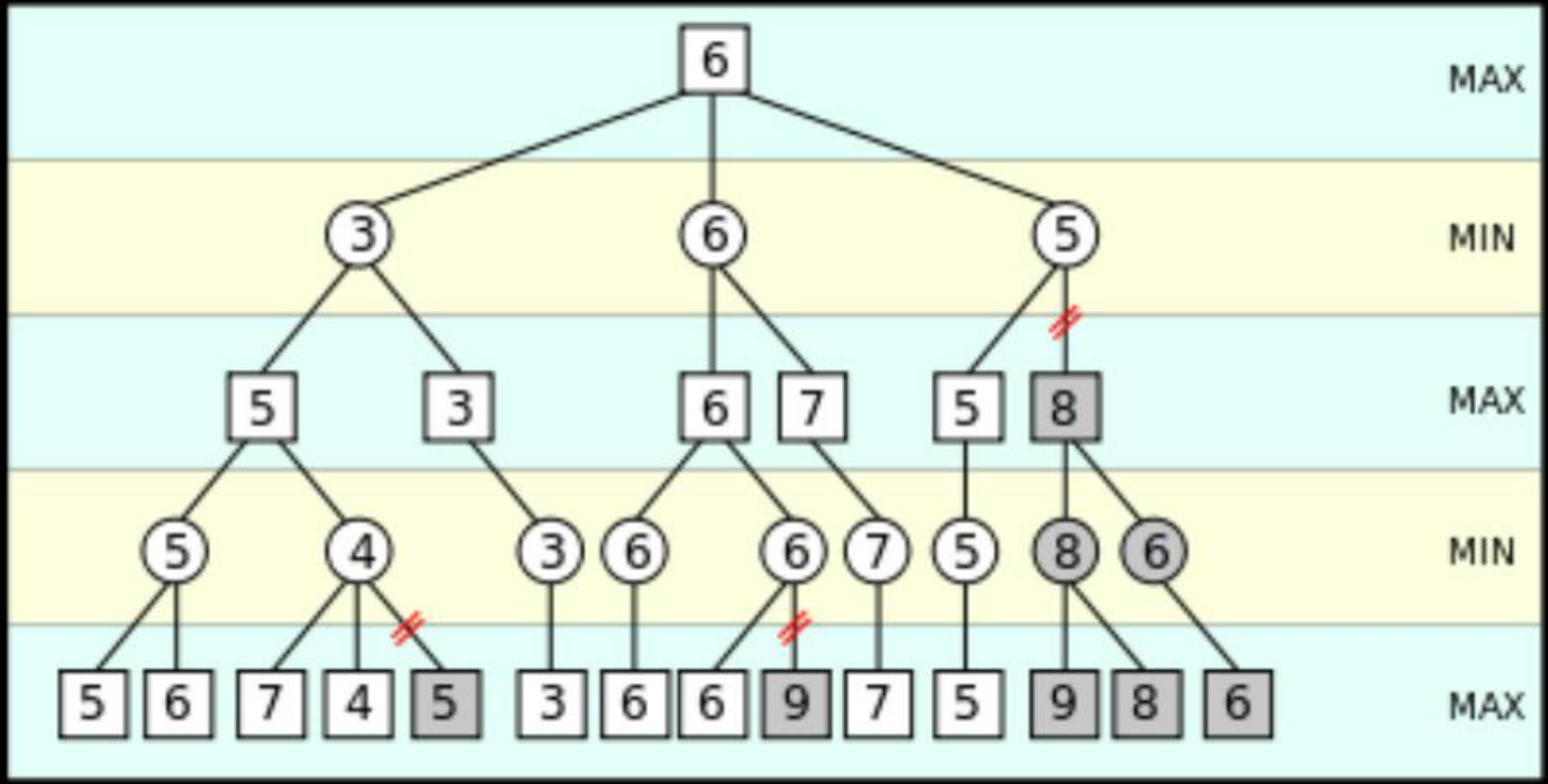
GUESS WHAT?

The Universe is only 10^{10} years old

ANY IMPROVEMENTS?

APLHA-BETA PRUNING

- Eliminating a branch without consideration is called pruning.
- A way to improve the performance of the Minimax Procedure.
- Basic Idea: "*If you have an idea which is surely bad, don't take the time to see how truly awful it is*".
- The final result is same as result obtained from Minimax algorithm without Alpha-Beta pruning.


MAX
MIN
MAX
MIN
MAX
6
3
6
5
5
3
6
7
5
8
5
4
3
6
6
7
5
8
6
5
6
7
4
5
3
6
6
9
7
5
9
8
6

PROPERTIES OF ALPHA-BETA PRUNING

- Pruning does not affect the final result.
- Good move ordering improves effectiveness of pruning.
- With "perfect ordering", time complexity = $O(b^{m/2})$
  - doubles depth of search
- Worst case: no pruning, same as Minimax $O(b^m)$

IBM DEEP BLUE

- Deep Blue was a chess-playing computer developed by IBM.
- It is known for being the first computer chess-playing system to win both a chess game and a chess match against a reigning world champion under regular time controls.
- Deep Blue defeated Garry Kasparov, the world champion at that time.
- For Deep Blue, alpha-beta pruning reduced the average branching factor from 35-40 to 6.

SO DOES ALPHA-BETA PRUNING SOLVES THE PROBLEM?

- Even on doubling the search depth, we are not even close to reducing the time complexity of the problem.
- As for today, it is not possible to compute a resonable chess game till end.

- WHATS NEXT?

"LIMITED DEPTH SEARCH" COMES TO THE RESCUE

- But if the depth is limited, how the nodes are given values (as only the leaves contains the values initially)?

SOLUTION

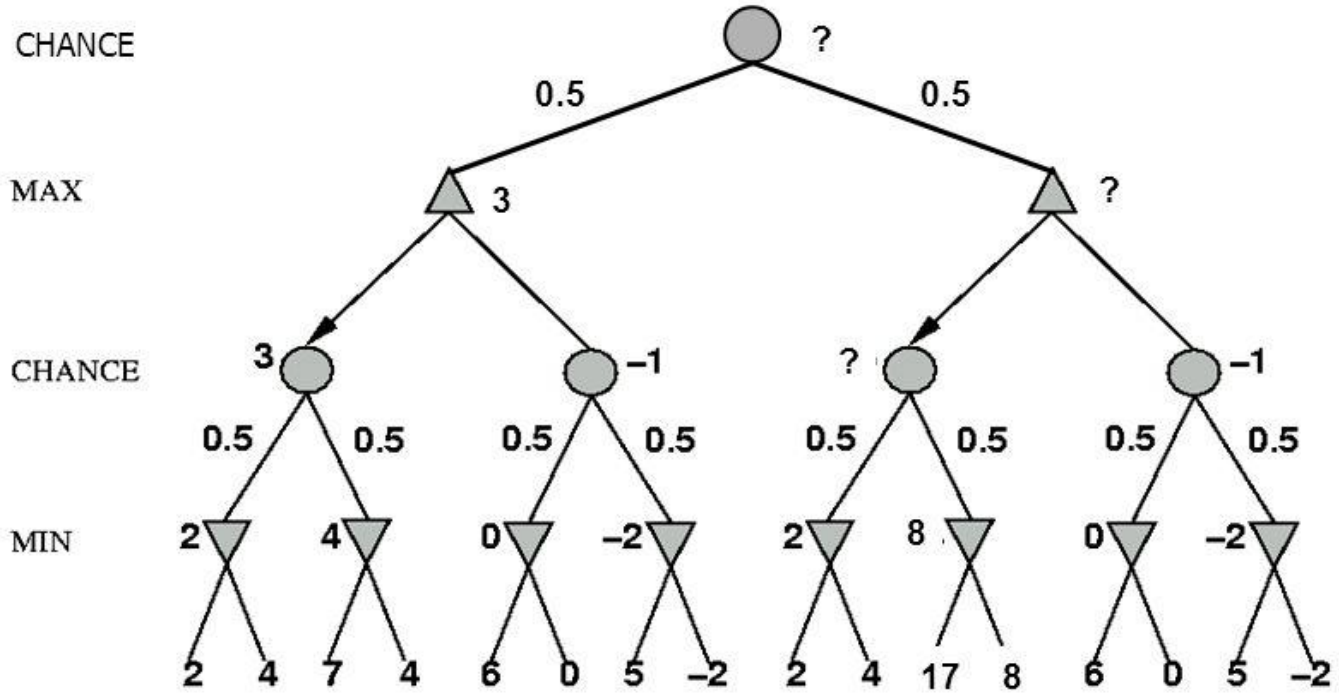
"EVALUATION FUNCTION"

MINIMAX FOR NON-DETERMINISTIC GAMES

- For MIN node, compute min of children.
- For chance node, compute weighted average of children.
- For max node, compute max of children.

Nondeterministic games: the element of chance

expectimax and **expectimin**, expected values over all possible outcomes



Problem Solving by Search

- An important aspect of intelligence is goal-based problem solving.
- The solution of many problems (Eg.- N-puzzle, 8 Queens) can be described by finding a sequence of actions that lead to a desirable goal. Each action changes the state and the aim is to find the sequence of actions and states that lead from the initial (start) state to a final (goal) state.
- A well-defined problem can be described by:
  - Initial state.
  - Operator or successor function - for any state x returns s(x), the set of states reachable from x with one action.
  - State space - all states reachable from initial by any sequence of actions.
  - Path - sequence through state space.
  - Path cost - function that assigns a cost to a path. Cost of a path is the sum of costs of individual actions along the path.
  - Goal test - test to determine if at goal state.

THE N-PUZZLE

AUTO-SOLVING N-PUZZLE

- DFS (DEPTH FIRST SEARCH)
- BFS (BREADTH FIRST SEARCH)
- A* SEARCH

HEURISTIC

- Heuristic, is any approach to problem solving, learning, or discovery that employs a practical method not guaranteed to be optimal or perfect, but sufficient for the immediate goals.

Heuristics for 8-puzzle I

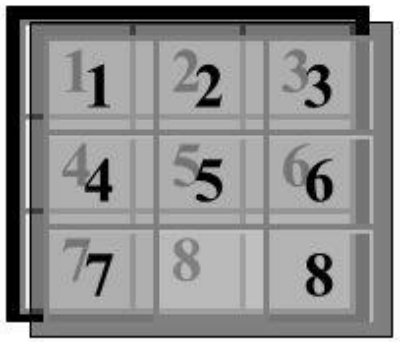
- The number of **misplaced tiles** (not including the blank)

Current State

1	2	3
4	5	6
7		8

Goal State

1	2	3
4	5	6
7	8	



N	N	N
N	N	N
N	Y	

In this case, only "**8**" is misplaced, so the heuristic function evaluates to 1.

In other words, the heuristic is *telling* us, that it *thinks* a solution might be available in just 1 more move.

Notation: $h(n)$ h(current state) = 1

Heuristics for 8-puzzle II

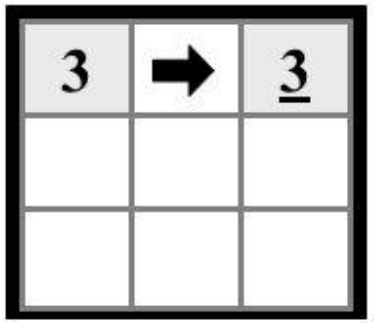
- The **Manhattan Distance** (not including the blank)

Current State

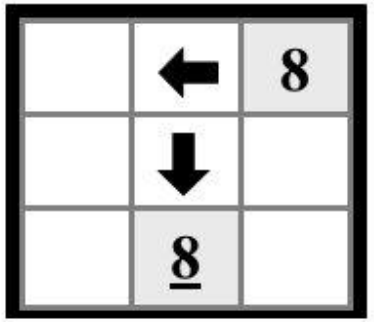
3	2	8
4	5	6
7	1	

Goal State

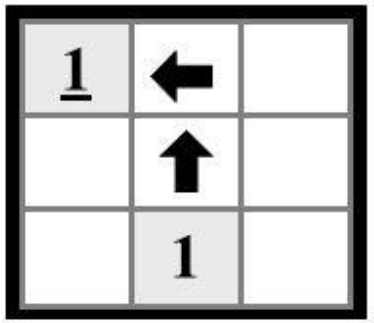
1	2	3
4	5	6
7	8	



2 spaces



3 spaces



3 spaces

Total 8

In this case, only the "**3**", "**8**" and "**1**" tiles are misplaced, by 2, 3, and 3 squares respectively, so the heuristic function evaluates to 8.

In other words, the heuristic is *telling* us, that it *thinks* a solution is available in just 8 more moves.

Notation: $h(n)$ $h(\text{current state}) = 8$

THANK YOU

By -

Gurpreet Singh
(TEAM.EXE)