CS362
Lab 8:Matchbox Educable Noughts and Crosses Engine

Presented by:

202051050 - BOORNA SAI PAVANI

202051092 - JUKTA BARUA

202051106 - KOTINI SWATEJ

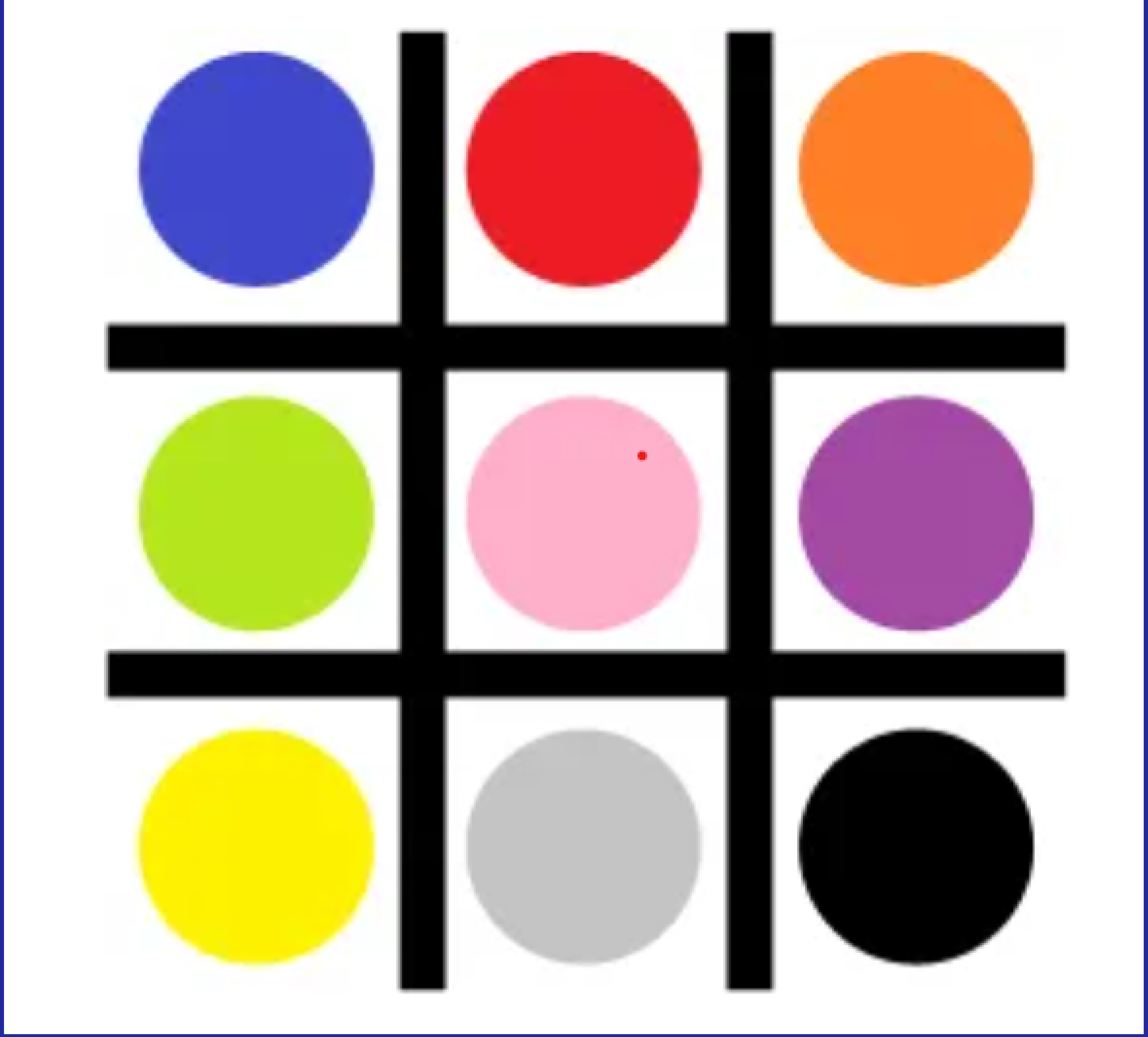
202051119 - MIDDE PAVANA SRI

Problem statement

Read the reference on MENACE by Michie and check for its implementations. Pick the one that you like the most and go through the code carefully. Highlight the parts that you feel are crucial. If possible, try to code the MENACE in any programming language of your liking.

- **MENACE** (Matchbox Educable Noughts and Crosses Engine) is an algorithm developed by Donald in the 1960s.
- Donald created an approach for training a set of matchboxes which contains beads in it to play knots and crosses or tic-tac-toe.
- Each matchbox represents a state of the game, and each bead represents a possible move that the player can make.

Tic-Tac-Toe Board

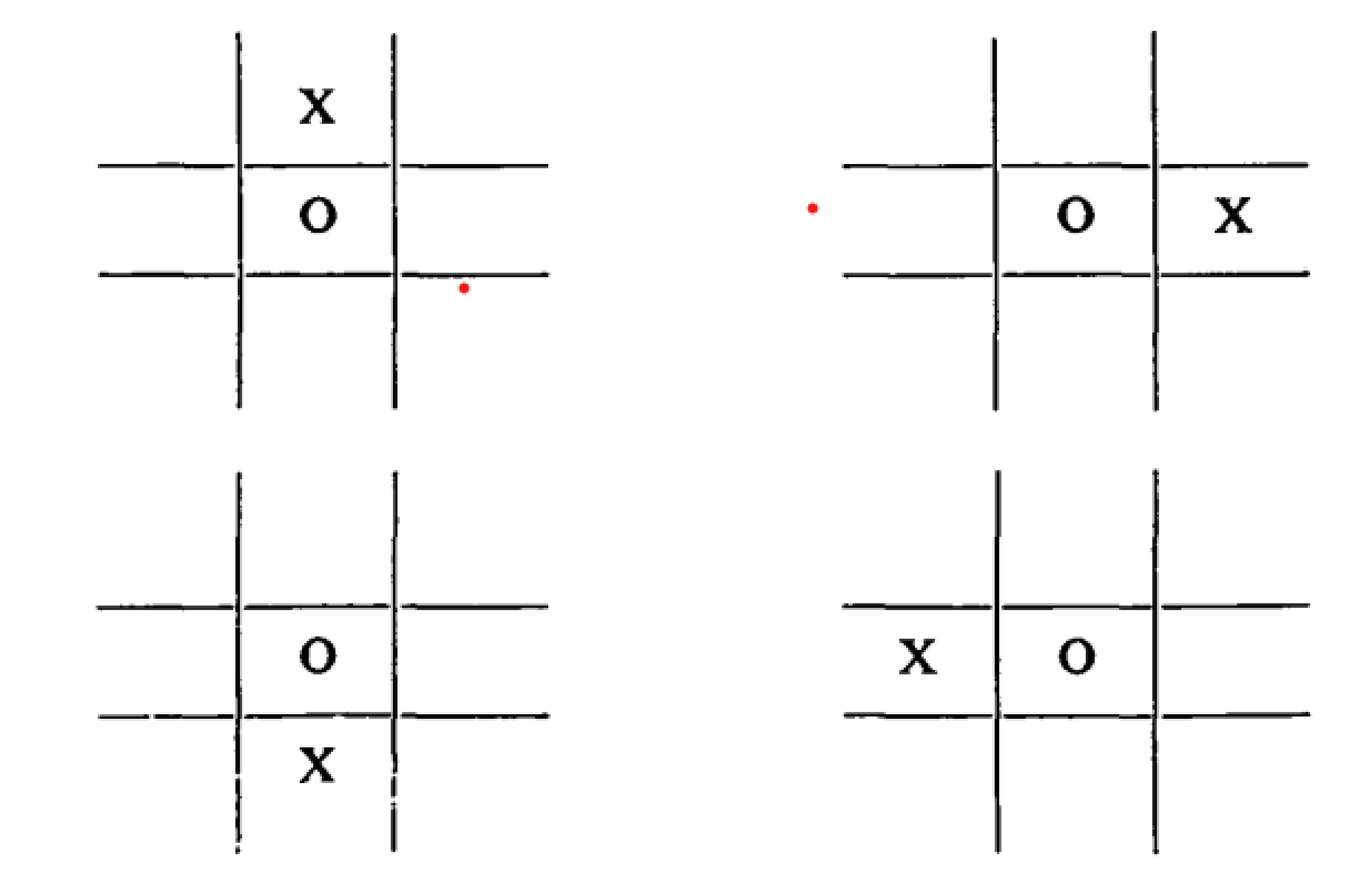


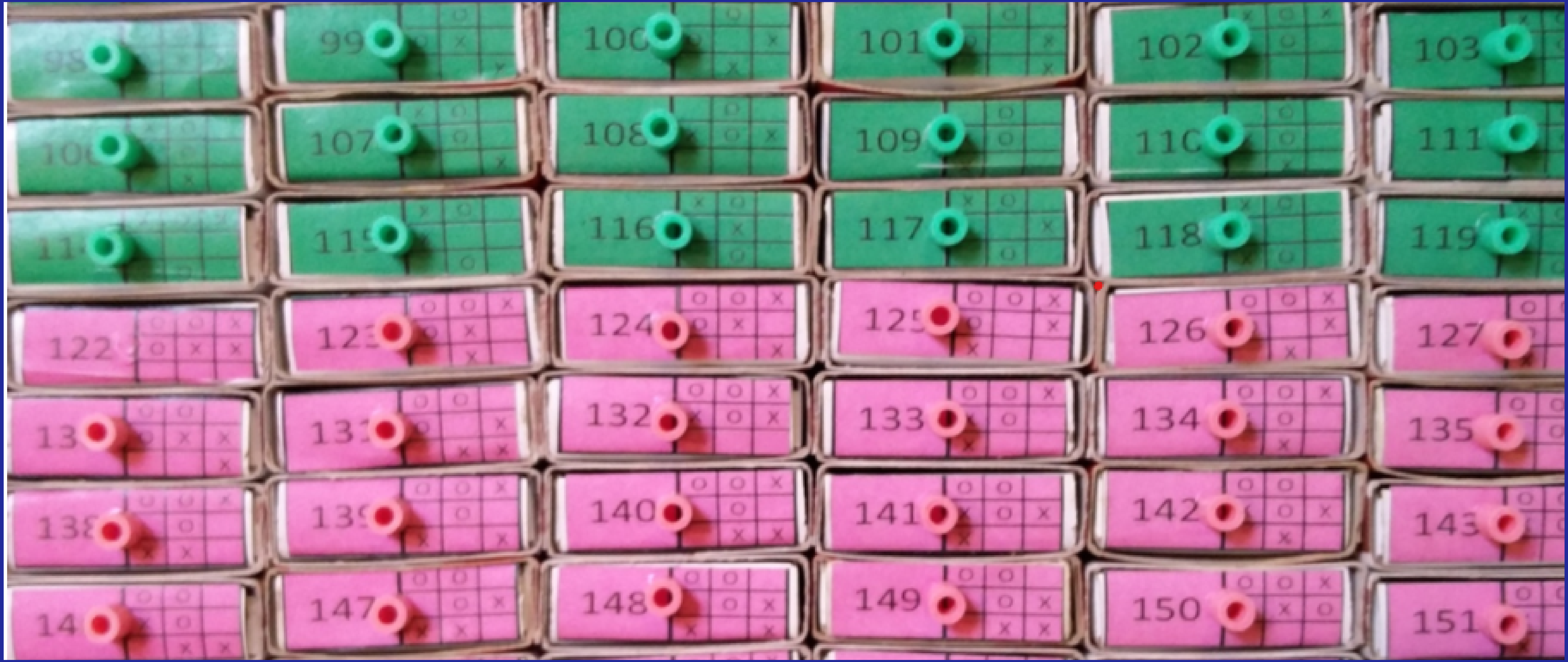
- He started out with 287 matchboxes all with an equal number of different beads in each box.

WHY 287 MATCHBOXES?

- After removing duplicate arrangements (ones that were simply rotations or mirror images of other configurations), there will be 287 possibilities.

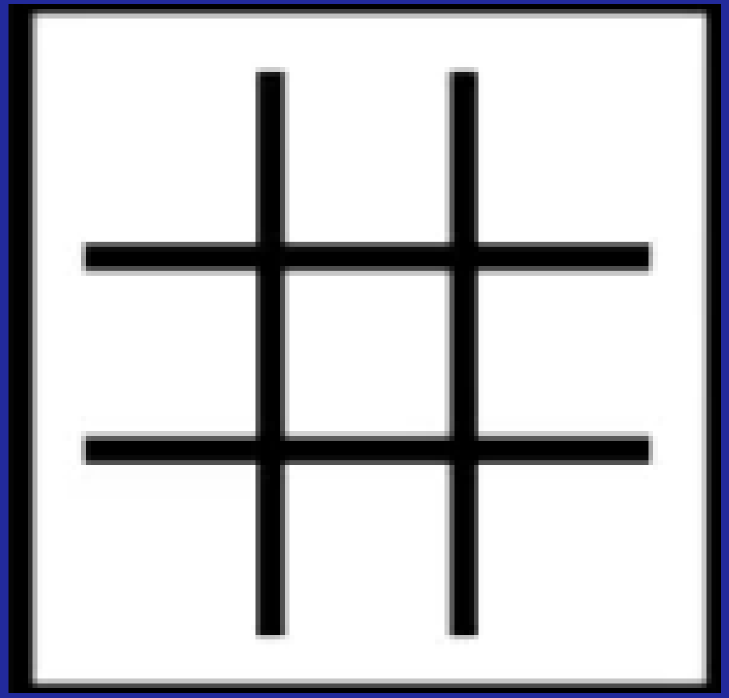
For Example:



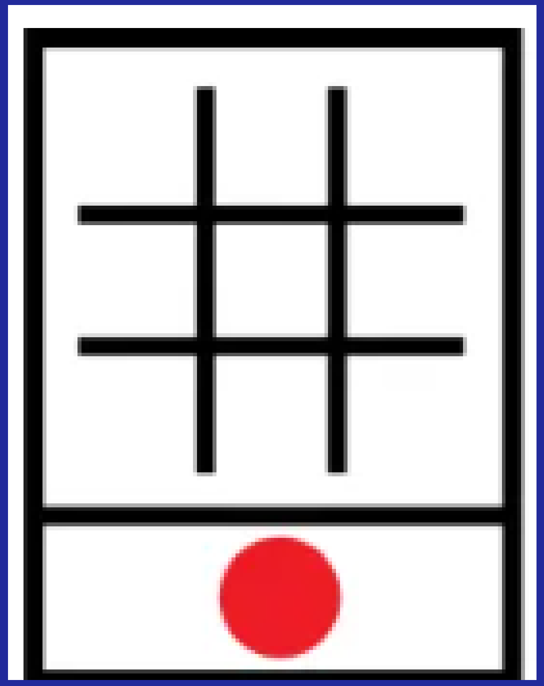


Matchboxes with labels on it

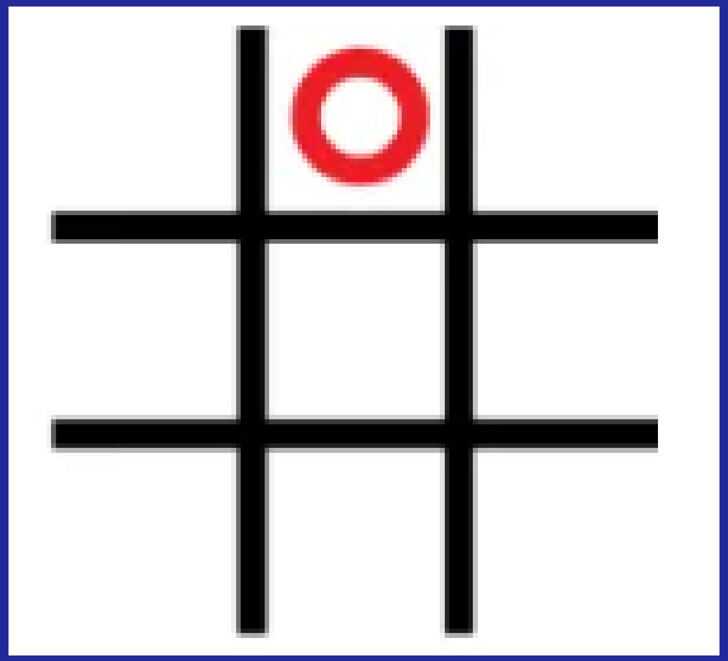
Where should we put O on the board?



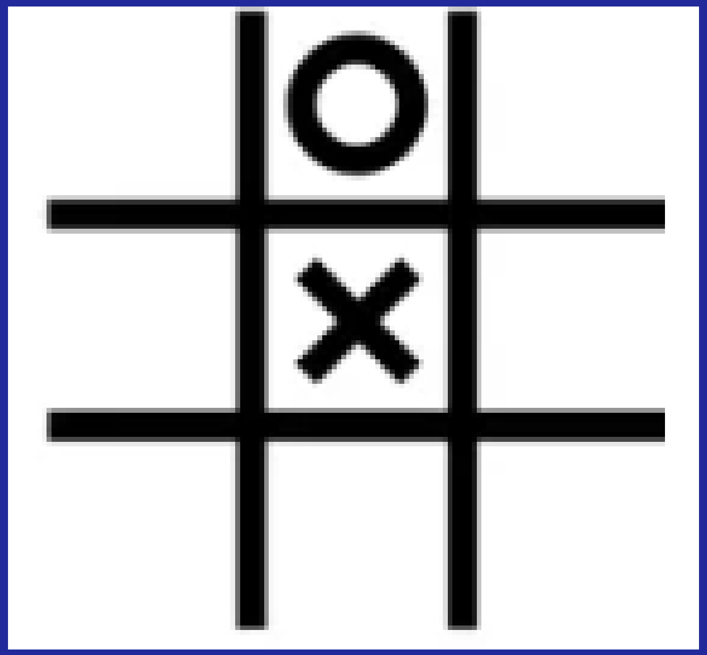
Board



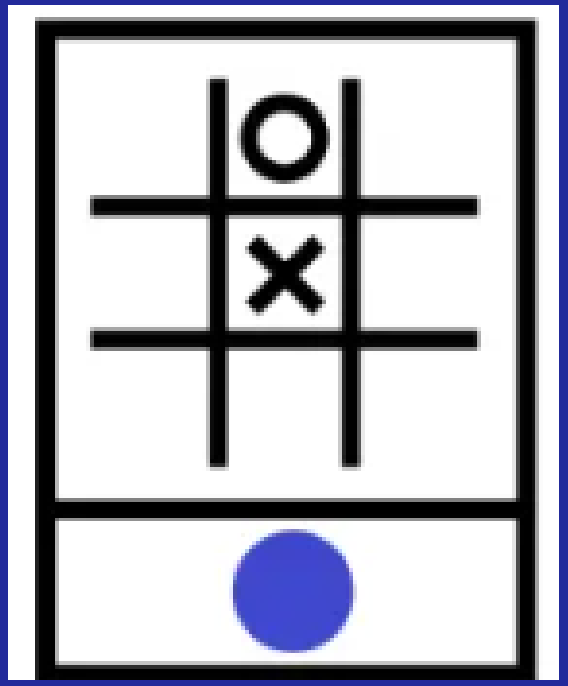
Matchbox



Board



Board

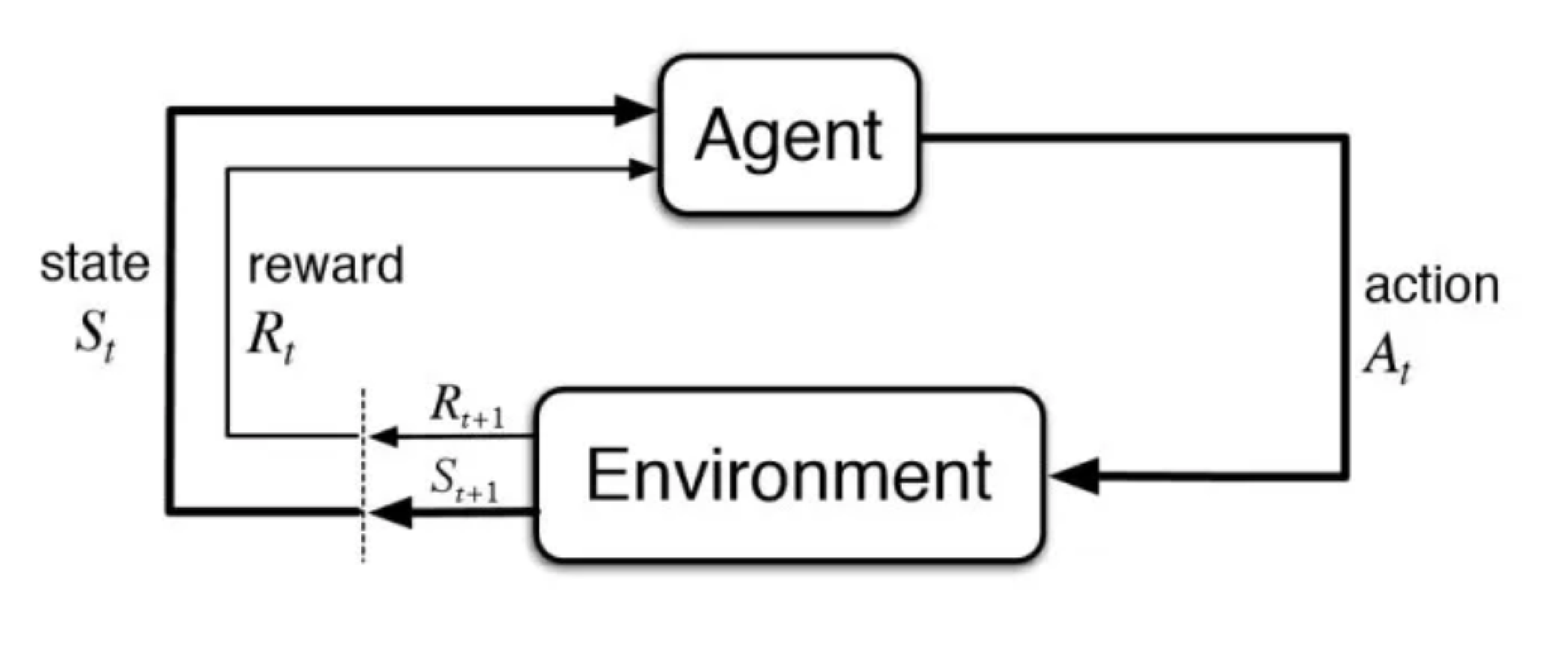


Matchbox

WHAT IS REINFORCEMENT LEARNING ?

Reinforcement learning is a type of machine learning in which an agent learns to make decisions by interacting with an environment in order to maximize a cumulative reward signal.

Diagramatical representation



The agent's goal is to learn a policy, which is a mapping from states to actions, that maximizes the expected cumulative reward over time.

Reinforcement learning is often implemented using a Markov decision process (MDP), which is a mathematical framework that formalizes the problem of decision-making in uncertain environments.

The reinforcement Learning Process can be simply summarized by the following process:

1. Agent plays a number of games
2. In every game, the agent chooses an Action from the action space by using Policy and Value Function
3. Action impacts the environment and the Reward and the new State is returned to the agent.
4. Agent keeps track of what reward it receives after choosing a specific action from a specific set.
5. After completing the game, the agent updates the estimated reward for each state and action by using the actual rewards values received while playing the game.
6. The whole process repeats again.

REINFORCEMENT LEARNING IN MENACE

Reinforcement learning plays a key role in the trial-and-error game of MENACE (Matchbox Educable Noughts and Crosses Engine). During the learning phase, the machine starts with a random set of matchboxes and beads. After each game, the machine updates the contents of the matchboxes based on the outcome of the game.

- If the machine wins the game, it adds a bead of the corresponding color to each matchbox it used during the game. If the machine loses the game, it removes a bead of the corresponding color from each matchbox it used during the game. If the game is a draw, nothing changes.
- This process of updating the matchboxes and beads is a form of reinforcement learning. By updating the contents of the matchboxes based on the outcome of the game, the machine is able to learn from its mistakes and improve its strategy over time.

Thank You