2048 GAME

Name: Shubam Limbachiya

Course: Artificial Intelligence-01 (CSCI-4740-01)

Banner Id.: 001382964

DEVELOPMENT SUMMARY

OBJECTIVE: Create an agent which can play the game- 2048 which is a 4x4 tiles game where the main objective is to reach at least the value of 2048 by combining same number of tiles which appears in a random manner whenever we move. The movements are UP,DOWN, LEFT, RIGHT and two tiles can only be combined when tiles with same values are adjacent to each other.
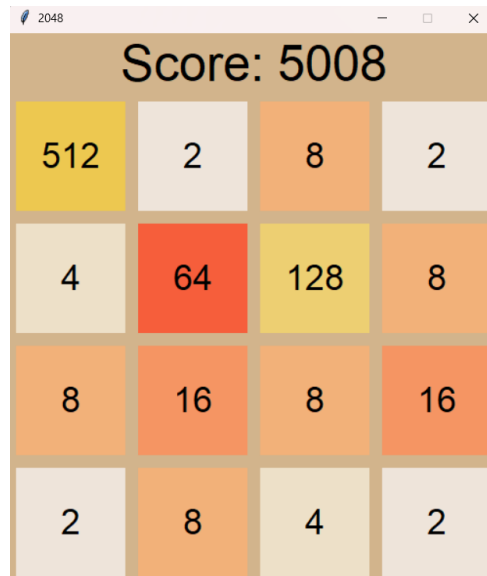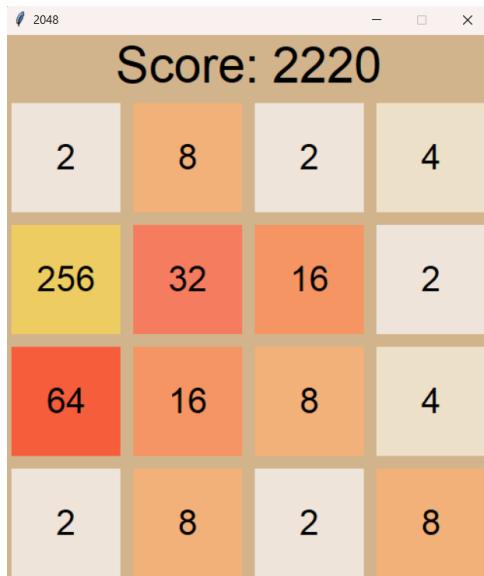
Version 1:
MinMax with Heuristics

Purpose: This code is a simple MinMax search with custom heuristics to evaluate the board states of the game.

The heuristics I used in this are:
- emptycells: Which gives more score with more empty tiles.
- maxTile: Check whether the maximum tile is in the corner of the board.
- cornerBonus: It rewards bonus if the maxTile is in the corner.
- Mergecells: Gives reward when board has some merging tiles together.
- Combining: Rewards when tiles combines with each other.

Testing:

Score of 10 Runs for 0.25 secs:

Sr. No	Score	Max Tile
1	7504	512
2	5008	512
3	3612	256
4	2840	256
5	2380	256
6	6160	512
7	2220	256
8	5144	512
9	5432	512
10	2980	256

Analysis:

- Using this code I observed that this algorithm was not able to keep the maxTile in one corner.
- The maximum tile reached was only 512.
- Score was not more than 10K in any runs.
- Movement of tiles was so random.
- Search dept was approximately 4-5 for each move.

Improvements to do:

- Upgrade Minmmax to expectimax
- Focus on Maxtile to keep it in the corner.
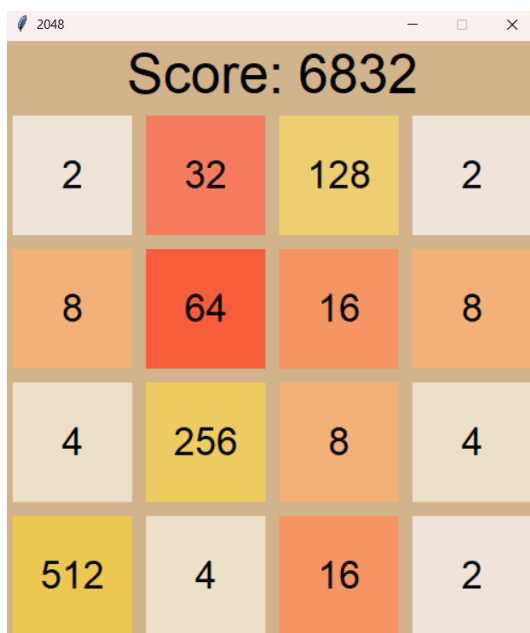- Keep more empty tiles in the board.

Version 2:
Expectimax With One heuristic

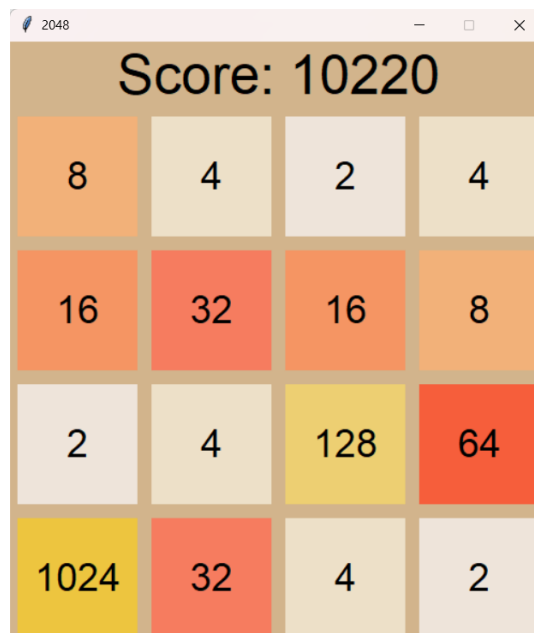**Purpose:** Simple Expectimax search algorithm with minimal heuristics.

Heuristic(empty) : This heuristics returns a rewards if the board has any empty cells present after each move.
And adding a weight to the heuristics to increase reward.

Testing:

Score of 10 Runs for 0.25 secs:

Sr. No	Score	Max Tile
1	2936	256
2	3028	256
3	6652	512
4	10220	1024
5	6832	512
6	6884	512
7	6232	512
8	5132	512
9	6112	512
10	7260	512

Analysis:
- Most of the time the Maxtile was not in the corner.
- Sometimes got no moves due to no adjacent same tiles.
- The max tile reached was 1024.
- Improved from MinMax but still tiles moves to be adjusted.
- Approx search depth was 5-6.

Improvements to do:
- Adding more heuristics with good rewards.
- Cornering the Maxtile.
- Some penalty for unwanted moves.
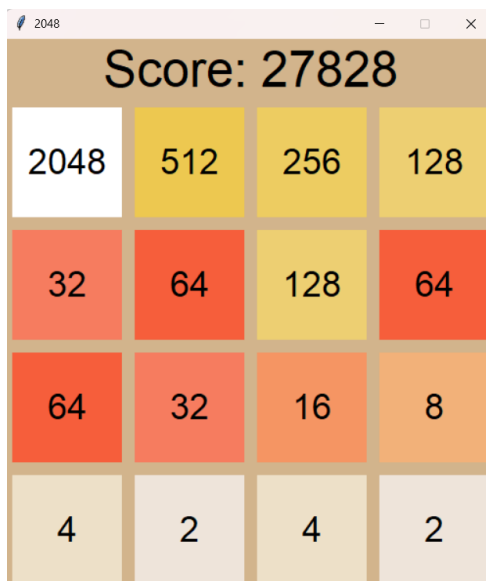- Move ordering for the tiles to combine more efficiently.

Version-3
Enhanced Heuristics in Expectimax

Purpose: Added more heuristics in the expectimax search algorithm to control the tiles moves so that the max tiles are in corner and does not block any other tiles.
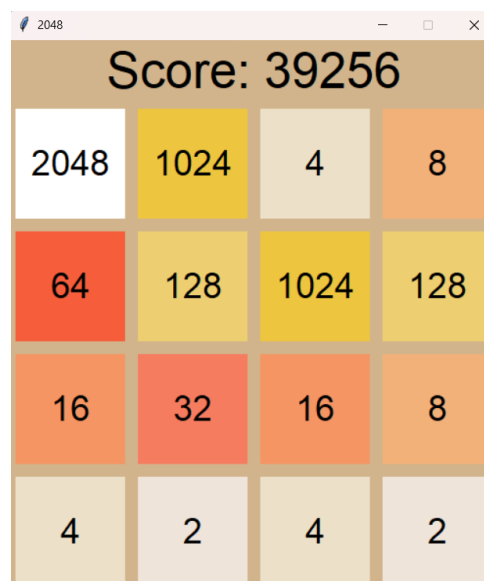
The heuristics added are:

- empty: Reward for more empty tiles in the board which is same as last version.
- monotonousrnc: This helps in increasing monotonicity in the game whose main function is to check how the values increase or decrease along the rows and columns.
- smoothrnc: Smoothness rewards for adjacent tiles with similar values.
- center_penalty: this penalizes of the high value tile is near to center.
- blockweightScore: This works like a snake pattern where each tile has its own unique weight and I have kept top left corner with the highest.
- avoidmove: Penalize if the high value tile is blocking lower valued tiles to merge.

- Move Ordering: Have a possible moves at a position related to max tile and prefers to keep it in the corner or on any edge to keep the board stable.

Testing:

Score of 10 Runs for 0.25 secs:

Sr. No	Score	Max Tile
1	19812	1024
2	12856	1024
3	18836	1024
4	23388	2048
5	16360	1024
6	18128	1024
7	23004	2048
8	26484	2048
9	39256	2048
10	27828	2048

Analysis:

- This code was performing very well as expected
- The max tile is always in the top left corner.
- Next highest value tile is always adjacent to max tile dur to blockweights.
- Minimum max tile was always 1024.
- Most of the time it reached 2048.
- The score was never below 10K and most of the time above 15K.
- The average search depth was 5-6.

CONCLUSION:

In this project, I got to learn about various strategies used in 2048 game and how can I implement them. Starting with MinMax Algorithm where I used simple heuristics to reach maximum tiles and more score but it was not much efficient in performance.

Then switching to Expectimax which is a probabilistic search algorithm performed well as this game has randomly appearing tiles. Adding a number of good heuristics like rewarding empty cells, maximizing the value and position of the largest tile by keeping it in a corner and penalizing the blocked tiles by high tiles. I even added some weight to the each board tiles which is known as Snake pattern which keeps the tiles in descending order.

And at last adding some Move Ordering based on the position of the Maxtile so that it can be kept at the corner and does not block other lower valued tiles.