

## 2048 Game


Chandana Banur Kalmarudappa

### Version 1- MinMax with Score Heuristic

#### Development Summary:

In this version I ran basic MinMax agent for the 2048 game using a score-based heuristic.

#### Testing Results:



A screenshot of a 2048 game window. The title bar says '2048'. The score is displayed as 'Score: 6040'. The game board is a 4x4 grid of tiles. The tiles are colored based on their value: 2 and 4 are light beige, 8 is orange, 16 is dark orange, 32 is yellow, 64 is red, 128 is light yellow, and 512 is bright yellow. The grid contains the following values:

4	2	4	8
64	128	64	4
2	16	512	16
8	128	4	2

- Final Score: 6040
- Time-limit: 0.1
- Search depth: 10

#### Analysis:

- The current MinMax agent avoids invalid moves and plays okay.
- It mostly moves right or down
- It has bad score mostly because of bad tile placement
- Big tiles don't end up in the corners
- Score is not consistent, sometimes it stops around 2000-4000

#### Next improvement:

- Try to fit the highest tile stays in corner.
- Make it work good by adding penalty if sudden jump.
- Make score consistent.

### Version 2- MinMax with an improved corner based heuristic

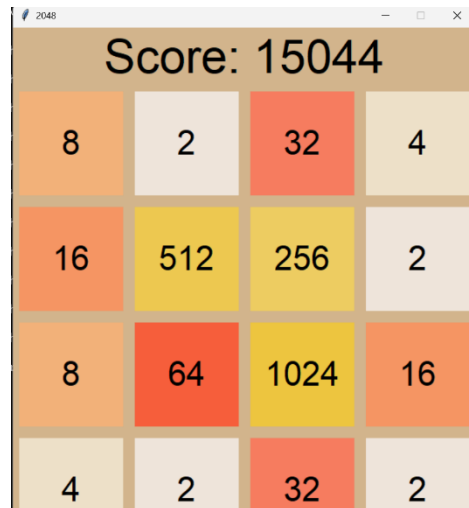
### Development Summary:

In this version the goal is to keep the maximum tile in a corner and reward empty spaces. So the Highest tile won't fluctuate and gives higher score

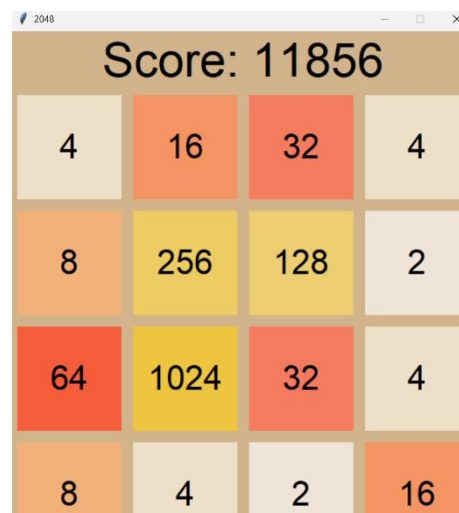
### Testing Results:



Score: 8960			
512	16	8	4
8	128	64	2
16	512	16	4
4	8	4	2



Score: 15044			
8	2	32	4
16	512	256	2
8	64	1024	16
4	2	32	2



Score: 11856			
4	16	32	4
8	256	128	2
64	1024	32	4
8	4	2	16

- Maximum Score: 15044
- Average Score: 8000
- Time-limit:0.1
- Max search depth: 108

### Analysis:

- My Agent avoids bad moves.
- Max tile most of the time stays in a corner.
- Board has more empty tiles.
- Score is more consistent compared to version 1.

#### Next improvement:

- I will try to add snake pattern and make the tiles evaluated with descending order.
- And will try to keep the highest tile in right corner without moving

#### Version 3- Expectimax with monotonic and smoothing heuristic

#### Development Summary:


In this version I used Expectimax instead of Minimax, I implemented:

- Highest tile in the corner
- I used snake pattern
- I implemented merge and smoothness

#### Test Results:



Score: 36036			
2048	1024	8	2
2	128	512	4
16	64	256	128
8	32	2	16



Score: 29812			
512	16	512	4
64	2048	256	2
8	16	64	16
4	8	4	2

- Maximum Score: 70000
- Average Score: 35000
- Time-limit:0.1

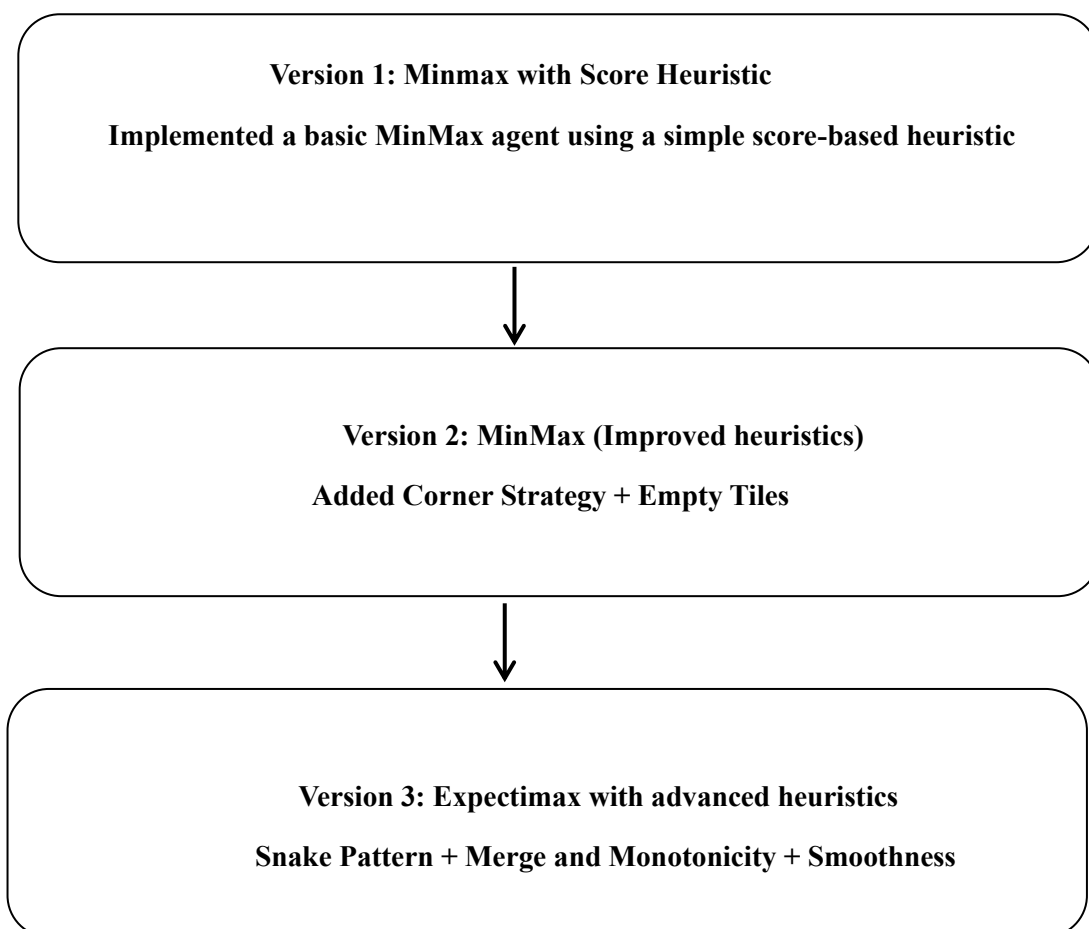
### Analysis:

- Expectimax made better decisions compared to minmax.
- For this version the max tile will always stays in corner which is one of the main heuristics in 2048
- Used weighted snake pattern which made the agent think even smarter
- The agent reached 2048+4096+ tiles
- This version was more stable with median score 35k

### Next Improvement:

- I will try to tune the weights to get more scores
- Try to merge both Minmax and Expectimax to implement more pruning.

### Flowchart:



**Conclusion:**

In all three versions my agent got better over time.

- Version 1 was a simple I used scores to make moves, but it did not think where the tiles were placed so it didn't do that well.
- Version 2 it got little smarter it tried to keep the highest tile in a corner. This helped the agent make better moves.
- Version 3 was even better I used Expectimax and added advanced heuristics where it kept the board smooth, highest tile cornered and using a snake pattern I was able to score more. This made version scored the highest and it was more stable too.