

Minesweeper Final AI Report

Team name insertGoodNameHere

Member #1 (name/id) Connor Zwick, cozwick, 86593203

Member #2 (name/id) Ryan Cruz, rncruz1, 44821443

I. Minimal AI

I.A. Briefly describe your Minimal AI algorithm. What did you do that was fun, clever, or creative?

Our Minimal AI was truly a barebones AI. We took advantage of the fact that there would only be 1 mine and used that along with the labels to determine which square was the mine with 100% accuracy. The algorithm essentially just summed the count of labels surrounding a square and used that to determine which square had to be the mine. Once we saw that this perfectly solved all Super Easy worlds we didn't make any additional improvements to our Minimal AI (because there was nothing to improve, it already met the requirements perfectly)

I.B. Describe your Minimal AI algorithm's performance:

Board Size	Sample Size	Score	Worlds Complete
5x5	100	100	100
8x8	100	0	0
16x16	100	0	0
16x30	100	0	0
Total Summary	400	100	100

II. Final AI

I.A. Briefly describe your Final AI algorithm, focusing mainly on the changes since Minimal AI:

Our Final AI was very different from our Minimal AI because now there was more than 1 mine on the board. We started by implementing a brute force algorithm but quickly found that this was too slow when the frontier grew too large (especially in Expert worlds). We then created a combinatorial depth first search algorithm inspired by the brute force combinatorial algorithm which performed significantly better (it was much much faster) but it too struggled when the search space grew large. The algorithm would essentially create a path along the frontier squares, using the effective labels of those squares to determine whether or not a bomb could be placed in a particular square. Sometimes this approach would allow us to find squares that could never be bombs, other times we had to use the different valid combinations to determine which square was least probable to be a bomb. This algorithm was only run as a last resort due

to its time complexity, simpler algorithms, such as our effective label algorithm, were run first to see if they could find any valid moves.

I.A. Describe your Final AI algorithm's performance:

Board Size	Sample Size	Score	Worlds Complete
5x5	100	100	100
8x8	100	88	88
16x16	100	162	81
16x30	100	78	26
Total Summary	400	428	295

II. In about 1/4 page of text or less, provide suggestions for improving this project (this section does NOT count as part of your two-page total limit.)

One of the improvements we attempted to implement was segmentation of the frontier search space (which would've allowed our depth first search algorithm to handle larger search spaces), however we found it was too difficult to implement correctly without introducing many more bugs. A better heuristic for quickly guessing which square is least likely to be a bomb would also help since this was the cause of many of our worlds failing (our DFS algorithm would take too long to finish so we had to resort to quickly guessing based off of effective label counts).