Minesweeper Solver

***Analysis and implementation of algorithm***

Rushabh Kadam (180001045)

Jemin Vagadia (180001023)

We have decided to work on minesweeper solving algorithms. Also we will analyze time complexities of these algorithms.

Introduction:

Minesweeper is a single-player puzzle game available on several operating systems and GUIs. At the start of a game, the player receives an n × m rectangular grid of covered squares or cells. Each turn, the player may probe or uncover a square revealing either a mine or an integer. This integer represents the number of mines adjacent to that particular square. As such, the number on a cell ranges from 0 to 8 since a cell cannot have more than eight neighbours. The game ends when the player probes a cell containing a mine. The objective of the game is to uncover every square that does not contain a mine.

Algorithms:

The trivially straightforward algorithm is actually good enough to solve the beginner and intermediate versions of the game. Using this algorithm for tough puzzles guesses must be made which may lead to incorrect solution. Tough problems can be solved by tank solver algorithm. The idea for the Tank algorithm is to **enumerate all possible** configurations of mines for a position, and see what’s in common between these configurations. We always try to apply the simple algorithm first, and only if that gets us stuck, then we bring in the Tank algorithm.

To implement the Tank algorithm, we first make a list of **border tiles**: all the tiles we aren’t sure about but have some partial information. Using backtracking in this algorithm will reduce number of all possible configurations. we can make one important optimization. The optimization is **segregating** the border tiles into several disjoint regions.



In this case, the green region has 10 tiles, the pink has 7. Taken together, we need to search through 2^{17} combinations. With segregation, we only have 2^{10} + 2^7: about a 100x speedup.

There might occur a situation in which we are forced to guess so we will have to study the probability for such spot.

Goals:

1. Solving minesweeper puzzle of each difficulty efficiently.
2. To analyze time complexity of algorithm.
3. To analyze the ways to improve the algorithms (if possible).

Project Outline:

To solve minesweeper efficiently and study different ways to do it in different conditions. Studying probability for selecting first square. Also comparing different approaches to solve minesweeper for different puzzles.

References:

<https://www.aaai.org/ocs/index.php/WS/AAAIW17/paper/download/15091/14775>

<https://luckytoilet.wordpress.com/2012/12/23/2125/>

<https://hal.inria.fr/hal-00750577v2/document>

<https://dash.harvard.edu/bitstream/handle/1/14398552/BECERRA-SENIORTHESIS-2015.pdf?sequence=1>

<http://worldcomp-proceedings.com/proc/p2016/PDP7565.pdf>

<https://pdfs.semanticscholar.org/2083/3f71d74ff26ffd18979796cf1cbc8b3d92e6.pdf>