(Kind of) Real World Problem description:

The 8-puzzle is made up of eight sliding squares and one empty square. The squares are slid to be numbered in order, from 0 to 8. This puzzle is difficult for humans as well as computers. However, it can be solved with breadth-first search1.

The basic algorithm is to note keep track of the actions that can be taken. A blank space can always move up, down, left and right. If there is a wall in the direction that it is moving, then it will remain in the same state. Each node produces four new nodes (if you include the possibility that it will loop back on itself). Each of these nodes produces a search tree of its own, iterating through the same options, up, down, left, and right.

Hand-traced example:

|  |  |  |
| --- | --- | --- |
| 3 | 6 | 8 |
| 7 |  | 2 |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
| 3 | 6 | 8 |
| 7 | 4 | 2 |
| 5 |  | 1 |

|  |  |  |
| --- | --- | --- |
| 3 | 6 | 8 |
| 7 | 2 |  |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
| 3 |  | 8 |
| 7 | 6 | 2 |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
| 3 | 6 | 8 |
|  | 7 | 2 |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
| 3 | 6 | 8 |
| 7 |  | 2 |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
| 3 | 8 |  |
| 7 | 6 | 2 |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
|  | 3 | 8 |
| 7 | 6 | 2 |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
| 3 | 6 | 8 |
| 5 | 7 | 2 |
|  | 4 | 1 |

|  |  |  |
| --- | --- | --- |
| 3 | 6 | 8 |
| 7 |  | 2 |
| 5 | 4 | 1 |

|  |  |  |
| --- | --- | --- |
|  | 6 | 8 |
| 3 | 7 | 2 |
| 5 | 4 | 1 |

References:

[1]I. Parberry, *A Memory-Efficient Method for Fast Computation of Short 15-Puzzle Solutions*. Denton: IEEE, 2014, p. 2.