Activity 12: Artificial Intelligence

Today's activity will focus on a classic AI problem: how to make decisions for winning a simple tile game. We will also begin to develop an algorithm for playing the game automatically.

Model 1   The 8-Puzzle

Imagine a 3x3 grid of tiles numbered 1 to 8 that can slide horizontally or vertically. The gray square below represents the empty space into which adjacent tiles can move.

1	2	3
4	5	6
7	8	

A (goal)

1	2	
4	5	3
7	8	6

B (easy)

7	4	8
1	5	3
	6	2

C (hard)

1		3
5	2	6
4	7	8

D (easy)

Explore the puzzle a bit by visiting mypuzzle.org/sliding, and make sure you understand how the tile movement works. Don't spend too much time actually trying to solve the puzzles yet.

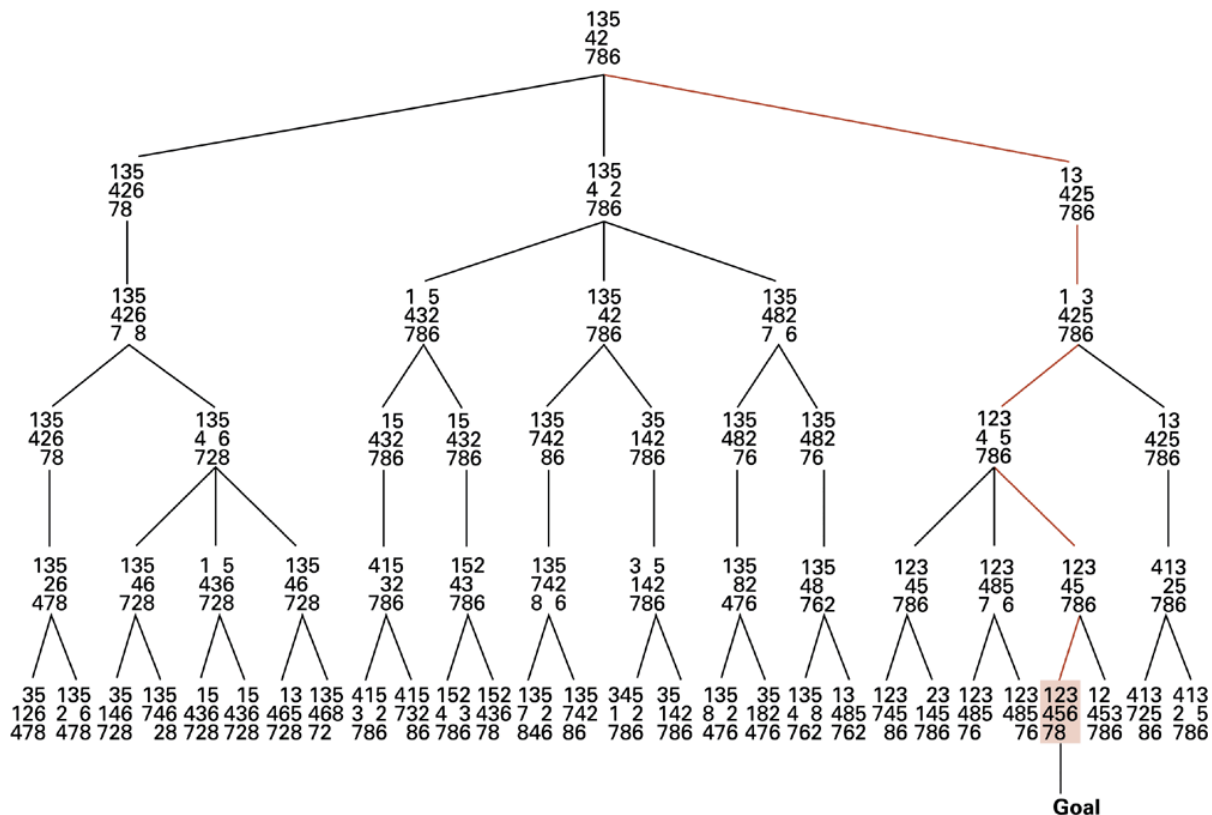## Questions (20 min)   Start time: _____

1. How many different tiles can be moved in examples B and C? How many in example D? Describe a general way of classifying the number of moves from any given layout.

2. In the center of the space below, copy the grid from example B. Then draw the possible next grids, one on each side of B, and draw a line between B and each neighbor. Imagine you can transform the puzzle from one grid to another by traveling along the lines. Continue this process for all possible moves two steps from the original example B grid.

3. Imagine repeating #2 for example C. How big might the drawing get?

4. What makes example C harder to solve than B and D?

5. Assuming you could remove all the tiles and then place them back on the grid, how many different puzzle layouts are there?

6. Now consider the game of Tic-Tac-Toe (or see playtictactoe.org if you're unfamiliar with it). Draw a diagram similar to #2 for the first two moves, assuming X moves first, starting from an empty 3x3 grid. Don't worry about completely drawing all possible boards, but do sketch enough to illustrate you know how it works.

7. Name three ways the Tic-Tac-Toe diagram differs from what you did in the 8-Puzzle.

8. Would it be possible for a computer to store the solution for all possible 8-tile and tic-tac-toe games? Why or why not?

Model 2   Search Trees

```
                                  135
                                   42
                                  786
```

```
        135                       135                        13
        426                       4 2                        425
         78                       786                        786
```

```
        135              1 5      135      135                1 3
        426              432       42      482                425
        7 8              786      786      7 6                786
```

```
    135      135      15   15    135   35    135   135     123        13
    426      4 6      432  432   742  142    482   482     4 5        425
     78      728      786  786    86  786     76    76     786        786
```

```
135   135   1 5  135   415  152  135  3 5  135  135  123  123  123  413
 26    46   436   46    32   43  742  142   82   48   45  485   45   25
478   728   728  728   786  786  8 6  786  476  762  786  7 6  786  786
```

```
 35 135  35 135   15   15   13 135  415 415 152 152 135 135 345  35  135  35 135  13  123  23 123 123 123 12  413 413
126 2 6 146 746  436  436  465 468  3 2 732 4 3 436 7 2 742 1 2 142 8 2 182 4 8 485 745 145 485 485 456 453 725 2 5
478 478 728  28  728  728  728  72  786  86 786  78 846  86 786 786 476 476 762 762  86 786  76  76  78     786  86 786
```

Goal

Questions (15 min)   Start time: _____

9. In your own words, describe what the above diagram represents.

10. Given that there are always at least two possible moves, why do some board configurations seem to have only one branch?

11.   Consider two methods for making the move choices at each step: *Random* (just choose randomly among all legal moves), and *Lowest* (always choose the lowest numbered available tile to move). Could either of these methods get to a solution?

12. If you were given several puzzle layouts and had to choose the easiest to solve, how would you choose?

13. Describe an algorithm that can take any 8-Puzzle grid and compute some measure of the difficulty. Use your thinking from the previous question: think about how to get to a number that is bigger, the harder the puzzle might be to solve.

14. Apply your algorithm to the second row of the model. You should get a number for each of the three board configurations.

15. Describe how you could use your algorithm from #13 to make a better next-step choice method than the methods in #11.