Connor Bettermann

CS441

2/10/19

Programming HW #1

Below are the results from the trial implemented in first programming assignment. The goal of this assignment was to implement three different heuristics for solving the “8 puzzle.” Of the three heuristics, each was implemented both in a best-first algorithm as well as an A\* algorithm.

The results were very conclusive, in that all trials were completed by the A\* search algorithms, while none were completed by the best-first algorithms. I checked to make sure the code was correct for the best-first algorithms, but soon realized that the only puzzles they could solve were ones of trivial nature, where the maximum number of steps needed to solve the puzzle was 2. I determined this, because the algorithm itself would always choose the best available option when deciding which path to take and in every trial, the solution path required a move that would “take it further from the goal” in the logic of the best-first algorithm. All three heuristics performed equally well in the A\* star search algorithm, providing identical paths as well as average steps to complete. The first two heuristics were given out of the book, being misplaced tiles and Manhattan distance. The third heuristic that I implemented was the Euclidian distance the tiles were from there goal state. This turned out to be an equally effective heuristic as the other two provided to me. Finally, I feel as though the addition of the A\* algorithm, while crucial in this case, was not utilized to its full potential in this puzzle, because each weight between a parent and child node was the same regardless of the placement on the tree. If there was a way to implement the “distance traveled” variable in a more varying and descriptive manner, there might be some variation between the efficiency of the heuristics as well.

Best-First search:

**Misplaced Tiles:**

**PATH 1**

Initial State: (4 1 2 5 8 3 7 b 6)

Maximum recursion reached

: DNF

**PATH 2**

Initial State: (2 b 3 1 7 5 8 4 6)

Maximum recursion reached

: DNF

**PATH 3**

Initial State: (1 3 6 5 2 b 4 7 8)

Maximum recursion reached

: DNF

**PATH 4**

Initial State: (1 b 2 4 6 3 7 5 8)

Maximum recursion reached

: DNF

**PATH 5**

Initial State: (1 2 3 4 8 5 b 7 6)

Maximum recursion reached

: DNF

A\*search:

**Misplaced Tiles:**

**PATH 1**

Initial State: (4 1 2 5 8 3 7 b 6)

(4 1 2 5 8 3 7 b 6) -> (4 1 2 5 b 3 7 8 6)

-> (4 1 2 b 5 3 7 8 6) -> (b 1 2 4 5 3 7 8 6)

-> (1 b 2 4 5 3 7 8 6) -> (1 2 b 4 5 3 7 8 6)

-> (1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 7 steps

**PATH 2**

Initial State: (2 b 3 1 7 5 8 4 6)

(2 b 3 1 7 5 8 4 6) -> (b 2 3 1 7 5 8 4 6)

-> (1 2 3 b 7 5 8 4 6) -> (1 2 3 7 b 5 8 4 6)

-> (1 2 3 7 4 5 8 b 6) -> (1 2 3 7 4 5 b 8 6)

-> (1 2 3 b 4 5 7 8 6) -> (1 2 3 4 b 5 7 8 6)

-> (1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 9 steps

**PATH 3**

Initial State: (1 3 6 5 2 b 4 7 8)

(1 3 b 5 2 6 4 7 8) -> (1 b 3 5 2 6 4 7 8)

-> (1 2 3 5 b 6 4 7 8) -> (1 2 3 b 5 6 4 7 8)

-> (1 2 3 4 5 6 b 7 8) -> (1 2 3 4 5 6 7 b 8)

-> (1 2 3 4 5 6 7 8 b)

: 7 steps

**PATH 4**

Initial State: (1 b 2 4 6 3 7 5 8)

(1 b 2 4 6 3 7 5 8) -> (1 2 b 4 6 3 7 5 8)

-> (1 2 3 4 6 b 7 5 8) -> (1 2 3 4 b 6 7 5 8)

-> (1 2 3 4 5 6 7 b 8) -> (1 2 3 4 5 6 7 8 b)

: 5 steps

**PATH 5**

Initial State: (1 2 3 4 8 5 b 7 6)

(1 2 3 4 8 5 7 b 6) -> (1 2 3 4 b 5 7 8 6)

(1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 4 steps

**AVERAGE NUMBER OF STEPS: 6.4 steps**

Best-First Search:

**Manhattan Distance:**

**PATH 1**

Initial State: (4 1 2 5 8 3 7 b 6)

Maximum recursion reached

: DNF

**PATH 2**

Initial State: (2 b 3 1 7 5 8 4 6)

Maximum recursion reached

: DNF

**PATH 3**

Initial State: (1 3 6 5 2 b 4 7 8)

Maximum recursion reached

: DNF

**PATH 4**

Initial State: (1 b 2 4 6 3 7 5 8)

Maximum recursion reached

: DNF

**PATH 5**

Initial State: (1 2 3 4 8 5 b 7 6)

Maximum recursion reached

: DNF

A\* search:

**Manhattan Distance:**

**PATH 1**

Initial State: (4 1 2 5 8 3 7 b 6)

(4 1 2 5 8 3 7 b 6) -> (4 1 2 5 b 3 7 8 6)

-> (4 1 2 b 5 3 7 8 6) -> (b 1 2 4 5 3 7 8 6)

-> (1 b 2 4 5 3 7 8 6) -> (1 2 b 4 5 3 7 8 6)

-> (1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 7 steps

**PATH 2**

Initial State: (2 b 3 1 7 5 8 4 6)

(2 b 3 1 7 5 8 4 6) -> (b 2 3 1 7 5 8 4 6)

-> (1 2 3 b 7 5 8 4 6) -> (1 2 3 7 b 5 8 4 6)

-> (1 2 3 7 4 5 8 b 6) -> (1 2 3 7 4 5 b 8 6)

-> (1 2 3 b 4 5 7 8 6) -> (1 2 3 4 b 5 7 8 6)

-> (1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 9 steps

**PATH 3**

Initial State: (1 3 6 5 b 2 4 7 8)

(1 3 6 5 b 2 4 7 8) -> (1 3 b 5 2 6 4 7 8)

-> (1 b 3 5 2 6 4 7 8) -> (1 2 3 5 b 6 4 7 8)

-> (1 2 3 b 5 6 4 7 8) -> (1 2 3 4 5 6 b 7 8)

-> (1 2 3 4 5 6 7 b 8) -> (1 2 3 4 5 6 7 8 b)

: 7 steps

**PATH 4**

Initial State: (1 b 2 4 6 3 7 5 8)

(1 b 2 4 6 3 7 5 8) -> (1 2 b 4 6 3 7 5 8)

-> (1 2 3 4 6 b 7 5 8) -> (1 2 3 4 b 6 7 5 8)

-> (1 2 3 4 5 6 7 b 8) -> (1 2 3 4 5 6 7 8 b)

: 5 steps

**PATH 5**

Initial State: (1 2 3 4 8 5 b 7 6)

(1 2 3 4 8 5 7 b 6) -> (1 2 3 4 b 5 7 8 6)

(1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 4 steps

**AVERAGE NUMBER OR STEPS: 6.4**

Best-First Search:

**Euclidian Distance:**

**PATH 1**

Initial State: (4 1 2 5 8 3 7 b 6)

Maximum recursion reached

: DNF

**PATH 2**

Initial State: (2 b 3 1 7 5 8 4 6)

Maximum recursion reached

: DNF

**PATH 3**

Initial State: (1 3 6 5 2 b 4 7 8)

Maximum recursion reached

: DNF

**PATH 4**

Initial State: (1 b 2 4 6 3 7 5 8)

Maximum recursion reached

: DNF

**PATH 5**

Initial State: (1 2 3 4 8 5 b 7 6)

Maximum recursion reached

: DNF

A\* search:

**Euclidian Distance:**

**PATH 1**

Initial State: (4 1 2 5 8 3 7 b 6)

(4 1 2 5 8 3 7 b 6) -> (4 1 2 5 b 3 7 8 6)

-> (4 1 2 b 5 3 7 8 6) -> (b 1 2 4 5 3 7 8 6)

-> (1 b 2 4 5 3 7 8 6) -> (1 2 b 4 5 3 7 8 6)

-> (1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 7 steps

**PATH 2**

Initial State: (2 b 3 1 7 5 8 4 6)

(2 b 3 1 7 5 8 4 6) -> (b 2 3 1 7 5 8 4 6)

-> (1 2 3 b 7 5 8 4 6) -> (1 2 3 7 b 5 8 4 6)

-> (1 2 3 7 4 5 8 b 6) -> (1 2 3 7 4 5 b 8 6)

-> (1 2 3 b 4 5 7 8 6) -> (1 2 3 4 b 5 7 8 6)

-> (1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 9 steps

**PATH 3**

Initial State: (1 3 6 5 2 b 4 7 8)

(1 3 b 5 2 6 4 7 8) -> (1 b 3 5 2 6 4 7 8)

-> (1 2 3 5 b 6 4 7 8) -> (1 2 3 b 5 6 4 7 8)

-> (1 2 3 4 5 6 b 7 8) -> (1 2 3 4 5 6 7 b 8)

-> (1 2 3 4 5 6 7 8 b)

: 7 steps

**PATH 4**

Initial State: (1 b 2 4 6 3 7 5 8)

(1 b 2 4 6 3 7 5 8) -> (1 2 b 4 6 3 7 5 8)

-> (1 2 3 4 6 b 7 5 8) -> (1 2 3 4 b 6 7 5 8)

-> (1 2 3 4 5 6 7 b 8) -> (1 2 3 4 5 6 7 8 b)

: 5 steps

**PATH 5**

Initial State: (1 2 3 4 8 5 b 7 6)

(1 2 3 4 8 5 7 b 6) -> (1 2 3 4 b 5 7 8 6)

(1 2 3 4 5 b 7 8 6) -> (1 2 3 4 5 6 7 8 b)

: 4 steps

**AVERAGE NUMBER OF STEPS: 6.4 steps**