1. State space: All the moves done by player or all the possible moves 9 can go.

Moves: the player can only move one of the cells in this case the number 9 and it can only go up, left down or right.

Goal test: The goal test checks whether the current state is a valid magic square, i.e., whether the sum of the numbers in each row, column, and diagonal is the same.

Goal state: each column, row, and diagonal add up to 15.

We can use a search algorithm(greedy best in this case but A\* also works) to find all possible moves from the initial state to the goal state.

|  |  |  |
| --- | --- | --- |
| 6 | 9 | 8 |
| 7 | 1 | 3 |
| 2 | 5 | 4 |

|  |  |  |
| --- | --- | --- |
| 9 | 6 | 8 |
| 7 | 1 | 3 |
| 2 | 5 | 4 |

|  |  |  |
| --- | --- | --- |
| 6 | 1 | 8 |
| 7 | 9 | 3 |
| 2 | 5 | 4 |

|  |  |  |
| --- | --- | --- |
| 6 | 8 | 9 |
| 7 | 1 | 3 |
| 2 | 5 | 4 |

|  |  |  |
| --- | --- | --- |
| 7 | 6 | 8 |
| 9 | 1 | 3 |
| 2 | 5 | 4 |

|  |  |  |
| --- | --- | --- |
| 6 | 1 | 8 |
| 7 | 5 | 3 |
| 2 | 9 | 4 |

|  |  |  |
| --- | --- | --- |
| 6 | 1 | 8 |
| 9 | 7 | 3 |
| 2 | 5 | 4 |

**Goal state**

1. *h(n)* = (8 + 4 + 4) + (0 + 0 + 0) + 4 + 4 = 24

*h(n)* = (8 + 4 + 4) + (3 + 3 + 0) + 1 + 4 = 27

*h(n)* = (0 + 4 + 4) + (0 + 0 + 0) + 4 + 4 = 16

*h(n)* = (8 + 4 + 4) + (0 + 1 + 0) + 4 + 3 = 24

*h(n)* = (6 + 2 + 4) + (3 + 3 + 0) + 3 + 4 = 25

*h(n)* = (0 + 4 + 4) + (2 + 2 + 0) + 2 + 2 = 16

*h(n)* = (0 + 0 + 0) + (0 + 0 + 0) + 0 + 0 = 0 **Goal state**