# CSCI1410 Search

## Heuristic

**A1:** Informally describe your heuristic:

```
heur = 0
for row in range(3):
    tuples = state[row]
    for col in range(3):
        value = tuples[col]
        if value == 1 :
            heur += abs(0-row) + abs(0-col)
        if value == 2 :
            heur += abs(0-row) + abs(1-col)
        if value == 3 :
            heur += abs(0-row) + abs(2-col)
        if value == 4 :
            heur += abs(1-row) + abs(0-col)
        if value == 5 :
            heur += abs(1-row) + abs(1-col)
        if value == 6 :
            heur += abs(1-row) + abs(2-col)
        if value == 7 :
            heur += abs(2-row) + abs(0-col)
        if value == 8 :
            heur += abs(2-row) + abs(1-col)
        if value == 9 :
            heur += abs(2-row) + abs(2-col)
return (heur / 2)
```

In my implementation of the heuristic, I used the Manhattan distance to calculate the heuristic. Since a state is a tuple of tuples, I used two for loops to find the value and the corresponding row and column. Since the goal state is ((1,2,3),(4,5,6),(7,8,9)), I used the row and column of the value and calculated where in the board the value should go. This way, I found the absolute value of the location of where the value should go subtracted by the location of where the value is. I added the displacement of both the row and the column and added to the total integer heuristic and divided it by two. This heuristic will give an equal value as the optimal heuristic.

**A2:** Prove that your heuristic is admissible:
My heuristic calculates the number of steps each value in a row and column to get to the row and column of the goal state. Since my heuristic adds all of the steps of each value to get to the goal state and then divides it by two.

so if:

$$f(n) = g(n) + h(n)$$

Where, $h^*(n)$ is the optimal cost to get to $n$

if $h(n) \leq h^*(n)$

the heuristic is admissible.

If a state is at the goal state, my heuristic will produce 0. Say a state is:

$$State = \begin{bmatrix} 1 & 2 & 3 \\ 7 & 4 & 5 \\ 6 & 9 & 8 \end{bmatrix}$$

The calculation of my heuristic will be 4. The number of switches required to get to the goal state, in this example is 4 at the minimum. Since the value of switches required is equal to my heuristic, it is plausible to say that my heuristic is admissible.

**A3:** Explain why you chose this heuristic over other possible heuristics.

I chose this heuristic because for a heuristic to work most optimally on a given algorithm, it should produce the lowest score RELATIVE to other states. For example, take the StateOne we looked at in the previous question.

$$StateOne = \begin{bmatrix} 1 & 2 & 3 \\ 7 & 4 & 5 \\ 6 & 9 & 8 \end{bmatrix}$$

Then, take a look another state StateTwo:

$$StateTwo = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 7 \\ 7 & 9 & 8 \end{bmatrix}$$

My heuristic score of StateOne is 4. My heuristic score StateTwo is 1. The number of steps required, or optimal heuristic would be 4 for StateOne and 1 for StateTwo. Taking this account, my heuristic calculates exactly the distance between the goal state and the start state then divides it by two to comply with the number of swaps. Since my heuristic is equal to the optimal heuristic, it is admissible.