

**PROBLEM 3**

--- USE 2 LATE DAYS ---

Bob Wagner

The heuristic I chose to implement calculates cost by determining the number of tiles out of their respective goal row and/or column. I rationalized this choice by analyzing the way the successor function was set up. We can use the information that successors are Cardinal-direction shifted in conjunction with the lumped (returned value) cost in order to infer relationships about tile position.

For example, if we know that the heuristic cost is a small multiple of 5 or 3, we can see that our matrix could only contain a specific number of incorrect row/column arrangements. This could allow for expansion on our A\* algorithm to skip expanding nodes and instead jump to possible, more efficient solutions. In this way, it provides some interesting benefits.

This heuristic *is* admissible, and would most likely work best for small matrices/state sizes.