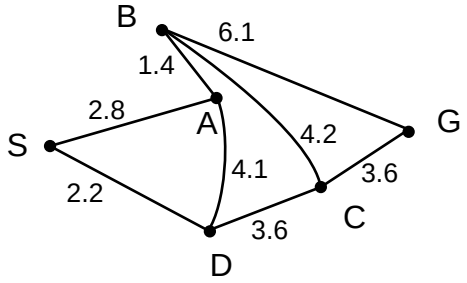
CSCI 4150 Introduction to Artificial Intelligence, Fall 2001
Assignment 3 (120 points): out Monday September 17, due Thursday September 27

Notes on this assignment

- Questions 1 and 5 are to be turned in on paper. Please note that there is a new policy for automatic extension and late policy deadlines for written work starting with this assignment. The automatic extension will only be until 5pm on the day it is due. The first tier late deadline for written work will be 5pm on Friday. Deadlines for the electronic portions remain the same.
- The other questions involve Scheme programming. Your solutions for these questions are to be turned in electronically.
- This is most likely the last assignment that you will be able to do using Dr. Scheme.

A* search

1. (30 points; written) For this question, you will apply the A* algorithm to find a path in the graph below from the start node (S) to the goal node (G). The cost of each edge is labeled in the graph, and the accompanying table provides the heuristic function you are to use for this problem



node	h(n)
A	4.5
B	5.1
C	3.6
D	7.2

Show which nodes are on the OPEN and CLOSED list after each step of the search; show the values of $f(n)$, $g(n)$, and $h(n)$ for each node on these lists.

Sliding block puzzles

For this problem, you will write Scheme procedures to solve sliding block puzzles using the A* algorithm. Perhaps the best known example of this type of puzzle is the 8 puzzle which consists of 8 tiles in a 3 by 3 grid. Each move in this puzzle consists of sliding into the empty grid cell (i.e. the space) one of the adjacent tiles. For example:

1	2	3
4	5	6
7	8	

$\rightarrow$

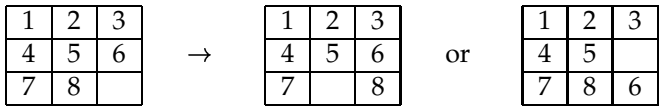
1	2	3
4	5	6
7		8

For this problem, we will only consider sliding block puzzles where each tile is the size of one grid cell.

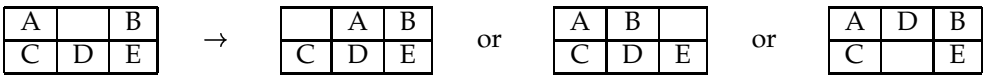
We will represent this kind of puzzle so that any geometry can be used, not just the 3 by 3 grid of the eight puzzle. The puzzle state will consist of a list of rows; each row will be represented by a list of tiles; and each tile will be represented by either a number or a symbol. Assume each tile has a unique number or symbol. The space will be represented by the symbol `space`.

An implementation of the A* algorithm and other support code will be available on the Assignment 3 information (web) page for you to run and test your code.

2. (30 points) Write a procedure `(sbp-children s)` that generates the children of a given state of the puzzle `s`. For example:




```
(sbp-children '((1 2 3) (4 5 6) (7 8 space)))
  ==> (((1 2 3) (4 5 space) (7 8 6)) ((1 2 3) (4 5 6) (7 space 8)))
```




```
(sbp-children '((A space B) (C D E)))
  ==> (((space A B) (C D E)) ((A B space) (C D E)) ((A D B) (C space E)))
```

   The order of the children is not important.

3. (30 points) Write a procedure `(sbp-mahnattan state goal)` which returns the sum (over all tiles) of the manhattan distance from the current position of the tile to its position in the goal state. Note that the `space` is not a tile. Test your procedure using the provided A* implementation.

4. (15 points) Write a procedure `(ep-heuristic state goal)` which implements a heuristic (other than manhattan distance) for the eight puzzle. The Assignment 3 information page will have a number of known heuristics that you may implement, but I encourage you to try designing your own. Your score on this part may be based on the performance of your heuristic. (Test your procedure using the provided A* implementation.)

5. (10 points; written) Explain your heuristic. Is it original, a variation on a known heuristic, or a known heuristic? Is this heuristic admissible? Is it monotonic?

⋆6. (5 points) Challenge problem — see the Assignment 3 information page.