Problem 2. [15] Hill Climbing

a) [3] If you have *n* locations, how many neighboring states does the *neighbors*(*s*) function produce?
This is a combination problem.
If there are n locations in the current state, we pick two locations out of n and swap them.
So we have neighbors: $C_2^n = \frac{n(n-1)}{2}$ neighbors

b) [3] What is the total size of the search space, i.e., how many possible states are there in total? Assume again that there are *n* locations.
This is a permutation problem.
Assuming the start city is arbitrary and a tour does not return to the start city, then the size of the search space is $P_n^n = n!$

c) [9] Imagine that a student wants to hand out fliers about an upcoming programming contest. The student wants to visit the Memorial Union (*M*), Wisconsin Institute of Discovery (*W*), Computer Sciences Building (*S*), and Engineering Hall (*E*) to deliver the fliers. The goal is to find a tour as short as possible. The distance matrix between these locations is given as follows:

	M	*W*	*E*	*S*
M	0	0.6	0.9	0.7
W	0.6	0	0.3	0.2
E	0.9	0.3	0	0.4
S	0.7	0.2	0.4	0

The student starts applying hill-climbing algorithm from the initial state: <*M-E-S-W*>. What is the *next* state reached by hill-climbing, or explain why there is no neighboring state. When will we know if we should stop or continue the search? Will we know if the state is a global optimal solution when we stop? Briefly explain your answers.
For the initial state <M-E-S-W> which has cost of 0.9+0.4+0.2 = 1.5. We have 6 successors:
$< E - M - S - W >: 0.9 + 0.7 + 0.2 = 1.8$
$< S - E - M - W >: 0.4 + 0.9 + 0.6 = 1.9$
$< W - E - S - M >: 0.3 + 0.4 + 0.7 = 1.4$
$< M - S - E - W >: 0.7 + 0.4 + 0.3 = 1.4$
$< M - W - S - E >: 0.6 + 0.2 + 0.4 = 1.2$
$< M - E - W - S >: 0.9 + 0.3 + 0.2 = 1.4$
So we choose $< M - W - S - E >$ .... [3]
The algorithm stops if no neighbor has lower distance. .... [3]
The solution is, in general, a local optimal one, not a global optimal solution. .... [3]