Ben Anderson
CSCI 379B
HW01 – Question 4

The Hill Climbing algorithm is a local search algorithm that tries to identify the optimal state to be in. If you imagine plotting each state with its associated cost value in a bar graph, the algorithm searches either for a local maxima (a peak) or a local minima (a low point) depending on the constraints of the problem.

The algorithm begins by randomly picking a state to start at. If searching for a local maxima, the algorithm checks the neighbors of the state to see if one has a greater value than the current state. If it does, that becomes the new state and the process is repeated. If not, the current state is returned as being the optimal state to be in. A similar process occurs with local minima, except the algorithm is looking for neighbor states with a lower value. The reason this algorithm does not always find the best state to be in, but only a locally optimal state, is because once it reaches a state where all neighbors are less desirable, it stops, even if there is a state with a couple of neighbors in between that is more desirable to be at.

This is where simulated annealing comes into play, it is very similar to the hill-climbing algorithm except for when it reaches a local optimum, it uses probabilities to decide whether it should keep exploring or stay put.

**Note, the written responses for Questions 1, 2, and 3 are included as comments in the respective code files. **