Hill A7:

3. Hill Climbing algorithm looks at initial state and then see’s if it has reached the goal. If it has not then it will set the state as the new initial state. After this it will select a different state to make the new state and will look to see if it is closer to the goal state than the state it is at. If it is not better than the current state it will skip that state. It will then loop through this process until it finds the goal state. The issues with this are that it can find a “path” but it may not be the most optimal because it can run into a local maximum. To try to combat local maximums we can use the random restart method that will do what the hill climbing search does but will repeat this form random start states based on how many times the programmer wants the method to repeat. It will compare the goal states and return the most optimal solution. Again this method has an issue of not reaching a completely optimal solution as you could get multiple local maximums. To combat this we use simulated annealing. Initially the algorithm will go to a random state and will need to have a set temperature. After this we loop until a solution is found. Then the program will pick a new state by changing the solution. After this point is reached the program will cool temp by whatever interval the programmer has chosen until it reaches 0. Local beam search is going to be like hill climbing search but will use as many states as put by the programmer. It adds these states to a list then the program will loop through each state and finds all possible successors. It will add all of these to a priority que. The state that is not worth as much will be deleted from the list and will add new states that are like the better solutions. It will loop through this until it finds the optimum solution or the best one that state can find.

4. I believe that the Beam search algorithm is the most effective. Through my different iterations I found that it may be slower than random restart and hill climbing but it put out an optimal solution every time. It was also more effective than simulated annealing search because most times it did not take as much time and found an optimal solution.