Function Optimization With Local Search

1) In the N-Queens problem, we want to place N queens on an $N \times N$ board with no two queens on the same row, column, or diagonal. Come up with a value function and use hill climbing to try to solve the problem by minimizing this value function, starting with the configuration given below. Generate the successors of a state by moving a single queen vertically.

	A	B	C	D
1				
2	Q1		Q3	
3		Q2		Q4
4				

2) How would you approach the Traveling Salesman Problem if we wanted to find a good (but not necessarily the best) solution to it using hill climbing?

3) What are the advantages/disadvantages of local search methods (such as hill climbing and simulated annealing) compared to A*? For which kind of optimization problems should local search be preferred?

4) We want to find a local minima of the function $f(x) = 2x^3 - 3x^2 - x + 1$. Start with $x = 0$ and apply three iterations of the gradient descent algorithm, using a learning rate of $\alpha = 0.2$.

5) What are the advantages and disadvantages of carrying over the fittest two individuals to the next generation?

6) What are the advantages and disadvantages of running genetic algorithms with only mutations and no crossovers? How about only crossovers and no mutations?

7) How would you encode a state if you were using a genetic algorithm to solve the Traveling Salesman Problem but only wanted to use a straightforward crossover operation that switches prefixes of both parents' encodings (that is, we randomly pick a cutoff point in the encoding, use the encoding of the first parent up to that cutoff point, and use the encoding of the second parent after that cutoff point)?