Problem #1

Question: Provide an example of boards in which beam search with two start points does not give the same result as hill climbing with a single restart.

The local beam search algorithm begins with *k* randomly generated states. At each step, all successors of the *k* states are generated. The *k* successors from across all the boards with the *k* best score are passed to the next round, and the process is repeated. Unlike with the Hill Climbing algorithm (both without and without restart), local beam search allows sharing of successor quality information between the *k* states.

Overview of this Example

Heuristic Cost Function (*h*): The number of pairs of attacking queens. In this case, the algorithms are trying to minimize the heuristic function (i.e. $h = 0$). This is not Hill Climbing in the strictest sense of the interpretation. If we wanted to maximize a parameter, we could use heuristic, h', where h' is defined as:

$$h' = \sum_{i=1}^{8}(i-1) - h$$

The summation

$$\sum_{i=1}^{8}(i-1)$$

is used because it represents the maximum number of possible collision combinations:

$$7+6+5+4+3+2+1=28$$

You could then maximize h' to 28. For simplicity, I will discuss below minimizing h to zero due to its greater simplicity. However, the concept is identical regardless of which of the two approaches is used.
Queen Symbol in the Board: ♠

Board Overview

Board # 1:

	A	B	C	D	E	F	G	H
8							♠	
7					♠			
6		♠						
5				♠				
4						♠		
3								♠
2			♠					
1	♠							

Board # 1 – Initial Board for the Hill Climbing Algorithm and Initial State #1 for Local Beam Search.
Description: This board comes from the textbook *Artificial Intelligence: A Modern Approach* by Russell and Norvig (see page 123 in the third edition).

Heuristic Value of this Board:

$$h = 1$$

List of Pairs of Attacking Queens:

(D5, G8)

Importance of this Board:

Per Russell and Norvig, this board is a local minima. Any movement of a queen results in a higher heuristic function (i.e. $h > 1$). Hence, if Hill Climbing was run on this board, it would immediately terminate since it is at a local minimum.

Board # 2:

	A	B	C	D	E	F	G	H
8	18	12	14	13	13	12	14	14
7	14	16	13	15	12	14	12	16
6	14	12	18	13	15	12	14	14
5	15	14	14	♠	13	16	13	16
4	♠	14	17	15	♠	14	16	16
3	17	♠	16	18	15	♠	15	♠
2	18	14	♠	15	15	14	♠	16
1	14	14	13	17	12	14	12	18

Board # 2 – Restart Board for the Hill Climbing Algorithm and Initial State #2 for Local Beam Search.

Description: This board also comes from the textbook *Artificial Intelligence: A Modern Approach* by Russell and Norvig (see page 123 in the third edition). Those squares that do not contain a queen have numbers describing the heuristic cost if a queen in that column was moved into that space. For example, if the queen in cell B4 was move to B8, the heuristic cost, h, would be 12.

Heuristic Value of this Board:

$$h = 17$$

List of Pairs of Attacking Queens:

(A4, B3)	(A4, C2)	(A4, E4)	(B3, C2)	(B3, D5)	(B3, F3)	(B3, H3)	(C2, E4)	(C2, G2)
(D5, E4)	(D5, F3)	(D5, G2)	(E4, F3)	(E4, G2)	(F3, G2)	(F3, H3)	(G2, H3)	

Importance of this Board:

Per Russell and Norvig, the minimum heuristic cost by moving one queen within its column is 12 (see cells: B8, B6, E7, E1, F8, F6, G7, and G1).

Board Traversal Using Hill Climbing

Step #1: The hill climbing algorithm examines all possible successors of Board # 1. Per Russell and Norvig, no successors have lower heuristic costs. Hence, the algorithm terminates. Since it is not a goal state, it generates another random (i.e. restart) board (in this case Board # 2).

Step #2: The hill climbing algorithm examines Board # 2 and observes the eight successor states that have identical minimum value (i.e. B8, B6, E7, E1, F8, F6, G7, and G1). The algorithm chooses **only one** of the successors and then generates subsequent successors from that one state until a local minimum is found or the goal is reached.

Conclusion: The hill climbing algorithm did not investigate any possible solutions using Board # 1 even though it was much closer to the heuristic target than Board # 2. Moreover, Hill Climbing only ever followed one path through the board at a time.

Board Traversal Using Local Beam Search

Step #1: The local beam algorithm generates all possible successors for Board # 1 and Board # 2. Per Russell and Norvig, there are no moves within the same column of Board # 2 that yields a heuristic cost of less than 12. In contrast, there are multiple moves in Board # 1 that yield heuristic costs of less than 12. Examples include:

a. Move the queen from A1 to A3. The heuristic cost (h) of that successor is 3.
b. Move the queen from A1 to A8. The heuristic cost (h) of that successor is 3.
c. Move the queen from B6 to B8. The heuristic cost (h) of that successor is 3.

The successors are ordered by lowest heuristic cost. Since there are multiple (i.e. more than 2) successors from Board # 1 with a lower heuristic cost than 12, the two states chosen for the next generation are from Board # 1. Board # 2 is not considered at all in future rounds. Moreover, there are always **two states** being considered in parallel per round not only a single one.

Summary

In Hill Climbing, Board # 1 was immediately passed over in the first round (since it was a local minima) while in contrast Board # 1 became the entire focus of the Local Beam algorithm after the first round (since it was closer to the goal heuristic value). Moreover, Local Beam has two dependent states running after the first round while Hill Climbing only has a single series of successors it considers after the restart.