structuring questions for  
recursive solutions

# 0 structure of the problem

**0a. What problem are you asked to solve?** and  
**What does "size" mean in the context of that problem?**

For the length of the list, I am asked to find how many elements are in the given list. The “size” is any list, including empty lists.

**0b. What samples will test the procedure and what answers are required?**

The first sample to test the procedure is finding the length of the empty list, which should be 0. The second sample to test the procedure is finding the length of the list consisting of one element, which should be 1. Other samples/tests can be used, but should be one that can be figured out mentally and quickly without thought.

**0c. What information do you have to be told?**

I have to be told the list.

**0d. What is the structure of the answer you will produce?**  
**An integer? A list? A Boolean? A stack of disks?**

For the length of the list, I am to produce an integer.

# 1 structure of the recursive solution

**1a. What does Mr. Brooks know how to do?**  
**That is, what is the recursive sub-problem?**

Mr. Brooks knows how to do find the number of elements in a given list.

**1b. Which sub-problem will you ask Mr. Brooks to solve?** and  
**How will you use Mr. Brooks’s answer to construct your own?**

I will ask Mr. Brooks to find how many elements are in the given list, if the list is not a empty list.

# 2 structure of the base case solution

**2a. What is the base case problem?**

The base case problem is the opposite, if the recursive case problem is false in a case. In this case, it would be if the list is not an empty list.

**2b. How can you solve the base case problem?**

I can solve the base case problem by writing out the circumstance when the recursive case problem does not apply.