1) What must all recursive functions have?

2) Take a look at the following recursive function.

```python
def string_mystery(s):
    if len(s) <= 1:
        return s
    else:
        one = s[0] * len(s)
        rest = string_mystery(s[1:])
        return one + rest
```

a) What code represents the base case?

b) Recursive Case
   i) Divide: What is the simple piece of the problem we can handle now? What is the harder piece that is a smaller version of the same problem?
ii) Recurse: What do we do to recurse?

iii) Combine: How are the pieces combined to form the solution to the problem?

c) List the recursive calls and the return values for the function if the function is called with the argument ‘ABCD’.

d) What does the function output?

e) Describe what the function does in words.
3) Writing a recursive function to sum the numbers in a list.
   a) What should the results look like?

b) **Computation Thinking:**
   i) Base Case: In words not code, what is the base case?

   ii) Recursive Case:
       (1) Divide: In words not code, how should the problem be divided?

       (2) Recurse: What does the recursive call look like?

       (3) Combine: How should the pieces be recombined?
c) Write the code.