Quiz #7: Recursion

Read the code for the recursive function `count_odd`, which operates on a nested list.

```python
from typing import Union, List

def count_odd(obj: Union[int, List]) -> int:
    """ Return the number of odd numbers in <obj>. """

    if isinstance(obj, int):
        return obj % 2  # remainder of L / 2
    else:
        for element in obj:
            return count_odd(element)
```

Answer the questions below about the following client code.

```python
>>> count_odd([1, [2, 6, 5], [9, [8, 7]]])
```

1. Based on the *structure* of the argument, state the three relevant recursive calls in this case, and what each one *should* return assuming `count_odd` is implemented correctly.

2. Now, assuming all recursive calls are correct, what recursive call(s) does this implementation of `count_odd` actually make, and what does the overall call to `count_odd([1, [2, 6, 5], [9, [8, 7]]])` actually return?

3. Write a correct implementation of `count_odd`.

*If you need more space, continue on the back of this page.*