CS 61A Exam-Prep Section – Week of 9/10

Topics: recursion fill-in, tree recursion fill-in

Attendance link: links.cs61a.org/295

Print Numbers

```
def print_numbers(n, k):
    """Print all numbers that (A) can be formed from the digits
        of `n` in reverse order and (B) are multiples of `k`.

    This is essentially Fall 2015 Midterm 2 #3c written to not
        depend on knowledge of lists.

    Args:
        n (int): The number that results must use digits from.
        k (int): The number that results must be multiples of.

    >>> print_numbers(97531, 5)
    135
    15
    35
    >>> print_numbers(97531, 7)
    1379
    357
    35
    >>> print_numbers(97531, 2)
    """
    def inner(n, s):
        if n == 0:
            if _____:
                _____
        else:
            _____
            _____
    inner(n, 0)
```

Sixty Ones

```
def sixty_ones(n):
    """Return the number of times that a 1 directly follows a 6
        in the digits of `n`.

    This is essentially Fall 2014 Midterm 2 #3a written to not
        depend on knowledge of lists.

    Args:
        n (int): The number whose digits are to be examined.

    Returns:
        int: The number of occurrences.

    >>> sixty_ones(461601)
    1
    >>> sixty_ones(161461601)
    2
    """
    if _____:
        return 0
    elif _____:
        return _____
    else:
        _____
```

No Elevens

```
def no_elevens(n):
    """Return the number of `n`-digit numbers whose digits
        consist of 1's and 6's and do not contain a `1` and
        then another `1` consecutively.

    This is essentially Fall 2014 Midterm 2 #3b rewritten to
        not depend on knowledge of lists.

    Args:
        n (int): The length of the numbers.

    Returns:
        int: The number of numbers.

    >>> no_elevens(2)  # 66, 61, 16
    3
    >>> no_elevens(3)  # 666, 661, 616, 166, 161
    5
    """
    if n == 0:
        return _____
    elif _____:
        return _____
    else:
        return _____
```