CSCI 241
Scott Wehrwein
Recursion: Executing Recursive Methods
Goals

Understand how recursive methods are executed.
Why are we talking about recursion, I thought we were sorting things?

```python
mergeSort(A, start, end):
    if (end - start < 2):
        return
    mid = (end + start)/2
    mergeSort(A, start, mid)
    mergeSort(A, mid, end)
    merge(A, start, mid, end)
```
How do we execute recursive methods?
How do we execute non-recursive methods?

\[ x = \text{max}(1, 3) \]

\[ \Rightarrow 3 \]
How do we execute non-recursive methods?

\[ x = \max(1, 3) \]

The function call is replaced by its return value.
How do we execute recursive methods?

```python
/** return n!; pre: n >= 0 */
fact(n):
    if n == 0:
        return 1
    return n * fact(n - 1)

fact(3)
=> 3 * fact(2)
    => 2 * fact(1)
        => 1 * fact(0)
            => 1
```
How do we execute recursive methods?

```python
/** return n!; pre: n >= 0 */
def fact(n):
    if n == 0:
        return 1
    return n * fact(n - 1)

fact(3)
=> 3 * fact(2)
  => 2 * fact(1)
    => 1 * fact(0)
        1

```
How do we execute recursive methods?

```python
/** return n!; pre: n >= 0 */

fact(n):
    if n == 0:
        return 1
    return n * fact(n - 1)

fact(3)
=> 3 * fact(2)
=> 2 * fact(1)
=> 1 * 1
```
How do we execute recursive methods?

```python
/** return n!; pre: n >= 0 */

fact(n):
    if n == 0:
        return 1
    return n * fact(n - 1)

fact(3)
=> 3 * fact(2)
=> 2 * fact(1)
    1
```
How do we execute recursive methods?

/** return n!; pre: n >= 0 */

```python
fact(n):
    if n == 0:
        return 1
    return n * fact(n - 1)
```

```python
fact(3)
=> 3 * fact(2)
2
```
How do we execute recursive methods?

```python
/** return n!; pre: n >= 0 */

fact(n):
    if n == 0:
        return 1
    return n * fact(n - 1)

fact(3)
=> 6
```