

# Recursion

---

## Announcements

Self-Reference







## Discussion Question: Factorial Two Ways

---

Rewrite `fact(n)` so that the result of `fact(5)` is computed using the following steps:

```
5 (1 * 5)
20 (1 * 5 * 4)
60 (1 * 5 * 4 * 3)
120 (1 * 5 * 4 * 3 * 2)
```

```
def fact(n):
    """Compute n factorial.

    >>> fact(5)
    120
    >>> fact(0)
    1
    """
    if n == 0 or n == 1:
        return 1
    else:
        return fact(n-1) * n
```

## Discussion Question: Play Twenty-One

---

Rewrite play as a recursive function without a while statement.

- Do you need to define a new inner function? Why or why not? If so, what are its arguments?
- What is the base case and what is returned for the base case?

```
def play(strategy0, strategy1, goal=21):  
    """Play twenty-one and return the winner.
```

```
>>> play(two_strat, two_strat)
```

```
1
```

```
.....
```

```
n = 0
```

```
who = 0 # Player 0 goes first
```

```
while n < goal:
```

```
    if who == 0:
```

```
        n = n + strategy0(n)
```

```
        who = 1
```

```
    elif who == 1:
```

```
        n = n + strategy1(n)
```

```
        who = 0
```

```
return who
```

```
def play(strategy0, strategy1, goal=21):  
    """Play twenty-one and return the winner.
```

```
>>> play(two_strat, two_strat)
```

```
1
```

```
.....
```

```
def f(n, who):
```

```
    if n >= goal:
```

```
        return who
```

```
    if who == 0:
```

```
        n = n + strategy0(n)
```

```
        who = 1
```

```
    elif who == 1:
```

```
        n = n + strategy1(n)
```

```
        who = 0
```

```
    return f(n, who)
```

```
return f(0, 0)
```