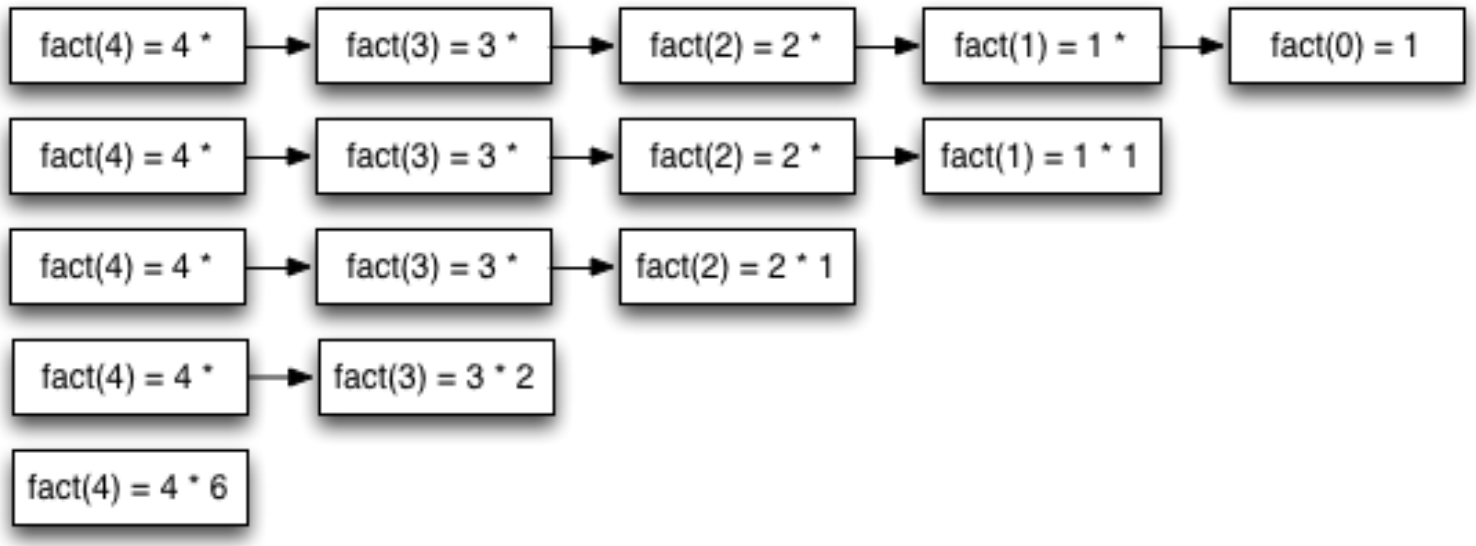
Lecture 28: Recursion

Factorial

```
1  def fact(n):
2      if n == 0:
3          return 1
4      else:
5          return n * fact(n-1)
```



Question 1: Exponentiation

Write a recursive version of exponentiation called `exp(n, k)` that computes n^k. Note that exponentiation is repeated multiplication.

```
>>> exp(2,0)
1
>>> exp(2,1)
2
>>> exp(2,2)
4
>>> exp(2,3)
8
>>> exp(2,10)
1024
```

Question 2: Product

Write a recursive version of production called `prod(L)` that computes the product of the numbers in the list `L`.

```
>>> prod(list(range(1,5)))
24
>>> prod(list(range(1,6)))
120
>>> prod(list(range(1,7)))
720
```

Question 3: Reverse

Write a recursive version of reverse called `rev(L)` that returns the members of `L` in reverse order.

```
>>> rev(list(range(10)))
[9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
>>> rev([1])
[1]
>>> rev([])
[]
```