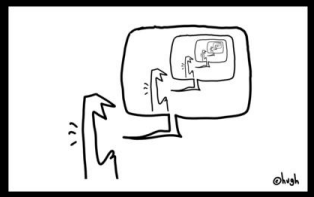


CS151 Fall 2014
Lecture 12 – 10/2

Recursive/Inductive Definitions

Prof. Tanya Berger-Wolf
http://www.cs.uic.edu/CS151

Adapted from Lap Chi Lau - The Chinese University of Hong Kong
and David Liben-Nowell - Carleton College

Factorial

How many multiplication operations will it take to compute ***n!*** ?

n! = 1*2*3*…*(n-2)*(n-1)*n

```
Factorial (n)
if (n == 0 or n== 1)
        return(1)
else
        return (n*Factorial(n-1))
```

```
Factorial (n)
i := 1
factorial := 1
while i < n
        i := i+1
        factorial := factorial * i
return(factorial)
```

Factorial(n) returns ***n!***

P(n) := "_____"

Base Case: P(?) := "_____"

P(n-1) :=

P(n) :=

Factorial

How many multiplication operations will it take to compute ***n!*** ?

n! = 1*2*3*…*(n-2)*(n-1)*n

```
Factorial (n)
if (n == 0 or n== 1)
        return(1)
else
        return (n*Factorial(n-1))
```

```
Factorial (n)
i := 1
factorial := 1
while i < n
        i := i+1
        factorial := factorial * i
return(factorial)
```

Factorial(n) uses ____ multiplication operations

Recursive (Inductive) Definitions

We completely understand the function Factorial(n) = n!, right?

As a reminder, here's the definition:

$$n! = 1 \cdot 2 \cdot 3 \cdot \ldots \cdot (n-1) \cdot n,\ n \geq 1$$

But equivalently, we could define it like this:

Recursive Case / Base Case

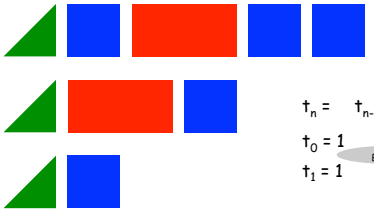
$$Fac(n) = \begin{cases} n \cdot Fac(n-1) & \text{if } n > 1 \\ 1, & \text{if } n = 1 \end{cases}$$

Inductive (Recursive) Definition

Recurrences

A train is defined to be an engine, followed by cars of two different kinds…long and short. Long cars are 2 units long, and short ones are 1.

Examples:



How many trains of length n are there?

$t_n = t_{n-1} + t_{n-2}$ Recursive Case

$t_0 = 1$

$t_1 = 1$ Base Case

We still don't really know.