61A Lecture 4

Announcements

Iteration Example

The Fibonacci Sequence

def fib(n):
    """Compute the nth Fibonacci number, for N >= 1."""
    pred, curr = 0, 1  # Zeroth and first Fibonacci numbers
    k = 1
    while k < n:
        pred, curr = curr, pred + curr
        k += 1
    return curr

Discussion Question 1

Designing Functions

Characteristics of Functions

def square(x):
    """Return x * x."""
    x is a real number
    returns a non-negative real number
    A pure function's behavior is the relationship it creates between input and output.
    return value is the square of the input

def fib(n):
    """Compute the nth Fibonacci number, for N >= 1."""
    n is an integer greater than or equal to 1
    returns a Fibonacci number
    A pure function's behavior is the relationship it creates between input and output.
    return value is the nth Fibonacci number

A Guide to Designing Function

Give each function exactly one job.

Don't repeat yourself (DRY). Implement a process just once, but execute it many times.

Define functions generally.
Generalization

Regular geometric shapes relate length and area.

Finding common structure allows for shared implementation

Higher-Order Functions

The common structure among functions may be a computational process, rather than a number.

Summation Example

Functions defined within other function bodies are bound to names in a local frame

Call Expressions as Operator Expressions