Algorithm
- Series of steps
- transforms inputs
- has outputs

Efficiency
- Running time
- Memory
- Quality
- Simplicity

Run time
- depends on input
- also depends on data
  (sorted vs. unsorted)
  (structured vs unstructured)
- Best/Worst/Average cases

Theoretical Analysis:
- We need to develop a general methodology
- Running time as a function of input size
- independant of environment
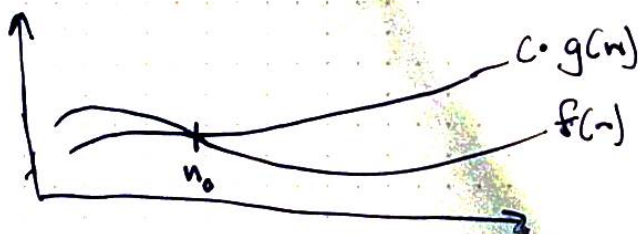
Primitive Operations:
Low-level comparisons from the programming language can be identified in pseudocode

By inspecting pseudocode we can count the number of primitive operations executed by an algorithm

Big-Oh (upper-bound)

given $f(n)$ and $g(n)$, $f(n)$ is $O(g(n))$ iff there are positive constants c and n_0 such that

$$\{ f(n) \leq c \cdot (g(n)) \text{ for } n \geq n_0 \mid c > \emptyset \}$$

after some point n_0, $c \cdot g(n)$ is always greater than $f(n)$.