Goals of 6.00

Prepare freshmen and sophomores with no prior programming experience for entry into Course 6
Help students feel justifiably confident of their ability to write small and medium sized programs
Understand how to map problems into a computational framework
Position students to compete successfully for UROPs and summer jobs
Declarative knowledge is composed of statements of fact

“A good health care plan improves the quality of medical care while saving money”

“$y$ is the square root of $x$ if and only if $y\cdot y = x$”
Imperative Knowledge

Imperative knowledge is about how to accomplish something. Think of it as a recipe.

1) Start with a guess, $g$

2) If $g \times g$ is close enough to $x$, then $g$ is a good approximation of the square root of $x$

3) Otherwise, create a new guess by averaging $g$ and $x/g$. I.e., $g_{\text{new}} = (g_{\text{old}} + x/g_{\text{old}})/2$

4) Using this new guess, go back to step 2
Stored Program Computer

Treat data and instructions as the same thing.
Syntax, Static Semantics, and Semantics

Syntax: which sequences of characters and symbols constitute a well-formed string

Static semantics: which well-formed strings have a meaning

Semantics: what that meaning is
Compiled vs. Interpreted

**Interpreted**

source code $$\rightarrow$$ checker $$\rightarrow$$ interpreter $$\rightarrow$$ output

**Compiled**

source code $$\rightarrow$$ checker/compiler $$\rightarrow$$ object code $$\rightarrow$$ interpreter $$\rightarrow$$ output