Unit 2: Programming

Dave Abel

February 10th, 2016
Logic: Reflection

1. There is a correct notion of reasoning (logic)

2. At their core, computer’s are doing logic via gates and electrical pulses.
Programming: Takeaway

1. Physical gates are inflexible.

2. Programming lets us reconfigure what a computer does!
Trouble in Gateland?

\[ OR(P, NOT(Q)) \]
Trouble in Gateland?

Physically represents \( OR(P, NOT(Q)) \)
Trouble in Gateland?

Physically represents $OR(P, NOT(Q))$

“Hardware” logic. **Extremely fast.**
Trouble in Gateland?

Physically represents $OR(P, NOT(Q))$

Q: What if we want to reconfigure things?
Trouble in Gateland?

\[ OR(P, \neg Q) \]

Q: What if we want to reconfigure things?

A: Programming.
Programs

**Central Idea:** the hardware does not have to change for a computer to change its behavior.
Programs

**Central Idea:** the hardware does not have to change for a computer to change its behavior.

A fixed set of circuits can change its behavior to represent any desired function! Build one, reprogram into anything.
Programs

Central Idea: the hardware does not have to change for a computer to change its behavior.

A fixed set of circuits can change its behavior to represent any desired function! Build one, reprogram into anything.

Drawback: much slower.
Programs: An Extension of the Will
Programs: An Extension of the Will

Idea!
Programs: An Extension of the Will
Programs: An Extension of the Will

Idea!

Action!
Programs: An Extension of the Will

With programs, I can delegate my idea to the computer
Programming

‣ Lots of languages!

‣ Each language provides a different way to write commands to the computer.

‣ They all do basically the same thing…
Programming

‣ Lots of languages!

‣ Each language provides a different way to write commands to the computer.

‣ They all do basically the same thing…

‣ Allow us to access the wonderful world of computation!

Things that can be computed, period.

Things a regular computer can compute before the sun goes supernova

Dominos!
This Class: Scratch

• Developed by the “Lifelong Kindergarten” group at MIT

• Sort of like LEGO! Clip together blocks.

• Edit/run in your browser:

  scratch.mit.edu
This Class: and Python!

- End of term extra-credit project.
- Several optional workshops to learn python
- Python: a rich language that looks an awful lot like english!

```python
>>> print "Hello CS8!"
Hello CS8!
```
This Class: and Python!

- End of term extra-credit project.
- Several optional workshops to learn python
- Python: a rich language that looks an awful lot like english!
  (also it’s my favorite)
Mostly: Scratch!

Let’s take a look!
Demo 1: A Square

Let’s take a look!
Demo 2: Loops

Repeats!
Demo 2: Loops

Repeats!
Demo 2: Loops

Idea: tell the computer to repeat!

Let's take a look!
Demo 3: Logic

Anything with this shape:

evaluates to True or False
Demo 3: Logic

Let’s take a look!

Anything with this shape:

evaluates to True or False
Demo 4: Making Blocks

Let’s take a look!

Make a Block

bounce!

Add an Extension
Demo 5: Variables

Let's take a look!
Demo 6: Conditions and Loops

Let’s take a look!
Demo 7: Coin Flipping!

And more… Let’s take a look!
Things You’ll Do in Scratch

‣ **Machine Learning:** Write a classifier, similar to how your email determines what is “Spam” and what is “Ham”!

‣ **Programming:** A simple game

‣ **Vision + NLP:** Model Roald Dahl’s style of writing!

‣ **Recursion:** Draw recursive pictures

‣ And more…
Block Types