Whoa! That homework may have been too much

UW’s Pilot

Lawrence Snyder [with Susan Evans and Brandon Blakeley]

University of Washington, Seattle
CSE120: Computer Science Principles

“Must-know computing knowledge for 21st Century”

Credits: 5

3 Lectures, 2 Labs (Closed)

Pre-requisites: None

Follow-on Classes: None require it (yet)

Implementation of 7 Big Ideas and 6 Comp Practices

- Thread 1: Principles, such as all information encoded in bits
- Thread 2: Capabilities, such as CT, abstraction, program’ g

http://www.cs.washington.edu/cse120/
Thread 1: Principles

- Principles covered in UW CSP –
  - **Bits**: sufficient to encode all information
  - **Binary**: like decimal but with radix 2, not 10
  - **Info**: physically is presence/absence of phenomenon
  - **Functional abstraction**: enables software layering
  - **Meta-data**: enables automatic processing of info
  - **TCP/IP**: like sending novel by postcards
  - **EtherNet**: like cocktail party chat
  - **Privacy**: right of people to decide the extent ...

... Content people should know, direct from 7BI & 6CP
Thread 2: Capability

- What we do in UW CSP –
  - Programming in several forms
    - LightBot, an introduction to programming & recursion
    - Processing, graphic-centric design language
    - XML data structuring and personal database design
    - ... also some HTML, CSS, Scratch and other software
  - Functional abstraction, recursion
  - Creating artifacts to implement personal intent
  - Repurpose tools for own use; programming by analogy
  - ...

Practices that reinforce principles ... be bold, creative, exploratory
Programming Experience

- **Week 1**: Lightbot … it’s a fun game & it’s

Students write recursive code before the opening lecture
Processing

- Graphics prototyping language built on Java

```java
void setup() {
    // Snow Angel
    size(400, 400);
    stroke(255);
    background(0, 0, 255);
}

void draw() {
    line(150, 150, mouseX, mouseY);
}
```

check it out!
Processing, A Pedagogical Wonder

- Processing is ...
  - “Totally fun!” Students are engaged by it
  - Free and trivial to install: all students pgm on their own machines, having installed SW themselves
  - Graphics are fun and trivial to do; interaction is trivial; text is actually harder
  - IDE is very forgiving despite being equiv. to Java
  - Trivially export a Web-embeddable version of code
  - Makes all standard programming concepts available in standard form
Each week students are asked to fill out the After Image Survey (AIS) – free form
- What was engaging?
- What worked?
- What didn’t work?
Susan Evans (my HS teacher) summarizes and sends me a “report card”
Probably not scientifically reliable, but it’s good to measure the “temperature” of class
Students say: “I didn’t expect to like it; I do!”
“I didn’t expect I could do it; I can!”

Complaint: Assignments too long; “unclear”

My Prob: Sequencing ... too much adv’ed prep

One Reward: Teaching CS ideas for own sake;
contrast Fluency with IT, CS ideas you can use

Challenge: students work harder than they’re used to ... need to keep it interesting, fresh
Links