CS1/CS2 @ UBC

Gail C Murphy
UBC

large public research institution

~47,000 students across two campuses

UBC Computer Science

60+ faculty including 8 teaching faculty

Majors (BSc)
Combined Majors (BSc, BA, Biz)
BCS
• <Statistics graph>
“Old” CS1/CS

CS1
Java-based
Largely syntax driven
Lecture and labs (13 weeks)

CS2
Java-based
Small-scale software design
Lecture and labs (13 weeks)

Big Java book
New CS1/CS2

knowledge and skills to write programs

understand more deeply how computation and programs work

vital for scientists and other professionals

abstraction and decomposition manifest repeatedly to build larger computations and systems
CS1 (Computations, programs and programming)

HtD = How to Programs (Felleisen, Findler, Flatt and Krishnamurthi

recipes provide problem solving structure for all students
atomic data through to recursive data definition
functional and data abstraction
...
unit testing built into approach

lectures (3 hrs per week) and labs (3 hours per week)
builds towards programming Pacman on Androids

Dr. Ratchet
CS2 (Software construction)

use models to explain code
produce and explain OO design exhibiting good
design principles from NL description
produce robust implementation given spec

from “Ratchet” (~Scheme) to Java by reading code

lectures (3 hours per week) and labs (2 hours per week)
builds toward multi-week lab to extend
open-source system running on Android
marking in lab (i.e., discussing your code)

Eclipse
CWSEI

four-step scientific approach to teaching
1. establish what students should learn
2. scientifically measure what is being learned
3. adapt methods to achieve desired outcomes
4. disseminate and adopt what works

studies being conducted in new CS1/CS2 longitudinal studies being planned
CS1 current web site:
http://sites.google.com/site/ubccpsc1102010w1

CS2 current web site:
http://sites.google.com/site/ubccpsc2102010w1/Home