

---

## CSE 142

Welcome!  
Organization & Administration

1/6/2002

(c) 2001-2, University of Washington

A-1

---

## Outline for Today

- Course Overview
- Administrative details
- Workload and grading
- Resources

This information (and more) is included in today's handouts, and is on the web – no need to transcribe

1/6/2002

(c) 2001-2, University of Washington

A-2

---

## Why Are We Here?

- Computers are everywhere!
  - Big ones serving databases and forecasting the weather
  - Medium sized computers on your desk top, for playing games, writing papers, surfing the internet
  - Tiny ones everywhere: cars, microwaves, toys, phones
- They're part of our world
- What can they do? How do they do it? What can't they do?

1/6/2002

(c) 2001-2, University of Washington

A-3

---

## Two Amazing Facts

1. Computers are multi-purpose
  - Unlike cars, toasters, dishwashers
  - The same physical computer can play games, solve equations, plan trips, send e-mail, etc. How is this possible??
  - Answer: the computer operates under direction of a "program": a set of precise instructions
2. The largest and the smallest computers have much in common
  - We can usefully think of about computers in general without worrying about hardware details
  - This is our first example of "abstraction", a key notion in computer science

1/6/2002

(c) 2001-2, University of Washington

A-4

---

## Computers & You & CSE142

- You'll learn to write programs
  - We use a particular language called Java
  - The principles apply to many other languages
- You'll use particular computers
  - Probably Windows PCs
  - Principles apply to many other computers
- Along the way, we may talk about computers in our world

1/6/2002

(c) 2001-2, University of Washington

A-5

---

## Staff

- Instructors: Hal Perkins (9:30) & Donald Chinn (11:30)  
cse142-instructors@cs.washington.edu
- TAs
  - cse142-tas@cs.washington.edu
- Pim Lustig: course administrator  
cse142-admin@cs.washington.edu
- Consultants: they are savvy students we've hired to help out in the lab. Once we get their hours worked out, we'll post a schedule.  
cse142-staff@cs.washington.edu reaches entire staff

1/6/2002

(c) 2001-2, University of Washington

A-6

## Teaching Assistants

- Nilesh Dalvi
- Kaustubh Deshmukh
- Richard Dunn
- Sangyun Hahn
- Susan Hewitt
- Daryl Jump
- Lin Liao
- Nathin Ratliff
- Jared Roberts
- Sumit Sanghai
- Kenneth Tam
- Adrien Treuille
- Jia-chi Wu
- Zizhen Yao

1/6/2002

(c) 2001-2, University of Washington

A-7

## What To Expect

- Course is for beginners
- Programming is quite different from using applications
  - Logic/problem solving skills
  - Can be challenging, but also very rewarding
- Important to keep up
  - Ask for help when you need it; don't fall behind
- If you have the background to skip this course, you can go directly to CSE 143
  - Automatic credit for CSE 142 after completing CSE 143 successfully
  - Both C++ and Java versions of CSE 143 this quarter; Java version only after that

1/6/2002

(c) 2001-2, University of Washington

A-8

## What's New This Year?



- Java!
- Goal of the course remains to teach programming, not a particular programming language, but...
  - A more modern view of programming
  - Cleaner, more comprehensible language
- Fundamental concepts remain (loops, conditionals, methods [functions], etc.)

1/6/2002

(c) 2001-2, University of Washington

A-9

## Course Organization

- 3 lectures per week (MWF)
- Quiz section once per week (Thursday)
  - Exercises, review, discussions, etc.
  - Regular quizzes (easy to do if you keep up)
- Designated quiz sections:
  - Low-background
  - High-background
  - All sections have the same assignments, take the same tests, and are graded the same
  - More about this on Wednesday. Meantime, be sure you know which quiz section you're in, and which kind it is.

1/6/2002

(c) 2001-2, University of Washington

A-10

## More About Quiz Sections

- Low-background: minimal experience; feel more comfortable with others in the same situation
- High-background: want to go into additional technical details, etc.
- All sections have the same assignments, take the same tests, and are graded the same
- Possible to informally switch sections with permission of TAs involved – no registration change needed

1/6/2002

(c) 2001-2, University of Washington

A-11

## Workload

- Regular assignments
  - Due at the beginning of most weeks
  - Fixed deadlines, no leeway, no make-ups
  - Mix of problem types: short programming drills, longer programming projects, short answers, some writing, some diagrams...
- Exams
  - 2 midterm exams in class; tentative dates: Fri. Feb. 1 & Fri. Feb 22
  - Final exam during finals week: **Wednesday, March 20**  
No other day. You must plan to be here for the final exam
  - Short quizzes now and then
  - What are tests like?  
some programming, some short answer, some multiple choice...

1/6/2002

(c) 2001-2, University of Washington

A-12

## Grading

- **Grade distribution (subject to change)**
  - 30% homework assignments
  - 20% + 20% midterm exams
  - 25% final exam
  - 5% quizzes & other
- **Assignments are your opportunity to learn; exams are used to assess how well you've mastered material**
- **Assignment and quiz grading will be very coarse**
  - Example: 2, 1, 0 for assignments
- **Median of final course grades is just below 3.0**

1/6/2002

(c) 2001-2, University of Washington

A-13

## Academic (Mis)conduct

- **Goal: balance the following**
  - Learning
  - Cooperation
  - Fairness and honesty
- **Policy**
  - You must do assignments by yourself (unless explicitly stated otherwise in an assignment)
  - You may discuss things with others, but
  - You may not *ever* trade code with others, or include code written by someone else in your assignments, etc.
- **We check this and act when trouble is discovered**

1/6/2002

(c) 2001-2, University of Washington

A-14

## Resources

- **Course staff**
  - Please feel free to talk to any TA or instructor
- **Lecture slides**
  - Will normally be posted on the web sometime after class
- **Course notes: Ben Dugan, *An Introduction to Programming in Java***
  - Required
  - Buy at Professional Copy and Print, 42<sup>nd</sup> and U. Way (or available on-line)
- **Textbook: Niño & Hosch, *An Introduction to Programming and Object-Oriented Design in Java*, Wiley, 2002**
  - Recommended, not required
  - Supplementary source of explanations, examples, and software
  - Also used in CSE143

1/6/2002

(c) 2001-2, University of Washington

A-15

## Communications

- **Course web site**
  - [www.cs.washington.edu/education/courses/142](http://www.cs.washington.edu/education/courses/142)
- **Newsgroup: uwash.class.cse142.bboard**
  - Open discussion – please contribute!
  - Course staff monitors and contributes as needed
- **Email**
  - Addresses on the web

1/6/2002

(c) 2001-2, University of Washington

A-16

## Computing Facilities

- **CSE142 uses the UWired general labs**
- **Primary location for course consultants is the Introductory Programming Lab (IPL), 3<sup>rd</sup> floor Mary Gates Hall (MGH)**
  - Pay a visit there today!
  - Can also use machines in Computing Commons in MGH and Odegaard (OUGL)
- **Computing at home**
  - Course software and tools are freely available for download
  - Support information on the CSE 142 web

1/6/2002

(c) 2001-2, University of Washington

A-17

## Can't Get In?

- **New slots open up as people drop**
- **No waiting list**
- **No entry codes**
- **Attend lectures and any old quiz section for the time being. But no guarantees – you might not get in.**
- **If you aren't registered by Wednesday or so – consider making a new plan**

1/6/2002

(c) 2001-2, University of Washington

A-18