
CSE 143 Java

Welcome!

9/30/2002

(c) University of Washington

00 1

Outline for Today

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

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

9/30/2002

(c) University of Washington

00 2

Staff

- Instructor: Martin Dickey
dickey@cs.washington.edu
Sieg 423b, office hours TBA
- TA's: Kuang Chen, Danielle Farrar, Chris Fitzner, Sam Kim, Jesse Kinkead, Zach Krisman, Eric Nordberg, Parag, Dmitry Portnov, Gary Ingve
cse143-ia@cs.washington.edu (goes to all TAs and the instructor)
- Consultants: once we get their hours worked out, we'll post a schedule.
- Course administrator: Pim Lustig
cse143-admin@cs.washington.edu
- Everyone:
cse143-staff@cs.washington.edu reaches entire staff

9/30/2002

(c) University of Washington

00 3

Can't Get In?

- Lots of space at 8:30! Tell your friends to sign up.
- No waiting list/entry codes

9/30/2002

(c) University of Washington

00 4

Java!



- A modern approach to programming including
 - Objects everywhere
 - Exceptions
 - Streams
 - Specifications, design by contract
 - Rich set of standard libraries
- We'll use Sun's Java SDK 1.4.01 (1.4.02 OK)
 - Please update your software!

9/30/2002

(c) University of Washington

00 5

Are You Ready?

- Course is a direct continuation of CSE 142 Java
- Must have a firm grasp of Java basics including declaring classes, creating objects, statements, methods, etc.
 - No systematic review
 - Look at old CSE 142 web pages – you should be able to handle those assignments and exams
- What if you took the C version of CSE 142?
 - Let's talk about that now
 - Not sure?
 - Sit in on both for a few days
 - Try the first 143 assignment

9/30/2002

(c) University of Washington

00 6

Content Overview

- Classes & interfaces
- Inheritance and class hierarchies; frameworks
- Java user interfaces & event-driven programming
- Exceptions
- Stream I/O (files and networks)
- Data structures (lists, queues, stacks, trees, tables)
 - Multiple implementations
- Overview of complexity & implementation tradeoffs
- Recursion
- Searching & Sorting
- And Much Much More!

9/30/2002

(c) University of Washington

00-7

Course Goals

- Learn more programming, and glimpse what computer science is beyond programming
- A key goal is learning to program well
 - Good design, good organization, good style
 - Good algorithms, good efficiency
- Not about learning all Java details
 - Some Java features we won't cover – see Java reference books for full descriptions of the Java language
 - We cover the parts of Java that support good programming
 - We'll talk about broader issues, too
- Fact: writing programs that work perfectly isn't enough to get a perfect grade (!)

9/30/2002

(c) University of Washington

00-8

My Goals for You

- Take you to the next technical step in programming
- Challenge you with material of considerable intellectual content, and with projects of considerable complexity.
- Set a foundation of computer concepts to prepare you for further study
- Develop your ability to learn independently
- Develop your ability to learn cooperatively
- Keep us aware of larger issues surrounding the use of information technology in our world
- If possible, make it fun. If possible...

9/30/2002

(c) University of Washington

00-9

My Goals For Myself

- Top goals for the course:
 - Help all of you learn
 - Keep the course on track
 - Make lecture and section events you look forward to!
- Plus some more personal goals...
 - Learn some more Java myself
 - Make better use of technology in the classroom
 - Refine some teaching techniques
 - Take lots of pictures
 - And... learn a bunch of names!
- And... one goal to be named later

9/30/2002

(c) University of Washington

00-10

Course Organization

- 3 lectures per week (MWF)
- Quiz section twice per week (T & Th)
 - Exercises, review, discussions, etc.
 - Be sure to enroll in one of the quiz sections for this lecture (Jx)
- Frequent quizzes
 - To keep you up with the reading
 - To test mastery of current material
 - To provide TAs and me with feedback

9/30/2002

(c) University of Washington

00-11

Assignments

- Regular assignments
 - Typically (but not always!) due Wed. night 9pm (electronic) and/or in sections Thursday morning (written)
 - Some written problems and short programming drills
 - Programming projects – substantially more complex than in CSE142
 - No late assignments accepted

9/30/2002

(c) University of Washington

00-12

Academic (Mis)conduct

- Goal: balance the following
 - **Learning:** each student must do the work to learn effectively
 - **Cooperation:** people learn best when they can cooperate with others
 - **Fairness and honesty:** Nobody should ever represent the work of someone else as their own or try to claim credit for it
- Policy
 - You must do assignments by yourself (unless explicitly stated otherwise in an assignment)
 - You may discuss general approaches and ideas with others, but
 - You **may not ever** give code to or receive code from others
- We check this and act when trouble is discovered
- Use your common sense and ask first if unclear

9/30/2002

(c) University of Washington

00-13

Exams & Quizzes

- Exams
 - 2 midterm exams in class; tentative dates: TBA
 - Final exam: Tuesday, December 17
- The exams will not be given on any other days. Don't make plans which would take you away!
- Format: mixture of short answer, short essay, multiple choice, programming

9/30/2002

(c) University of Washington

00-14

Grading

- Grade distribution (subject to change)
 - 28% homework assignments
 - 14% + 16% midterm exams
 - 24% final exam
 - 12% quizzes
 - 6% participation and service
- Class is curved
 - Median of final course grades is just below 3.0
 - Why?

9/30/2002

(c) University of Washington

00-15

Grading

- The parts of the course have different goals and styles
 - Tests and projects may seem disconnected from one another
 - Each measures things that the other can't
 - Tests may seem hard even when homework doesn't!
 - Lectures cover concepts and examples; will rarely talk about homework
 - Quiz sections are for active learning, practice, and review of recent topics
- Assignment and quiz grading will be very coarse
 - No partial points
 - 4, 3, 2, 1, 0 for assignments
 - Mastery || Good Job! || On the Right Track || Honest Effort, but... || Really, Now!
 - Separate scores for Operation/Practice
 - Quiz questions usually right or wrong

9/30/2002

(c) University of Washington

00-16

Resources to Help You Succeed

- Course staff
 - Your TA is your primary contact, but please feel free to talk to any of us
 - I'll try to be available right after class (not before!) for as long as there are questions
 - Consultants in the IPL
 - A limited resource!
- Help each other! Form study groups, spend time on the bulletin board, etc. Of course, within the academic conduct guidelines
- Undergraduate advisors, for general questions about the CSE programs (Sieg 114)
- College of Engineering has some special resources for women and minorities

9/30/2002

(c) University of Washington

00-17

For Reading and Study

- Lecture slides and course notes
 - **Alert!** Not all lecture material is on the slides!
 - Slides used will be posted on the web
 - Probably NOT distributed in lecture
- Textbook: Nino & Hosch, *An Introduction to Programming and Object-Oriented Design using Java*, Wiley, 2002
 - **Alert!** We won't follow the book very closely!
 - Source of additional explanations and examples
 - Covers material from both CSE142 & CSE143 – good review source
 - Will not always match our way of doing things, or our order!
- Other Material
 - Possibly handouts
 - All announcements, assignment descriptions, etc. should be considered required reading. They could even be tested on!

9/30/2002

(c) University of Washington

00-18

Communicate!

- UWNetID required
- Course web site
 - www.cs.washington.edu/education/courses/143/02wi-java/
- Message Board: will be linked from Web Site
 - Open discussion – please contribute!
 - Course staff monitors and contributes as needed
- Email to us
 - Addresses on the web
 - Email works better for some things than other
- E-mail from staff
 - We'll try to keep the spam to a minimum, but... please read what we do send!

1/30/2002

(c) University of Washington

00-19

Computing Facilities

- Introductory Programming Lab (IPL)
 - Mary Gates Hall 334
 - CSE 143 consulting staff in IPL
 - Hours posted on the web
 - Goal is to provide quick help when you're stuck and have already tried to diagnose and fix the problem
- Computing at home
 - Java software and tools are freely available for download
 - Java version MUST be 1.4. Install entire SDK
 - You're free to use any Java development environment
 - Recommended: BlueJ or DrJava
 - Even if you plan to compute at home, learn your way around the UW labs

1/30/2002

(c) University of Washington

00-20

Your First Assignment

- Required reading: syllabus, academic conduct policy page. Do this before quiz section tomorrow!
- Browse rest of web (still pretty incomplete)
- Visit the bulletin board (when linked) and find the announcements archive (two separate things!)
- Install the needed pieces of software on your home computer and/or visit a campus lab and locate the software
- Read chapters 1-4 of the textbook
- Find a partner for Project 0. It must be someone you did not previously know
- Watch the web for more about Project 0.
- (After tomorrow) memorize your quiz section # and TA's name

1/30/2002

(c) University of Washington

00-21