Course Goals

• To provide the skills needed to:
  – deconstruct relevant programming problems and
  – solve them in an object-oriented style.

Approach

• Methods and styles of (sequential) object-oriented (OO) programming.
  – abstraction, particularly in the use of interfaces and
  – design patterns to improve reusability and reliability
  – will use the Java programming language exclusively
  – homework projects will build on each other
• Principles and practice of concurrent OO programming. Will cover the two main forms of concurrency:
  – shared-memory multi-threading (as provided by Java Thread class), and
  – distributed message passing (as provided, along with more features, by Java Remote Method Invocation (RMI)).

The Waitlist (if any)

• I am unlikely to accept anyone from the waitlist
• But if I do, I expect to take in only:
  – CS undergraduate majors with GPA = 3.25 and 330 grade of B or higher
  – CS graduate students
• Send me a note
  – with GPA, grade in CMSC 330, other info

Office hours

• Professor Adam Porter (aporter@cs.umd.edu)
  – TuWTh 2:00-3:00PM
• TA: Hyoungjune Yi (aster@cs.umd.edu)
• Always posted on class webpage: www.cs.umd.edu/class/fall2002/cmsc433-0101
  – Or by appointment
• We will respond to questions only IF you provide 2 plausible answers to your question

Projects

• Projects due at 6pm on due date
  – by Unix time of day
  – you must submit a good-faith effort
    • you can be failed for the course if you do not
• Read the newsgroup – csd.cmse433
  – Only visible from inside UMD
  – Post public questions here, instead of using email
    • TA’s will read regularly, but may or may not respond. In any case, do not expect real-time responses. This is not a substitute for coming to office hours.
    • NB. As in the rest of life, don’t believe everything you read!!!
Projects (cont.)

- Class accounts will be emailed
  - to the email account registered with ???
  - so make sure you forward that account to wherever you read email
  - if you don’t receive an account by next Tuesday, talk to TA
  - We will have accounts on CSIC machines

submit

- Use online submit program to submit projects
  - e.g., submit 1 p1.h p1.cc
  - provide list of file names, not a tar ball
  - link to the submit program in grading account CSIC
    program name TBD
  - Submit early and often
  - Can provide recovery from previous submit

Test Cases

- You will be expected to provide unit test cases for your projects
- Will make use of JUnit testing package

Project Commentary

- You will be emailed two other submissions; respond with commentary on each
  - you will be graded on the usefulness of your commentary

Topics

- Java
- Programming techniques and tools
- Object-oriented design for sequential programs
  - OO principles
  - design patterns
- Concurrency
  - concurrent programming in Java
  - design patterns
- Distributed programming
  - distributed programming in Java
  - design patterns

Textbooks

- Primary Texts
  - Bruce Eckel, Thinking in Java (2nd Edition), Prentice Hall, 2000
    - A complete copy of the book can be downloaded for free
  - Doug Lea, Concurrent Programming in Java (2nd Edition), Addison Wesley, 2000
- Secondary Texts
  - Java API
    - http://java.sun.com/j2se/1.4/docs/api/index.html
## Tentative Grading Plan

<table>
<thead>
<tr>
<th></th>
<th>#</th>
<th>% each</th>
<th>% total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming projects</td>
<td>6</td>
<td>8.33</td>
<td>50</td>
</tr>
<tr>
<td>Mid-terms</td>
<td>2</td>
<td>15</td>
<td>30</td>
</tr>
<tr>
<td>Final</td>
<td>1</td>
<td>20</td>
<td>20</td>
</tr>
</tbody>
</table>