ECE 462
Object-Oriented Programming using C++ and Java

Course Organization and Grading

Yung-Hsiang Lu
yunglu@purdue.edu
Course Organization

Please feel free to ask questions at any moment.
Prerequisites

- ECE 264. If you have not taken ECE264, please talk to the instructor.
- Know how to write and compile C programs in UNIX-based (e.g. Linux or Solaris) machines, such as gcc, gdb, and Makefile
- Understand the concept of pointers in C
- We will not emphasize syntax. Instead, we will spend more time on how to design and implement non-trivial programs.
- Expect to read a lot of code!
Textbook

- "Programming with Objects" by Kak, John-Wiley
- many executable examples about the concepts explained in the book
- contains topics that are rarely discussed in other books, such as multiple inheritance
- provides frequent comparison between C++ and Java
Experiment:
Directed Problem Solving in Labs

- This semester, we continue a pedagogical experiment, similar to DPS in ECE270 and 362.
- All lectures are recorded and available on-line.
- Every week = 2 lectures + 1 lab
- There is no reduction of the course material because QA is handled outside lectures and no time is wasted setting up computer or demonstration of tools.
- Lab sessions = office hours. Additional office hours are also available by appointments.
- You can watch the lecture videos at any time. You are encouraged to watch the videos with classmates, pause, and discuss.
- Starting from 08/27, we meet in MSEE 190 (and MSEE 189 if MSEE 190 if full). Bring your earphone.
Students' Comments in 2007

• I enjoyed this course a lot. I thought the projects were extremely helpful to my learning.
• Good content, enjoyed learning OO.
• Good Job!
• I thought Professor Lu did a great job with this course and I can tell that he cares a lot about helping students learn.
• The professor is very good and dedicated to helping the students learn the material.
Goal: Enjoy the course!
Advantages of DPS

• **Learn at your own pace.** You can watch each lecture once or multiple times (*asynchronous* learning).
• Encourage group study by watching the videos together.
• Promote self assessment, using the time you need.
• Provide more flexibility in utilizing your time.
• Enhance student-instructor interactions in the lab.
• Accommodate the wide range of students’ background.
• Offer one-to-one attention to individual’s learning needs during the lab hours.
Disadvantage of DPS

• no classroom interaction with classmates ⇒ watch lecture videos with classmates
• no classroom interaction with instructor ⇒ use lab hours
• cannot ask questions during lectures ⇒ post questions in Blackboard
• Please check Blackboard discussion often.
• Do not procrastinate.
• Please provide frequent feedback and suggestions through Blackboard discussion.
Improvements from 2007

• We continue the successful experiment in 2007.
• Lectures, programming assignments, and lab exercises are recorded in separate videos.
• On-line homework in Blackboard is eliminated. Self tests are embedded in the videos.
• The lab exercises are more closely integrated with the programming assignments.
• Video is provided in Adobe Flash, supported by most web browsers.
• Bonus points are given to more categories.
• One more midterm exam so that fewer questions are asked in each exam. You have more time to answer the questions.
Teaching Staff

- Instructor: Yung-Hsiang Lu, MSEE 222, yunglu@purdue.edu
- Teaching Assistant: Guangwei Zhu, guangwei@purdue.edu
- Office Hours: MWF 0930-1020AM MSEE 190, or by appointments
- Please post all questions related to the course to Blackboard Discussion. Use email for only person issues (grades, appointments, early exam …).
- Check Blackboard before asking. Maybe the answer has been posted.
http://www.itap.purdue.edu/tlt/blackboard/

Everything is in Blackboard. Please check it often.
Object-Oriented Software Design

Course: ECE462 Object-Oriented Programming using C++ and Java

Instructor: Yung-Hsiang Lu

Textbook: Programming with Objects by Professor Avinash Kak at Purdue University.

Tools:

- NetBeans
- Eclipse
- Javassist
- DDD

Fall Semester 2007
Fall Semester 2008
ECE 462 Object-Oriented Programming using C++ and Java

Fall 2008

If you are taking this class, please enter Blackboard to submit your assignments, check your grades, and join the discussion.

[outcomes | lectures | assignments | lab exercises | exams | textbook | changes in 2008]

Your Feedback

Course Outcomes:

A student who successfully fulfills the course requirements will have demonstrated:

1. an ability to write object-oriented programs of moderate complexity in Java. [3,4,e]
2. an ability to write object-oriented programs of moderate complexity in C++. [3,4,e]
3. an understanding of the concepts of inheritance and polymorphism. [3,4,k]
4. an ability to use template classes and the STL library in C++. [3,4,k]
5. an ability to overload operators in C++. [4,c,e,k]
6. an ability to incorporate exception handling in object-oriented programs. [3,4,k]
7. an understanding of the difference between function overloading and function overriding. [3,4,k]
8. an ability to write programs with multiple threads and use synchronization among threads. [3,4,k]

Suggestions to Students:

Learning by watching on-line videos (also called "asynchronous learning") has unique challenges. The lectures may move faster than your own pace, because you cannot stop the instructor and ask questions. Meanwhile, there are many advantages, for example, you
Laboratory Exercises and Due Dates:

1. 2008/09/03, (Java) Netbeans programming interface, version control, drawing and handling events. slides, video (14:02), code.
2. 2008/09/10, (Java) Tracking mouse movement and bouncing ball. slides, video (8:59), code.
3. 2008/09/17, (C++) Qt graphical user interface: drawing and handling events. slides, video (15:57), code.
5. 2008/10/08, (Java and C++) Container classes' performance. slides, video (1:24), code.
7. 2008/10/22, (Java and C++) Multi-thread, convert C++ to Java. lab07.cpp.
8. 2008/10/29, (Java and C++) Multi-thread, convert Java to C++. lab08.java.
10. 2008/11/12, (Java or C++) Test coverage.
11. 2008/12/03, (Java) Java3D graphics.
12. 2008/12/10, (C++) Parallelization challenge.

Exam Dates:

1. 2008/09/26, Exam 1 (in class)
2. 2008/10/24, Exam 2 (in class)
3. 2008/11/21, Exam 3 (in class)
4. TBD, Final exam

All exams are open-book and open-note. Please bring your Purdue ID. No electronic device (including calculators) is allowed. Exams from 2007.
Course Outcomes

1. write object-oriented programs of moderate complexity in Java.
2. write object-oriented programs of moderate complexity in C++.
3. understanding of the concepts of inheritance and polymorphism.
4. use template classes and the STL library in C++.
5. overload operators in C++.
6. incorporate exception handling in object-oriented programs.
7. understanding of the difference between function overloading and function overriding.
8. write programs with multiple threads and use synchronization among threads.

Passing all outcomes is a necessary (but not sufficient) condition to receive a passing grade (A - D).
Check Blackboard for announcements.
Grading
Grading

• 10 lab exercises, 1 point each
• 5 programming assignments, 6 points each
• 4 midterm exams (09/24, 10/20, 11/10, 12/03), 10 points each
• 1 final exam, 15 points
• 5 points for class participation (next slide)
• All exams are open-book, open-note. No collaboration. No electronic devices (including phones)

Bring a photo ID for all exams.

• An exam can be taken earlier. Please talk with the instructor.
• A: 85 points or higher, B: 75 - 84.9, C: 65 - 74.9 … after normalization by the highest score in class (if < 100).
• F: fail any outcome. Each outcome is tested twice. Cheating.
Class Participation

• Each person can earn 5 points + up to 10 bonus points: You can receive 0.5 point for each of the following items
  – attend a lab hour
  – provide 3 or more questions for an exam (posted in Blackboard at least 12 hours before the exam)
  – post a "meaningful" message in Blackboard (not "Yes, I agree.")
  – post a "meaningful" content in Kiwi
  – fill a survey or the course evaluation
  – submit a programming assignment on time (09/19, 10/10, 10/31, 11/14)
  – attend "Education and Technology" discussion on 11/17
  – visit the instructor's office
Grade Calculation

\[ \sum \text{lab exercises} \quad + \quad \frac{10}{10} \]
\[ \sum \text{programming assignments} \quad \times 6 + \quad \frac{100}{100} \]
\[ \sum \text{midterm exams} \quad \times 10 + \quad \frac{100}{100} \]
\[ \text{final exam} \quad \times 15 + \quad \frac{100}{100} \]
\[ \text{class participation} \]

full score = 110, including 10 bonus points
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Type</th>
<th>Slides</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>09/05</td>
<td>17-17.7</td>
<td>Lab (MSEE 190)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>09/08</td>
<td></td>
<td>Slides</td>
<td>17.8-17.12</td>
<td>Code</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>User Interface (14:22)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Qt Signals and Slots (17:45)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Java Events and Handler (9:47)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Array (11:27)</td>
</tr>
<tr>
<td>09/10</td>
<td>LE02</td>
<td>Slides</td>
<td>5-5.2</td>
<td>Code</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Collision Detection (16:57)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>New Container in C++ (12:24)</td>
</tr>
<tr>
<td>09/12</td>
<td></td>
<td>Slides</td>
<td>17.13-17.17</td>
<td>Lab (MSEE 190)</td>
</tr>
<tr>
<td>09/15</td>
<td></td>
<td>Slides</td>
<td>6-6.9</td>
<td>Code</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Open and Save File (21:36)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Definition, Declaration, and Initialization (17:23)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lab Exercise 3 (15:57)</td>
</tr>
<tr>
<td>09/17</td>
<td>LE03</td>
<td>Slides</td>
<td>7-8.5</td>
<td>Code</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Reference (4:38)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Parameter Passing (8:23)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Copy Constructor (14:41)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Parameter Passing 2 (6:00)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Overload Resolution (15:01)</td>
</tr>
<tr>
<td>09/19</td>
<td>PA1 Java Breakout</td>
<td></td>
<td>9-9.13</td>
<td>Lab (MSEE 190)</td>
</tr>
<tr>
<td>09/22</td>
<td></td>
<td></td>
<td></td>
<td>Li. review at ARMS 1109 (bring your questions, no lecture)</td>
</tr>
<tr>
<td>09/24</td>
<td>Exam 1 (outcome 3,4)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>09/26</td>
<td></td>
<td>Slides</td>
<td>12-12.11</td>
<td>Code</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Operator Overloading (24:10)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Small Int and Conversion (24:30)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Lab Exercise 4 (1:24)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Programming Assignment 03 (4:47)</td>
</tr>
</tbody>
</table>
"Education and Technology"

• Throughout this semester, you will be requested to provide feedback on a study about "Education and Technology."
  – Does information technology improve learning?
  – Does technology help? Does it make people "lazy"?
  – Your opinions about "distant learning"
  – Do interactions with the instructor in the lab help?
  – ...

• On 11/17, group discussions by three researchers.
• More details will be provided later.
• Sponsors: Purdue TLT, NSF, Intel.
Exams
(open book, open note, individual)

- multiple choices, short answers, short code segments
- “zero-tolerance” of dishonesty: violations will be reported to the associate head of ECE.
- Cheaters (exams or assignments) will receive F.
- Regrading must be submitted by a written request (or email) within one week after the grade is posted. You are not allowed to ask or discuss with the TA.
- You can arrange to take an exam early.
- If you have to take an exam late, you must have a police report showing a true emergency.
Collaboration and Submission

• You can discuss lecture, homework, lab, or programming assignments with anyone. You can share code with only your programming partner (if you have one).

• You can have one and at most one partner for each programming assignment. All other coursework must be done by yourself only. If you discuss with anyone, please document it in your submission.

• All submissions must use Blackboard.

• All submissions will be graded in MSEE 190. Please ensure that your program can run on these computers.

• Code similarity will be checked.
Email submission will not be accepted.

Step-by-step instructions for submission in Blackboard are shown in later slides.

The instructor will not go to your house for grading. (Thank you for the invitation.)
Course Account and Tools

- Each student receives a class account "ee462xxx" in Blackboard.
- Please change the password by
  
  > ssh ee462xxx@shay.ecn.purdue.edu
  > passwd

- Use the account in MSEE 190 Lab (or any ECN Linux).
- Many tools (or newer versions) are available for your ee462xxx accounts. Do not use your personal Purdue account.
- Type "more /home/shay/a/sfwtools/public/README"
- Put "source /home/shay/a/sfwtools/public/settings" in your ~/.cshrc (if you use csh or tcsh)
- The videos are recorded in Windows but the programs can run on both Linux and Windows.
Programming Assignments

• 4 regular programming assignments:
  – Java 09/19
  – C++ 10/10
  – C++ ⇒ Java 10/31
  – Java ⇒ C++ 11/14

• You can do each assignment alone or work with one (only one) classmate. You **may change** the group mate for each assignment.

• You can discuss programming assignments with anyone but you are allowed to **share code only with your partner**.

• If you work in a team, **both students must submit** the same files + GROUP.
12/05 Programming Assignment 5

• You (and your group mate) decide what to do. You can choose Java or C++ or both.

• Requirements:
  – object-oriented
  – an interactive game + graphical user interfaces
  – networking (required)
Questions?
Submission Procedure
You should have ECE462 in your Blackboard already.
Log in to Blackboard Learning System - Mozilla Firefox

Purdue University
West Lafayette Academic Campus

West Lafayette Academic Campus Log In

Browser Check

Before logging in, it is highly recommended that you perform a browser check to see if your computer is properly configured to use the Blackboard Learning System.

Check Browser

Pop-up Blockers

The Blackboard Learning System sometimes uses small browser windows to deliver requested content. Some pop-up blockers incorrectly interpret these small windows as pop-ups and block them. If you have installed a pop-up blocker, turn it off while using the Learning System.

Java Security Certificate

After logging in to the Blackboard Learning System, open your browser's Java Security Manager and allow the Blackboard Learning System to install the required Java security certificate.
Notice the due date.
Confirmation

- Demo Student:
  You have submitted your assignment (Demo Student).
  Submitted Time: February 14, 2008 10:02 PM

OK
How to Resubmit