CMSC 132: OBJECT-ORIENTED PROGRAMMING II

Course Introduction

Department of Computer Science
University of Maryland, College Park
Course Catalog Description

- Introduction to use of computers to solve problems using software engineering principles
- Design, build, test, and debug medium-size software systems. Learn to use relevant tools
- Use object-oriented methods to create effective and efficient problem solutions
- Use and implement application programming interfaces (APIs)
- Programming done in Java
Things You Will Learn

• Object-oriented software development
  • Modern software development techniques
  • Object-oriented design

• Algorithms & data structures
  • Lists, trees, graphs

• Programming skills
  • Java API, IDE, testing, debugging
Course Is Not Just About Java

• May seem to focus on Java
  • All programming in Java
  • Many interesting Java language features

• Lessons intended to be general
  • Principles should apply to all languages
    • Ways of thinking about design
    • General ideas about software
  • Can translate skills to other languages
Assume You Already Know

- Coding
  - Variables, operators, loops, arrays
- Basic object-oriented programming
  - Classes, methods, inheritance
- Java
  - Class libraries, exceptions
- Tools
  - Eclipse IDE, debugger
Organization

• Class Web Page
  • http://www.cs.umd.edu/class/summer2013/cmsc132/
  • Notice that the submit server link is not the same one used in the spring. The link to use is:
    • https://submit.cs.umd.edu/summer2013

Personnel

• Coordinator
  • Nelson Padua-Perez
    • http://www.cs.umd.edu/~nelson

• Class Components
  • Lectures
  • Labs
Projects

• Around 8 projects
  • Evaluate design, coding, testing skills
  • Tries to involve interesting application areas
    • Networking, user interfaces, data compression

• Late policy
  • Projects due at 6 pm (notice this might not be the same time as in cmsc131 last semester)
  • 20% penalty, up to 9am the next morning
  • Plan to complete all projects on time
Projects (cont.)

- Environment
  - Eclipse IDE
  - Do not use your cmsc131 workspace (Create a new one)
    - In Eclipse select “File”→”Switch Workspace”
  - We are using a new repository (information will be provided later on)
  - Regarding Java version to use
- Automated submission & testing
  - Submit server
    - [https://submit.cs.umd.edu](https://submit.cs.umd.edu)
  - Maintains record of submissions
    - CVS repository
  - Release testing
    - Can evaluate project using real test cases
Grading

- Based on
  - Projects, quizzes/lab exercises, midterms, final
- Point distribution (roughly)
  - 40% Projects (8)
  - 12% Quizzes/Lab Exercises (some pop quizzes)
  - 28% Midterms (2)
  - 20% Final Exam
- Available on-line
  - https://grades.cs.umd.edu
Terpconnect Account

- We are going to use the grace cluster (linux.grace.umd.edu) for the CVS repository. For now, the only thing you need to do is to make sure you have a terpconnect account (we will provide information on how to set the CVS repository later on). If you don't have a terpconnect account, you can request one at:
  http://www.oit.umd.edu/new/
- Your TerpConnect account will be available within 1-3 days, that means you should activate it today. You should see the message TerpConnect Activated once your account is ready
Academic Honesty

- All individual assignments & exams must be done individually (except "open" assignments)
- Do not copy (or allow others to copy) your work in any way
- Submissions will be compared to submissions from current and previous semesters
- Cases of academic dishonesty will be referred to the University's Office of Judicial Programs
- Visit Student Honor Council website for more detailed explanation of academic dishonesty
Excused Absences/Academic Accommodations

• Excused absence does not typically translate into project extensions
• Students requesting reasonable academic accommodations due to a disability must provide a letter from the Office of Disability Support Services
• Please see the syllabus for additional information
Course Advice

• Read the syllabus
• Start projects **early**
  • Make use of release testing if offered
• Ask questions
• Read book
• Attend lectures
• Attend labs
• Attend office hours
• Pay attention to re-grade deadlines
Miscellaneous

- Regarding deadline to address grading concerns
  - It will be strictly enforced
  - At the end of the semester we will not address grading concerns for assignments/material already graded

- Regarding Email
- Regarding Electronic Devices
Miscellaneous

• If you are experiencing any problems that affect your performance in this class, please contact us immediately. Usually students wait until the end of the semester when probably nothing could be done.

• If for some reason you are considering dropping this course, see us first before making this decision.

• Do you have difficulties taking exams? Check:
  • http://www.cs.umd.edu/~nelson/classes/utilities/TakingExams.html

• Work hard from the beginning of the semester in order to avoid the following type of messages at the end of the semester:
  • Is there any extra credit so I can boost my grade?
  • I am .1 from an A; can something be done?
  • I need to pass this class otherwise I will …
Miscellaneous

• Some links of interest
  • What is Computer Science?
  • Information Sources
    • [http://slashdot.org/](http://slashdot.org/)
  • Google alerts