1/19/2015

CSCE 315: PROGRAMMING STUDIO

Spring 2015

Lecture	MW 4:10pm – 5:25pm		HECC 203
Labs	MW 1:50pm – 2:40pm	Sec. 504	RDMC 111C (Yin Qu)
	TR 8:00am – 8:50am	Sec. 505	RDMC 111C (Di Xiao)
	TR 2:20pm – 3:10pm	Sec. 506	RDMC 111C (Di Xiao)

Course Description Intensive programming experience that integrates core concepts in Computer Science and familiarizes students with a variety of programming/development tools and techniques; students work on 2 or 3 month-long projects each emphasizing a different specialization within Computer Science; focuses on programming techniques to ease code integration, reusability, and clarity.
Prerequisites: CSCE 312 and 314; or CSCE 350.
Co-requisite: CSCE 313.

Instructor Philip C. Ritchey
E-mail ritchey@cse.tamu.edu
Telephone 979-862-6476
Office HRBB 326

Office Hours TWR 11:10am – noon, and by appointment

TAs Di Xiao xiaodi_sean@tamu.edu
Yin Qu quyin@tamu.edu

Course Page http://faculty.cse.tamu.edu/ritchey/courses/csce315/spring15

Textbook *Code Complete*, 2nd ed., Steve McConnell, Microsoft Press, 2004

Goals This course is intended as an intensive programming experience that integrates core concepts in Computer Science and familiarizes students with a variety of programming/development tools and techniques. Students will primarily work in small teams on month-long projects emphasizing different specializations within computer science. The course focuses on honing good programming techniques to ease code integration, reuse, and clarity.

The primary goal for this class is to have students emerge with strong programming skills, able to address both individual and team programming challenges competently. The class is meant to allow students to improve their programming skills through significant practice.

Objectives The expected accomplishments of the students are as follows:

1. Become a confident software developer experienced in the full software development cycle.
2. Become a capable and effective member in a small software development team.
3. Become an effective communicator within the context of software projects.

Outcomes The students who take this course should be able to demonstrate the following upon the completion of this course.

1. Knowledge of programming and debugging tools.
2. Knowledge of various programming paradigms.
3. Ability to design and refine large software systems based on rough system requirements.
4. Ability to implement and test software system design.
5. Ability to work as a member of a software project development team.
6. Knowledge of various software development paradigms.
7. Ability to manage software development projects.
8. Ability to write technical documentation regarding software systems.
9. Ability to communicate the overall design and details of software systems.
10. Introductory-level knowledge in database systems, artificial intelligence, and software engineering.

Grading The contribution of each element to the course grade is

Individual Project	*15%*
Team Project 1	*25%*
Team Project 2	*25%*
Team Project 3	*25%*
Quizzes	*10%*

Individual grades for team projects will be adjusted based on contributions to the team.

There is no final exam but the time is reserved for project demonstrations. You must keep track of your own grades from the papers handed back. Final grades will be assigned according to this scale:
A: 90 and above, B: 80-89, C: 70-79, D: 60-69, F: below 60
The scale may be adjusted by the Instructor to reflect score variations.

Late Policy Late submissions will be charged a penalty of 1% per hour.

Attendance Lecture and lab attendance are expected. Infrequent unavoidable absences are understood, but each student is responsible for any missed material. For excused absences, students will not be penalized. See Section 7 of the Student Rules for the excused absence policy. For acute illnesses of less than three days, both option A and option B of section 7.1.6.2 are acceptable in this course. For unexcused absences, a grade of zero will be assigned for missed work.

Final Exam 3:30pm – 5:30pm Friday May 8th (reserved for project demonstrations)

Copyright The handouts used in this course are copyrighted. By "handouts," I mean all materials generated for this class, which include but are not limited to syllabi, quizzes, exams, lab problems, in-class materials, review sheets, and additional problem sets. Because these materials are copyrighted, you do not have the right to copy the handouts, unless I expressly grant permission.

Plagiarism As commonly defined, plagiarism consists of passing off as one's own the ideas, words, writings, etc., which belong to another. In accordance with this definition, you are committing plagiarism if you copy the work of another person and turn it in as your own, even if you have the permission of that person. Plagiarism is one of the worst academic sins, as the plagiarist destroys the trust among colleagues without which research cannot be safely communicated.

If you have any questions regarding plagiarism, please consult the latest issue of the Texas A&M University Student Rules, under the section "Academic Misconduct."

ADA The following ADA Policy Statement (part of the Policy on Individual Disabling Conditions) was submitted to the University Curriculum Committee by the Department of Student Life. The policy statement was forwarded to the Faculty Senate for information.

The Americans with Disabilities Act (ADA) is a federal anti-discrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires that all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. If you believe you have a disability requiring an accommodation, please contact the Department of Student Life, Services for Students with Disabilities, in Cain Hall or call 979-845-1637.

Academic Integrity

AGGIE HONOR CODE
"An Aggie does not lie, cheat, or steal or tolerate those who do."

Upon accepting admission to Texas A&M University, a student immediately assumes a commitment to uphold the Honor Code, to accept responsibility for learning, and to follow the philosophy and rules of the Honor System. Students will be required to state their commitment on examinations, research papers, and other academic work. Ignorance of the rules does not exclude any member of the TAMU community from the requirements or the processes of the Honor System. For additional information please visit: www.tamu.edu/aggiehonor/

You are expected to do your own work, and to do your fair share of the team projects. ALL RESOURCES USED MUST BE FULLY DOCUMENTED. IF IN DOUBT, DON'T USE IT! Any paper or action which violates the Aggie Honor Code will result in a course grade of F* (* = for academic dishonesty) if this is the first offense, plus additional penalties as determined by the Aggie Honor System Office (see http://www.tamu.edu/aggiehonor) if this is not the first offense.

You are responsible for complying with all provisions of the Student Rules posted at http://student-rules.tamu.edu/.

Tentative Schedule

Due to the project-intensive nature of the course, about 1/3 of the ***lecture*** times will be cancelled (to be announced in advance) in order for students to use that time for team meetings.

Week*	Topic*
1/19 – 1/23	Naming, Style, and Commenting in Code
1/26 – 1/30	Agile Development
2/2 – 2/6	Database Systems and Applications
2/9 – 2/13	Code Reviews and High Quality Code
2/16 – 2/20	Unit Testing and Test-Driven Development
2/23 – 2/27	Work on Team Project
3/2 – 3/6	Artificial Intelligence
3/9 – 3/13	Design Patterns
3/23 – 3/27	Work on Team Project
3/30 – 4/3	Web Programming and Information Security
4/6 – 4/10	"Waterfall" Development
4/13 – 4/17	APIs
4/20 – 4/24	Work on Team Project
4/27 – 5/1	Work on Team Project
5/4 – 5/5	Work on Team Project

* *Dates and topics are approximate and subject to change.*