CSC341 Software Design – Prerequisites: CSC226

(Thursday 6-9:30pm)

Instructor: Dr. Chenglie Hu (chu@carrollu.edu)
Office: Charles House 201, Phone: 262-524-7170
Office Hours: 1:00-2:00 M, TH (walk-ins, emails, and appointments are welcome)

Course Learning Outcomes:
In light of the Undergraduate Program Goals, the course learning outcomes of CSC341 are:
Upon completion of the course, the student will:
1. Improve understanding of Java language constructs that are critical in object-oriented design
2. Understand and be able to apply design principles, guidelines, strategies, and practices
3. Write effective use cases, and be able to use UML in software analysis
4. Understand design patterns and their applications
5. Be able to make design decisions in light of design trade-offs
6. Be able to carry out a design process from requirement analysis to design implementation

Textbook:

Coverage: the entire book and materials from other sources

Teaching Methods:
1. This is a class where students gain problem-solving skills in terms of solving design problems. Students achieve good understanding of design principles and methodologies by practicing on them. Therefore, active participation in class discussions and other activities is vitally important. You are expected to preview the chapter to be covered and bring questions to the class.
2. Lectures are to use PowerPoint presentations assisted with programming examples. Each class may also include a lab/discussion portion to reinforce the lecture content.

Assessment:
1. Exercise questions/problems are regularly assigned and collected on a weekly basis. All assignments should be typed and only the hardcopies are accepted. If design implementation is required, screen captures of the program execution are expected. (Learning Outcomes 1-6)
2. Quizzes may be given at instructor’s discretion. (Learning Outcomes 1-6)
3. There are two exams: midterm and final (each covers respective portion of the course). Final Exam date: May 4th, 2017. (Learning Outcomes 1-6)
4. There will be one design project (individual or team-based) in addition to many small design assignments. (Learning Outcomes 1-6)
5. Class participation (in discussion and in other class activities) (Learning Outcomes 1-6)

Grading Policies:
Letter grades will be determined using a percentage point evaluation as outlined below.
A 90%-100%  AB 86%-89%  B 80%-85%  BC 76%-79%  C 70%-75%  D 60%-69%  F Below 60%
(The final percentage is a weighted average, and the instructor reserves the right to determine the weights later in the semester)
Course Policies:
Missed Classes: Missing more than one week of classes will have negative impact on the final grade at instructor’s discretion. It is expected that you inform the instructor of an absence in advance in order for you to receive information about the class you are going to miss in a timely fashion to minimize the impact on learning.
Assignments: Each assignment is normally due one week after the assignment is given. Late submission of assignments will not be accepted without consent of the instructor for the delay.

Other Policy Statements:
1. ACADEMIC HONESTY: While discussions are encouraged, all assignments must be completed independently. Copying other people’s work, if found, will result in zero credit for both parties involved. Repeated copying other people’s work will result in failure of the course, in addition to other university penalties that may apply. For more information refer to the "Academic Dishonesty" policy in VII of the School’s Student Handbook.

2. The instructor and the University reserve the right to modify, amend, or change the syllabus (course requirements, grading policy, etc.) as the curriculum and/or program require(s)

Help Resources:
(1) There are ample online resources for software analysis and design. Just enter a topic (such as “software design principles”) into Google search. Often times, it might be a quickest way to find an answer.
(2) Contact Walter Young Centre if you need accommodation due to disabilities or other health issues.

Tentative Schedule
Week 1 – The nature of software design, basic design principles
Week 2 – The foundation of object orientation
Week 3 – Design principles I, inheritance, Java graphics
Week 4 – Design principles II, methods design
Week 5 – Design principles III, class design
Week 6 – UML and use cases
Week 7 – Software design patterns I – creational design patterns
Week 8 – Software design patterns II – structural design patterns
Week 9 – Software design patterns III – behavioural design patterns
Week 11 – Software architecture (MVC and architectural frameworks)
Week 10 – Case study
Week 12 – Design studio I
Week 13 – Design studio II
Week 14 – Wrap up