CSC600 Software Design – prerequisite: CSC506
(Monday, 6pm-8:50pm, room L121 CGS)

Instructor: Dr. Chenglie Hu (chu@carrollu.edu)
Office: Charles House 201, Phone: 262-524-7170
Office Hours: Mondays (5:00-5:50 in the room 112, CGS), and M, TH (1:00pm-2:00pm) (walk-ins and appointments are welcome)

Course Learning Outcomes:
In light of the MSE Program Goals, upon completion of the course, the student will:
1. Improve understanding of Java language constructs that are critical in object-oriented design and be able to leverage his/her object-oriented programming ability
2. Be able to model software system with UML and other relevant methodologies
3. Understand and be able to apply design principles, guidelines, strategies, and practices in software analysis and design;
4. Understand design patterns and be able to apply them in solving design problems
5. Be able to make design decisions in light of design trade-offs

Textbook (optional):

Other References: There are ample online references about OO, design patterns, design principles, UML, and use cases that students are encouraged to search for, study and use.

Teaching Methods:
1. This is a class where students gain problem-solving skills in terms of solving software design problems. Students will only achieve good understanding of design principles and methodologies by practicing on them. Therefore, active participation in class discussions and design activities is paramount.
2. Lectures use PowerPoint presentations assisted with programming examples. A typical class may also include a lab/discussion portion to reinforce what the lecture covers.

Assessment:
1. Homework exercises are regularly assigned and the due date for each assignment is normally the next class meeting time unless specifically instructed. All assignments should be typed and only hardcopies are accepted. For assignments that require code implementation, hard copies of the source code and screen captures of the execution are also expected. (Learning Outcomes 1-5)
2. There are two exams: midterm and final with each covering respective portion of the course. The format of an exam can be either an in-class written exam or a take-home exam. (Final exam date: May 8th, 2017) (Learning Outcomes 1-5)
3. Class participation is important and also part of the assessment. (Learning Outcomes 1-5)

Note: The following pages should be stapled together in that order when you assemble an assignment:
(1) The assignment handout
(2) Describe whether or not each item of the assignment was completed successfully. If not, describe the reason and the difficulties you had. You may also ask questions related to the assignment, and will see written answers when you get your assignment back.
(3) The completed assignment artefacts/deliverables. If implementation of a design is also required, then include the code and a screen capture that shows outputs of the test cases used.
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%

Tentative weights of a grade: assignments: 50%, Exams: 45%, class participation: 5%
(Class participation includes class attendance and active participation in class activities

Course Policies:
Missed Classes: Contact the instructor in advance if you are not able to attend the class so that you receive advice in a timely fashion to minimize the impact on learning. Missing more than one week of classes will have negative impact on the final grade at the instructor’s discretion.
Assignments: Late submission of an assignment will not be accepted without consent of the instructor in advance.

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 Student Handbook of the University.
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)
3. Contact Walter Young Centre if you need accommodation due to disabilities or other health issues.

Tentative Schedule
Week 1 – The nature of software design, GUI programming
Week 2 – The foundation of object orientation
Week 3 – Design principles I, inheritance
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 design frameworks)
Week 10 – Case study
Week 12 – Design studio I
Week 13 – Design studio II
Week 14 – Wrap up