

## **Com S 362: Object-Oriented Analysis and Design**

Hridesh Rajan

MWF, 11:00 AM – 11:50 AM, Pearson 1115

### **Course Objectives**

In this class, we will ask several questions and try to find answers together. The course objectives are open to your suggestions. If you feel that there is something missing here please let the instructor know.

- Why design? Why appreciate software design as a process? Is it critical?
- What were the design approaches before object-oriented design? What is object-oriented design? What are the design approaches beyond object-oriented design?
- What is a good design? How do you differentiate between a good and a bad design? What are some important characteristics of a good design?
- How to design? Are there recipes to common design problems?
- Etc.

This course is:

NOT a course on object-oriented programming, you are expected to have an intermediate knowledge of an object-oriented programming language such as C# or JAVA. If you do not, please refer to Ken Arnold and James Gosling and David Holmes. The Java Programming Language Fourth Edition. Addison-Wesley, Reading, Mass.

NOT a course on software processes, however, we will learn just enough about them to use them in course projects.

NOT a course on data structures/algorithms. CompS 228 is a pre-requisite for this course

### **Logistics**

Instructor: Hridesh Rajan (hridesh@cs.iastate.edu)

Meeting Times: MWF 11:00 AM - 11:50 AM in Pearson 1115

Office Hours: W 11:50 AM - 12:30 PM in 211 Nucl. Eng. Laboratory Building or by appointment

Course website: <http://www.cs.iastate.edu/~hridesh/ComS362>

### **Required Text Books**

Craig Larman. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process, Third Edition. Prentice Hall PTR, Upper Saddle River, NJ, 2004, ISBN 0131489062.

Ken Arnold and James Gosling and David Holmes. The Java Programming Language Fourth Edition. Addison-Wesley, Reading, Mass., 2005, ISBN 0321349806.

## **Recommended Text Books**

Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Boston, MA, 1995. ISBN 0201633612.

Martin Fowler. UML Distilled: A Brief Guide to the Standard Object Modeling Language, Third Edition., Addison-Wesley, Boston, MA, 2003, ISBN 0321193687

## **Prerequisites**

COMS 228 and ENGL 104

## **Disability Statement**

Please address any special needs or special accommodations with me at the beginning of the semester or as soon as you become aware of your needs. Those seeking accommodations based on disabilities should obtain a Student Academic Accommodation Request (SAAR) form from the Disability Resources (DR) office (515-294-6624). DR is located on the main floor of the Student Services Building, Room 1076.

## **Course Components**

This course has the following components:

Homeworks: 20

Class participation: 15

Exam 1: 20

Project: 25

Exam 2: 20

Extra Credit: variable

All components are essential; you will not receive a passing grade in this course if you haven't completed a component of the course. For example, let us assume John Doe didn't hand in any homeworks, but received a 'B' grade or better in all other components. His final grade will be 'F' without any exceptions. The class participation grade will be determined by your attendance in the class and your contributions to the class discussions. You will only be able to contribute to the class in a significant way, if you have carefully read the assigned readings for the class.

## **Grade Computation Logic**

You will receive an absolute letter grade for this course. There will not be any curving. As a result, everybody in this class may expect to receive an 'A'. The grades will be assigned as follows:

A: 95 and above

A-: 90 - 94

B+: 85 - 89

B: 80 - 84

B-: 75 - 79

C+: 70 - 74

C: 65 - 69

F: 64 and below

## **Late Homework Policy**

Late homework is not encouraged in this course. You are expected to submit your deliverables on time. Some homework solutions will be discussed in class after the due date. These homework

will not be accepted after the due date and you will automatically receive a zero grade for them. The penalties for other homework submission are as follows:

Upto 1 day late: 5%  
Upto 2 days late: 12.5%  
Upto 3 days late: 20%  
Upto 4 days late: 30%  
Upto 5 days late: 50%  
Upto 6 days late: 90%  
7 days or more: 100%

## **Programming Assignments**

**Test-Case Policy :** This policy applies to the programming assignment part of a homework. If we have provided test cases with the assignment, please ensure that your program passes all these test-cases before you turn in your homework. You will receive no grades for the failing program.

**Homework Packaging and Naming Convention Policy:** If we have provided a naming conventions for your files, classes, fields, methods, etc. and/or instructions on how to package your homework, please carefully follow them. Failure to do so may result in receiving no grades for that part of the homework.

**Teamwork Policies:** Software design and development is a team activity. Most organizations have teams of varying sizes. In this class, we will practice software development as it is done in the real world. We will simulate attritions, recruitment, and other such real world events that impact software development projects.

**Team Formation:** At the beginning of the class, 3-4 students teams will be formed. Each team will be assigned a name and a task to be completed in the rest of the semester.

**Attrition and Recruitment:** During the semester, 2 members will be removed from a team and 2 new members will be added from other teams. To survive this attrition and recruitment, you must maintain your project documentation in good shape at all time.

**Conflicts:** Please try to resolve conflicts within the team by a conversation among team members. If the conflict is still not resolved, please bring it to the attention of the instructor.

**Contributions:** The teamwork process is up to the individual teams. However, each student in the team is responsible for understanding the work in the team, thus teams should be very careful to have the whole team check and explain parts of the work if it is split up. You, as a team member, should be sure that you understand all that goes on and that you take on a fair share of the work.

**Certifying Contributions:** To help make sure work contributions are rewarded fairly, we have the team rate the work contributions of its team members at the end of each homework. Individual contributions are rated on a scale of 1-5, with 5 being highest, and 1 lowest. These are used to adjust the grades of team members above and below the team's grade. A rating  $X$  produces an increment to an individual's grade of  $(X - 3) * 2.5\%$ , which is added to the team's grade recorded for the individual (as long as the sum is between 0 and 100%).

For example, suppose Alice is in a group whose homework got 80%, and her individual rating was 5; then Alice's grade is recorded as 85%. In the same group if Bob has an individual rating of 2, then his grade is recorded as 77.5%.

The team is constrained to make the total individual work ratings average to 3, and all members must agree on the contributions of all other members. The certification of individual contributions

is to be done using a standard form, which is available as a MS Word document and as a HTML page and as a text file from the course webpage.

### **Thou Shall Not Cheat**

On confirmation of the evidence of cheating, all involved parties will receive a zero on that part. In addition, the final grade of all involved students will be reduced by one letter grade.

What constitutes cheating?

The simple rule of thumb is: Never give finished answers to someone else (or another team) or use someone else's finished answers. Such exchanges are definitely cheating and not cooperation. However, note that on team homework, there can be no cheating within a team.

### **Discussions, Cooperation, and Collaboration**

You are encouraged to discuss homework, and other parts of the class with other students. Certainly discussions within a team about team homework are fine and no harm comes from them. Even between teams, such discussions about ideas are not cheating, whereas the exchange of finished, written answers is cheating. However, when you have more than casual discussions about homework with students in other teams, or about homework that is not team homework with students in your own team, then you must cite the other person as described below.

When you cooperate on solution ideas or collaborate on producing final answers with other teams, you must cite the other teams you worked with as follows. This must be done for each problem on which you cooperate or collaborate. (That is, if you work with someone on a problem, you don't need to work with them on the entire homework.)

If you discussed ideas jointly, but wrote up a final answer for the problem independently, then each team should include a note with that problem's solution such as "the following solution was developed jointly with Alyssia P. Hacker in team X," or "the following idea is due to Ben Bittwiddle in team Y." Each person's final answer receives a grade independently of the other's; there is no bonus or penalty for such a citation.

If your team jointly worked on a final answer for a problem with another team (or teams), you should only hand in one answer for all teams concerned, and certify the answer as usual for teamwork. If the certification is not true for everyone, then only those people for whom it is true may participate in the joint solution; the others should use the ideas and write up their own final answer for the problem, with a note as in the previous paragraph.

Note that substantial collaboration on solutions between teams that is not cited as described above is considered cheating. Such cheating will be dealt with as described above. It should be clear that you will learn less by such exchanges of finished answers.

Finally, if you use reference materials (other than the course texts) to solve a problem, you must give a citation. This includes material from the web. Not doing so is plagiarism (i.e., cheating). We take plagiarism quite seriously, so note this policy well.