Syllabus - COMP 170-001 / ISOM 370-001
Introduction to Object Oriented Programming (IOOP)
Dr. William L. Honig

Catalog Description
This course is an introduction to the computer science major, covering basic concepts using the C# (C-Sharp) object-oriented (OO) programming language. The course addresses the following questions: What is an algorithm? How does one write, debug, run (“execute”), and test an effective computer program? How does one convert an algorithm into a computer program? How does one judge a program? What does “object-oriented” mean?

Topics include: variables, data types, input/output, loops and repetition, choice, arrays, subprograms, classes/objects, OO principles, system design and structure.

This course is programming intensive. Lab sessions will be held during class periods.

Prerequisites
Students are required to have taken MATH 117: College Algebra as a prerequisite or to have been placed in MATH 118: Precalculus or higher. Alternatively, students can take any one of the following courses as a prerequisite or as a co-requisite: COMP 163: Discrete Structures, COMP 150: Introduction to Computing, MATH 118: Precalculus, MATH 131: Applied Calculus I, MATH 161: Calculus I.

No prior programming experience is required (but it is helpful). The logical mindset of mathematics is generally helpful in learning programming. See me if you have any questions or concerns about your preparation. If you have absolutely no programming experience and you are concerned you may wish to take COMP 150, COMP 125, etc. prior to this course.

NOTE: The Syllabus and course schedule are subject to change; changes will be announced in advance in class and posted on Sakai. A change history is at the end of this Syllabus.

Special Course Requirements
1. COMP 170 / ISOM 370 is a programming intensive course. Considerable time will be spent creating programs in the lab sessions and outside class. The largest portion of your grade will be determined by your success in writing, compiling, running, and testing these programs.
2. The course may include some programming assignments done with another person (“Pair Programming”) or small teams of 2 to 6 students (“Team Programming”).
3. The course uses Sakai to organize materials. You will generally submit your assignments using Sakai or get grading information returned in Sakai. Ask for help if you are not familiar with Sakai.

4. The course may use Adobe Connect for online sessions; check it before class starts at: https://admin.acrobat.com/common/help/en/support/meeting_test.htm

Course Material and Optional Textbook, All Free and Online
We will use online course material first developed by Dr. Andrew Harrington and Dr. George Thiruvathukal for COMP 170 in Spring, 2012 (see this URL: http://anh.cs.luc.edu/170/). The required (but free) current online course textbook is at http://introcs.cs.luc.edu/ .

There is an optional free online textbook, the so-called C# Yellow Book, developed by Rob Miles at the University of Hull, at: http://www.robmiles.com/s/CSharp-Book-2012.pdf.

The official documentation for C# is from Microsoft, available at http://msdn.microsoft.com/en-us/library/67ef8sbd.aspx. It is useful to look up specific terms, concepts, and other things you want to see.

Academic Honesty
Students are expected to have read the statement on academic integrity available at http://www.luc.edu/academics/catalog/undergrad/reg_academicintegrity.shtml. This policy applies to this course. The minimum penalty for academic dishonesty is a grade of F for that assignment. Multiple instances or a single severe instance on a major exam or assignment may result in a grade of F for the course. All cases of academic dishonesty will be reported to the department office and the relevant college office and will be placed in your school record.

Academic dishonesty includes, but is not limited to, working together on assignments that are not group assignments, copying or sharing assignments or exam information with other students except in group assignments, submitting as your own information from current or former students of this course, copying information from anywhere on the web and submitting it as your own work, and submitting anything as your own work which you have not personally created for this course. If you do wish to use materials that are not your own, please check with me ahead of time and cite your source clearly. When in doubt, ask first!

Course Objectives and Goals
Upon successful completion of the course, the student will be able to:
1. Write good C# programs of small to medium size – programs that are correct, high quality, and use correct and appropriate Object Oriented Programming techniques.
2. Reuse C# classes and Application Programming Interfaces (APIs) developed by others, especially the standard C# APIs.
3. Understand and recognize proper programming style and demonstrate making design decisions consistent with Object Oriented methodologies.
4. Be able to read, understand, and interpret C# programs written by others.

Course Grading
Your grade will consist of several components with relative weights as follows (I reserve the right to adjust the percentages in your favor if circumstances warrant). Also see the sections on Timely Completion and Academic Honesty. Note: these components are weighted, so you can't just add your points on Sakai and divide by the total points to get your grade!
1. Homework Assignments (weight: 20%). C# assignments throughout the course. You will write, compile, run, and test your program before turning it in. Programs that do not successfully compile without errors will generally receive 0 points.

2. Lab Assignments (15%). The lab assignments will be completed during the class sessions each week and will be reviewed by me and/or the course TA to receive credit.

3. Quizzes and Examinations (40%). Any quizzes plus examination scores. A pop quiz may be given in one of the class sessions at any time without prior notice; it may cover any topic in the course to date. This approach serves to encourage you to keep up-to-date on readings and lecture material. There will be no “make-up” quizzes or exams unless you have made prior arrangements with me to be excused from them.

4. Final Project (20%). There will be a final team programming project for the course that includes periodic progress reports and a final team class presentation during one of the class sessions in the final week of the class; details will be provided in class.

5. Participation (5%). Students are expected to review all online lecture material prior to classes, and attend all class sessions for the full time period. The participation grade will be based upon attendance and enthusiastic involvement and contributions to discussions in the class sessions, and contributions to the course discussion board (see below). Attendance will be taken for all class sessions.

All course work will be graded numerically and your letter grade will be determined from total points earned, weighted as above. Your total points will be converted to a letter grade using approximately the following percentage ranges. Calculations will generally round fractional percentages below .500 down to the full number less than that value.

- 93-100 = A
- 90-92 = A-
- 87-89 = B+
- 82-86 = B
- 79-81 = B-
- 74-78 = C+
- 68-73 = C
- 62-67 = C-
- 56-61 = D+
- 50-55 = D
- 54 and lower = F

If for any reason you do miss a class session, it is your responsibility to determine what you missed, locate any handouts, determine any changes in assignments, course plans, or schedules, etc. It is not my obligation to help you make up for missing class.

I will not always cover all information from the online course material and textbook in class; additional materials may be added and additional guidance will be given. Information and activities in class and labs that are not in the book will likely appear on exams and quizzes, and will be helpful for your assignments and programming projects.

Please do not ask for personal “extra credit” to improve your grade as this is neither practical in the course nor fair to your fellow students. I reserve the right to provide extra credit assignments for the entire class if appropriate. I will be happy to discuss your performance in the course with you at any time, including discussing your possible grade based on current performance plus ways to improve your performance during the remainder of the course. In fact, I encourage you to arrange time to talk with me outside of class if you have any questions regarding the topics we have covered in the course, homework, quizzes and exams, etc.

Course Schedule
The preliminary COMP 170 / ISOM 370 course schedule will be posted on Sakai in the same area as this Syllabus.
Timely Completion

You are expected to complete all assignments, readings, video reviews, and projects on time. In computer systems in the “real world” there is always a strong emphasis on getting projects done on time. Use this class to develop your own skills at timely completion.

Personal and any team programming projects and other assignments will be due as described at the time of the assignment. See the class schedule for advanced planning.

Late assignment submission is strongly discouraged.

1. Each student will be allowed up to TWO (2) late assignments of their choice. Think of this as having two “Late OK” passes. These passes may be used for lab sessions, programming, and non-programming assignments (not for quizzes or exams). Late Passes allow you to turn in the assignment up to 48 hours after the due date. Late passes are used by noting your intent to use one BEFORE the regular due date using Sakai. Email me and our TA completed assignment materials with subject line “COMP 170 Late Pass Assignment” and say that you have used a Late Pass for that assignment. The email must be sent no more than 48 hours after the original assignment due point.

2. No assignments will be accepted after the due date unless you are using one of your late passes and the assignment is turned in within 48 hours of the original date. Once your two Late Passes are used up no more assignments will be accepted after the due date. There are no exceptions except for documented emergencies.

3. Assignments are generally submitted in Sakai, at least during the first part of the course. Please plan ahead and be sure you complete the submission of the assignment on time. Also note that you can Save assignment materials as many times as you want, but once you click Submit you may not be able to change your submission. As we get into the course you will also use the Xamarin Studio / MonoDevelop Integrated Development Environment (IDE) and LUC Box to share your files with me and our TA; toward the end of the course you may also be using Bitbucket/Git software.

You are welcome to ask questions on all assignments and course work, seek additional information on the assignments, and offer observations on the assignments to me either in or outside of class. There will be a class discussion board for your use on a system called Piazza to discuss the course and assignments; information about joining Piazza will be provided. Effective help to others by answering their questions adds to your participation points for the class!

Please plan your work ahead and do not wait for the last minute to begin work! Try the homework before the day it is due! This step will be key to your success in class. No sympathy given for procrastination!

Office Hours and Help

See my web site or Sakai for current office hours. Extra time and special arrangements on request.

The Department of Computer Science provides tutors for this and other courses; you can find information about tutors on the CS website at http://luc.edu/cs/people/tutors/.
I will let you know before all our course exams what information will be covered in each and what material you can have access to during those exams.

Continuous Improvement

I believe in a personal quality process of continuous improvement. Anything can be improved by applying the quality process of “Plan, Do, Check, Act” (PDCA). To improve the course and the learning of programming, C#, and computer science concepts, I welcome your feedback, comments, suggestions, and complaints at any time.

In support of this PDCA process above, I may ask you to participate in surveys and discussions during the course. These surveys will measure student impressions of the course; when time permits I will share the results with the class. Your inputs on these surveys are anonymous and in no way affect your grade.

We will also be using an LUC feedback system called IDEA toward the end of the course to gather your feedback on both the course itself and my teaching; I use this kind of feedback to improve how I teach the information in the course, and the CS Department uses it to help refresh our course offerings.