Preparing People to Lead Extraordinary Lives

LOYOLA
UNIVERSITY
CHICAGO

Department of Computer Science
Water Tower Campus, Lewis Towers 520
820 North Michigan Avenue
Chicago, Illinois 60611-2196
LSC: Cudahy Science Hall 415 or 417

Syllabus - COMP 271
Data Structures
Dr. William L. Honig

Catalog Description

This course introduces key data structures such as lists, sets, and maps, as well as their implementations. Performance and analysis of algorithms are covered along with applications in sorting and searching.

This continuation of COMP 170: Introduction to Object-Oriented Programming introduces the concepts of data abstraction and data structure, including stacks, queues, lists, sets, and trees. The issues of implementing a data structure in a language such as Java are examined using classes, arrays, and linked structures. Sorting and searching techniques are analyzed. The concepts of correctness and efficiency of algorithms are developed. Time/space comparisons of iterative algorithms with recursive algorithms are made. The course includes several major programming projects.

This course is programming intensive. Additional time working on reading, preparing, and programming will be required outside of class time.

Prerequisites

COMP 170, Introduction to Object Oriented Programming.
COMP 163, Discrete Structures.

Students are expected to be able to read, write, and debug basic computer programs using standard tools including compilers and editors. Students are expected to know the basics of Object Oriented Programming and be able to use classes and methods they or others have written. The logical mindset of mathematics is generally helpful in learning programming. See me if you have any questions or concerns about your preparation.

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 271 is a programming intensive course. Considerable time will be spent creating programs in and outside class. The largest portion of your grade will be determined by

your success in writing, reading, compiling, running, and testing these programs and how you explain and discuss programming with the class.

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 online class starts at**: https://admin.acrobat.com/common/help/en/support/meeting_test.htm

Course Material and Textbook

Data Structures: Abstraction and Design Using Java. E.B.Koffman and A.T.Wolfgang. Wiley 2015 3rd edition. ISBN 978-1-119-18652-6.
The online version is recommended. You can try the online text free for 14 days at www.WileyStudentChoice.com

Additional course materials will be online, in Sakai, or provided in class.

The official documentation for Java version used in class is from Oracle https://docs.oracle.com/javase/8/docs/api/
It is useful to look up specific terms, concepts, and other things you want to see.

Oracle has a much material to help Java programmers: start from this link: http://www.oracle.com/technetwork/topics/newtojava/overview/index.html

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 or 0 points 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 you turn in your work.

Students with Disabilities and Other Accommodations

Students seeking academic accommodations for a disability must meet with Services for Students with Disabilities (SSWD) to verify the disability and to establish eligibility for accommodations. Students may visit SSWD in Sullivan Center - Suite 117, call 773-508-3700, email SSWD@luc.edu or visit LUC.edu/sswd to begin the process. Students should schedule an appointment to discuss any academic learning concerns and/or accommodations desired. Students are encouraged to contact SSWD as early in the semester as possible.

Course Objectives and Goals

Upon successful completion of the course, the student will be able to:

1. Discuss the appropriate use of built-in data structures. Describe common applications and construct and debug programs that use each of the following data structures: arrays, linked lists, stacks, queues, sets, and maps.
2. Describe the concept of recursion and give examples of its use. Implement, test, and debug simple recursive functions and procedures. Determine whether a recursive or iterative solution is most appropriate for a problem.
3. Explain why the creation of correct program components is important in the production of high-quality software. Describe how a contract can be used to specify the behavior of a program component. Apply a variety of strategies to the testing and debugging of simple programs.
4. Analyze the extent to which another programmer's code meets documentation and programming style standards.

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 Expectations and Academic Honesty. **Note: these components are <u>weighted</u>, so you can't just add your points on Sakai and divide by the total points to get your grade!**

1. Assignments (weight: 30%). Java 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 in the standard tools and versions used for class will receive 0 points**.
2. Quizzes and Examinations (25%). Any quizzes plus examination scores. A pop quiz <u>may</u> 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 class materials. There will be no "make-up" quizzes or exams unless you have an approved absence with documentation ahead of time.
3. Final Project (20%). There will be a final individual or 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.
4. Participation (25%). **Students are expected to read and study all text and other assigned 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 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

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. I encourage you to arrange time to talk with me if you have any questions, are considering dropping, or want to know how to do better in the remainder of the class.

Course Schedule

The preliminary COMP 271 course schedule will be posted on Sakai in the same area as this Syllabus. Changes may occur based on class needs; any changes will be announced in class and updates posted in Sakai.

Expectations and Timely Completion

You are expected to complete all assignments, readings, presentations, labs, and projects on time and with correct process for submitting them. You are required to be prepared to participate in class room and online discussions on all topics up to the current point in the class schedule.

You are required to be in all class sessions for the full class time. 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.

You need to follow these rules to be successful in the class:

Timely Completion: Late assignment submission is **generally not allowed**.
1. 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.
2. 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. See the Late Pass process for full details.
3. 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 no further assignments will be accepted after the due date. There are no exceptions except for documented emergencies with approval and consent of your college advisor or the Wellness Center.
4. 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!**

Correct Assignment Submission: use a clear quality process to submit your assignments correctly. **Errors in submitting your work are equally bad as errors in the work itself.** Programming is about precision and quality.

5. Assignments are generally submitted in Sakai. Please plan ahead and be sure you complete the submission of the assignment on time. Also note that you can <u>Save</u> assignment materials as many times as you want, but once you click Submit you may not be able to change your submission. After you save your final version come back and download it to be sure it is the right file complete with all work.

6. Files submitted in Sakai must use a strict naming convention except where noted in the assignment. Files should be named with the assignment identifier, your LUC id, and then a descriptive file name (e.g. A4whonigFindingObjects.doc, L9whonigMaximumPseudoCode.pdf, Quiz1whonig.docx). No spaces in file name. Java code files cannot use this scheme and must be ClassName.java. Entire Java projects with multiple files can be zipped and use the naming convention above (e.g. A8whonigPalindromeProject.zip).

7. No points and no corrections allowed for submitting the wrong file, incorrect file types, submitting in the wrong assignment, etc.

8. See the SakaiSubmissionCheckList for an easy process to follow for success.

Communication: please use communications effectively and in a way that is fair to all students.

9. The first choice for communication outside class is the forum discussion board.

10. Use the discussion board properly. Do not just post your question when you have a problem. Read what others have said first. If you have a related question post in the proper thread. Use a complete and detailed subject line. When replying you can modify the current subject line to make it clear what you are contributing.

11. Look at the discussion board daily (several times a day is good). There may be updates on assignments, upcoming class schedules, and other things you need to know. No excuse for not being aware of information there.

12. Read your grading comments in Sakai regularly. These comments are to help you learn.

13. Questions on grades are welcome. However, to encourage timely review of grading notes (see immediately above), no changes will be made beyond 2 weeks after the comments are posted.

14. If you don't see a grade and you know you turned in the assignment, inform the instructor and any teaching assistants immediately and no later than two weeks after the original due date and time. The instructor has a goal that all grades will be returned promptly with a target of no more than one week after they are due.

15. Grading comments in Sakai are signed with initials of the instructor and/or teaching assistants so that you know who graded what. You are welcome to ask any of us (instructor, TAs, other class assistants) about the comments.

16. Email to the instructor is only to be used for confidential items, for example concerns on your performance in class, issues with fair treatment by the instructor or other students, or other information that you consider confidential and personal. When possible use the forum so that your question can help others.

17. Email to the instructor and teaching assistant(s) should always go to all of us and have a subject line beginning with "Comp 271" and a meaningful subject line (e.g. Comp 271 Assignment A5 Grading Concern, Lost 5 Points Too Many).

Attendance and Participation: you are required to be present in all classes and prepared to participate fully both in class and outside class sessions.

18. Attendance will be taken in class. Improperly reporting attendance is a violation of academic honesty (see above).

19. You are expected to come to class with assigned readings, lab work, and other preparation done as scheduled. These materials will not be fully covered in class. Bring queries and topics you want to know more about to class.
20. You will be required to participate in focused classroom discussions between students as a key part of the learning process. This active learning is the most effective way to learn programming.
21. You are required to participate fully in online discussions outside class time and expected to both post meaningful questions and give helpful answers to other students. As a guideline, a minimum of 5 discussion posts a week is a goal.
22. You are welcome to ask questions on all assignments and course work, seek additional information on the assignments, and offer observations on the assignments. Effective help to others by answering their questions adds to your participation points for the class!
23. Class will not cover only information from the online course material and textbook; 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.
24. Do not email the instructor with notices that you are sick, or otherwise unable to get to class. Excused absences only with approved documentation from your academic advisor or the Wellness Center.
25. Participation in active learning is an important part of your grade; you cannot do well in the class without full and meaningful participation. You cannot do well in the class just by passing the final exam.

Office Hours and Help

See my web site or Sakai for current office hours. Extra time and special arrangements on request. I encourage you to set up times to talk with me about the course, programing careers, questions you have, etc.

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, Java, 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 a 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.