Algorithms and Data Structures CPSC-34000

Syllabus

Spring 2025: Last Updated 1/18/25.

I. Instructor Information

Instructor's name: Dr. Sheikh "Sam" Shamsuddin

Office location: @COD Room BIC-1537

Office hours: Tu Th 12:00 pm – 2:00 pm, by appointment

How to make appointments outside of office hours: via email, Mobile text, Virtual Zoom

Mobile texting phone number: (312) 600-7493

Lewis email address: shamsush@lewisu.edu

II. Course Information

Algorithms and Data Structures, CPSC 34000-001

3 credits

Course Description: *This course is the study of the design and analysis of computer algorithms including the data structures used in these algorithms. Topics include design techniques, such as divide-and-conquer, dynamic programming, the greedy method and backtracking, sorting, searching, graph computations, pattern matching and NP-complete problems.*

Prerequisites
MATH-31000 AND CPSC-21000

ONLINE mode

Student Learning Outcomes:

Course student learning outcomes:

1. Explain what is meant by "best", "expected", and "worst" case behavior of an algorithm.
2. Identify the characteristics of data and/or other conditions or assumptions that lead to different behaviors.
3. Determine the time and space complexity of simple algorithms
4. List and contrast standard complexity classes.

5. Use big O notation formally to give asymptotic upper bounds on time and space complexity of algorithms
6. Explain the use of big omega, big theta, and little o notation to describe the amount of work done by an algorithm.
7. Use recurrence relations to determine the time complexity of recursively defined algorithms.
8. Use a basic algorithmic strategy to solve an appropriate problem.
9. Determine an appropriate algorithmic approach to a problem.
10. Describe various heuristic problem-solving methods.
11. Implement simple search and sorting algorithms and explain the differences in their efficiencies.
12. Describe the implementation of hash tables, including collision avoidance and resolution.
13. Solve problems using fundamental graph algorithms, including depth-first and breadth-first search
14. Demonstrate the ability to evaluate algorithms, to select from a range of possible options, to provide justification for that selection, and to implement the algorithm in a particular context.
15. Describe the heap property and the use of heaps as an implementation of priority queues.
16. Solve problems using graph algorithms.
17. implement different types of singly and doubly-linked lists
18. Implement binary search, AVL, red-black, and K-d trees
19. Implement heaps and graphs (adjacency list vs. matrix).

Program student learning outcomes:

1. Develop efficient programs using languages of various programming paradigms and for a variety of platforms.
2. Make use of mathematical structures and formulations to express theoretical ideas in computer science and solve problems.

Baccalaureate Characteristics/Graduate Student Learning Outcomes:

1. Essential Skills
2. Major Approaches to Knowledge
3. Critical Thinking
4. Lifelong Learning

ABET Student Learning Outcomes[1]

(a) an ability to apply knowledge of mathematics, science, and engineering.
(b) an ability to design and conduct experiments, as well as to analyze and interpret data

[1] ABET Accreditation Board for Engineering and Technology, Inc.

(c) an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
(e) an ability to identify, formulate, and solve engineering problems
(f) an understanding of professional and ethical responsibility
(h) the broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context
(i) a recognition of the need for, and an ability to engage in life-long learning
(j) a knowledge of contemporary issues
(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

III. University Mission Statement

Lewis University, guided by its Catholic and Lasallian heritage, provides to a diverse student population programs for a liberal and professional education grounded in the interaction of knowledge and fidelity in the search for truth.

Lewis promotes the development of the complete person through the pursuit of wisdom and justice. Fundamental to its Mission is a spirit of association, which fosters community in all teaching, learning and service.

How this course connects to the University Mission:

This course embraces the Mission of the University by fostering an environment in which each student is respected as an individual within a community of learners. In the spirit of Lewis University mission of promoting the complete person and lifelong learner, this course seeks to prepare students and equip them with essential skills that students will use and apply in their day to day activities in their professional work. Furthermore, knowledge that students obtain in this course outlines a framework that students can use to acquire new knowledge on a continuous basis.

IV. Required Course Materials

Textbook(s):

Title: Data Structures and Algorithms in Python
Edition: 1st edition
Author: Michael T. Goodrich, Roberto Tamassia and Michael H. Goldwasser
Publisher: Wiley
Publication date: March 18, 2013
ISBN: ISBN-10: 1118290275, ISBN-13: 978-1118290279
Required or Optional: Required

Hardware and software requirements:

- *General-purpose laptop/desktop*
- *Python 3.7 or above, or PyCharm community edition*

V. Instructional Methods and Activities

Modality of Instruction:

This course is an online section. We will utilize available videos as well as online lectures when necessary for the delivery of course material. The instructor will send an invitation when a lecture is conducted. Blackboard and email will be our primary communication tools. The weekly course lecture material will be thoroughly discussed using class notes, videos, and Python code examples. Students are expected to do required readings, watch posted recordings, and participate in all sessions. Outside of watching recordings videos and lectures, students are expected to spend a considerable amount of time writing and debugging Python code as part of their weekly programming projects.

VI. Course Schedule

Week 1	Chapter 1 Python Primer	**Written Assignment** **Programming Project**
Week 2	Chapter 2 Object-Oriented Programming	**Written Assignment** **Programming Project**
Week 3	Chapter 3 Algorithm Analysis	**Written Assignment** **Programming Project**
Week 4	Chapter 4 Recursion	**Written Assignment** **Programming Project**
Week 5	Chapter 5 Array-Based Sequences	**Written Assignment** **Programing Project**
Week 6	Chapter 6 Stacks, Queues, and Deques	**Written Assignment** **Programing Project**
Week 7	Chapter 7 Linked Lists	**Written Assignment** **Programing project**
Week 8	Midterm	**Midterm**

Week 9	**Spring Break**	**Spring Break**
Week 10	Chapter 8 Trees	**Written Assignment** **Programing project**
Week 11	Chapter 9 Priority Queues	**Written Assignment** **Programing project**
Week 12	Chapter 10 Maps, Hash Tables, and Skip Lists	**Written Assignment** **Programing project**
Week 13	Chapter 11 Search Trees	**Written Assignment** **Programing project**
Week 14	Chapter12 Sorting and Selection	**Written Assignment** **Programing project**
Week 15	Chapter 13 Text Processing	**Written Assignment** **Programing project**
Week 16	Chapter 14 Graph Algorithms	**Written Assignment** **Programing project**
Week 17	Final Exam	**Final Exam**

Schedule Changes: *the above schedule is tentative; the instructor reserves the right to do modifications based on student progress and feedback. Any modifications to the above schedule will be discussed and communicated verbally and via blackboard announcements.*

VII. Grading Criteria and Course Policies

Assignments and Course Requirements:

Grade Distribution:

75% - all Assignments
25% - all Midterm and Final

≥ 90% …….A

≥ 87% and below 90%.....B+

≥ 83% and below 87%.....B

≥ 80% and below 83%.....B-

≥ 77% and below 80%.....C+

≥ 73% and below 77%.....C

≥ 70% and below 73%.....C-

≥ 67% and below 70%.....D+

≥ 63% and below 67%.....D

≥ 60% and below 63%.....D-

below 60%.....F

Course Policies: submissions will be evaluated using criteria reflecting the nature of the assignment. Your work MUST be completed and submitted by the due date using dedicated links in Blackboard. No late submissions will be accepted except in extreme emergency situations and with Instructor's written approval (via email).

Written Assignments: these are selected sets of questions from the textbook end-of-chapter problems, all questions in a written assignment MUST be answered. All answers MUST be put in a single Word document and submitted via Blackboard.

Programming Projects: the purpose of programming projects is for students to practice writing programs that are efficient and easy to read. Submitted code MUST be free of syntax errors and it MUST run successfully, before considered for grading. Source code that does not run will receive a grade of zero. Other grading criteria include addressing of all functional and non-functional (performance) requirements listed in the problem description, code readability, style and use of best programming features for the job, this includes use of classes, inheritance, appropriate data structures, meaningful variable names, etc.

Submission *– if the assignment involves writing a program, you must submit your program file and your program output to get full credit. Please do not zip your files. Upload all at once and then click the SUBMIT button to submit to the blackboard. Email submissions will not be graded.*

Exams: there will be one Midterm and one Final exam, any material we cover in lectures, written assignments and programming projects may appear in the exams. Although you should not expect to be asked to write an entire program in those exams, you may/will be asked to write certain functions and code snippets. Code will be graded for correctness and efficiency; minor syntax errors are okay as long as they don't compromise instructors' ability to read and understand data structure and algorithms used in the code.

Collaboration on course projects: collaborating on certain aspects of your code to resolve syntax errors, discuss programming language features, debate its performance, etc. is permitted, in fact, encouraged. A discussion forum will be available in a blackboard shell to facilitate such collaboration. However, sharing/exchanging or verbatim use (i.e., copy/paste) of a large portion of your code is NOT allowed (>20%).

You can observe, participate in these discussions, and learn from others' ideas, however, you need to write down and submit your own implementation.

Attendance
Participation is particularly important for the first two weeks in order for the instructor to confirm students' registration status and avoid inclusion in the NO SHOW list submitted for administrative staff.

Changes to Course Assignments or Grades: any changes to the above grading components or criteria will be announced via email or the blackboard announcements.

VIII. Information for Students

Requests for Reasonable Accommodations
Lewis University is committed to providing equal access and opportunity for participation in all programs, services and activities. If you are a student with a disability who would like to request reasonable accommodation, please speak with the Learning Access Coordinator at the Center for Academic Success and Enrichment (CASE). Please make an appointment by calling 815-836-5593 or emailing learningaccess@lewisu.edu. Since accommodations require early planning and are not provided retroactively, it is recommended that you make your request prior to or during the first week of class. It is not necessary to disclose the nature of your disability to your instructor. For more information about academic support services, visit the website at: www.lewisu.edu/CASE.

Lewis University has adopted Blackboard Ally providing alternative formats for files uploaded by instructors. Students can click the down arrow next to any file, and select *Alternative Formats.*

Sanctified Zone
Guided by its Catholic and Lasallian heritage, Lewis University is firmly committed to fostering a campus atmosphere that is permeated by its Mission values of Fidelity, Wisdom, Knowledge, Justice, and Association. Accordingly, we have declared the University campus to be a Sanctified Zone, a place and a people *United in Diversity.* The active promotion of diversity and the opposition to all forms of prejudice and bias are a powerful and healing expression of our desire to be Signs of Faith (Signum Fidei) to each other. To learn more about the Sanctified Zone, please visit: http://www.lewisu.edu/sanctified zone

Academic Integrity
Scholastic integrity lies at the heart of Lewis University. Plagiarism, collusion and other forms of cheating or scholastic dishonesty are incompatible with the principles of the University. Students engaging in such activities are subject to loss of credit and expulsion from the University. Cases involving academic dishonesty are initially considered and determined at the instructor level. If the student is not satisfied with the instructor's explanation, the student may appeal at the department/program level. Appeal of the department /program decision must be made to the Dean of the college/school. The Dean reviews the appeal and makes the final decision in all cases except those in which suspension or expulsion is recommended, and in these cases the Provost makes the final decision.

University Student Complaint Policy
The University Student Complaint Policy can be found at lewisu.edu/studentcomplaints

University Grade Appeal Policy
The University Grade Appeal Policy can be found at lewisuedu/studentcomplaints

Additional policies and handbooks for this program, department, and college and where they can be found *(list policies or handbooks and where they can be found, or provide a link to the web location – delete if this does not apply)*

Center for Health & Counseling Services
To support student success, all Lewis students are eligible for free health and mental health services on the Romeoville campus. This includes commuters and those living on campus, part-time and full-time students, graduate and undergraduate students, and those taking Lewis classes at other locations. For more information, visit the Center for Health & Counseling website at www.lewisu.edu/studentservices/health or call (815)836-5455.

Flexibility, Accommodations, and Student Absences
Because we are committed to student success, the University community is committed to academic standards while maintaining flexibility and empathy. Absences relating to the Coronavirus crisis will be recognized as excused. Students experiencing disruptions in their lives related to the Coronavirus that impact class attendance and participation should contact their instructor and/or college Dean's Office for assistance. Students directly impacted by Coronavirus will have the ability to request alternative grading this semester. Requests will be evaluated on a case by case basis and will require documentation.

Students who require academic accommodations due to disability caused by COVID-19, or to limit the risk of exposure to Coronavirus, can engage in an interactive process with the Learning Access Coordinator to explore all avenues for accommodations. Students can contact the Academic Services office at 815-836-5593 or learningaccess@lewisu.edu to request an appointment.

Additional Academic Integrity statements

- Please do not post course materials of any kind on Internet sites such as (but not limited to) Course Hero and Chegg without the consent of the instructor.
- Do not copy or take pictures of course materials such as videos, exams, quizzes, or assignments and post the copied items and/or images on the Internet **or** share these copied items and/or photos with other students who have not yet completed the assignments.
- Do not take pictures or copy course materials that are considered confidential by the instructor such as exams or quizzes.