MAC 281: Discrete Structures

Spring 2020

Instructor:	Prof. Jessie Oehrlein	**Email:**	joehrlein@lagcc.cuny.edu
Call me:	Prof. Oehrlein/ Prof. O	**Pronouns:**	she/her
Class Time:	MW 4:35 – 5:35	**Classroom:**	Blackboard Collaborate

Course Information

Course Description: Any code we write can be thought of as an algorithm for the particular task we want to do. This class explores the mathematics of algorithms and objects that we often use algorithms to work with and understand. Mathematical tools allow us to show that our algorithms work and help us understand how efficient those algorithms are. The second half of the course will focus on graphs and trees, mathematical structures that flexibly represent relationships among different objects, places, or people.

Learning Goals: Upon completing this course, you will be able to:

- Design algorithms for solving different computational problems.
- Analyze the complexity of algorithms.
- Prove properties of recursively-defined structures or algorithms.
- Construct and analyze graph models for problems in different contexts.
- Use standard algorithms to solve common computational problems involving numbers, arrays, graphs, and trees.

Textbook: *Discrete Mathematics and Its Applications* by Kenneth H. Rosen. Any edition is fine, though the 6th, 7th, or 8th editions are preferred.

Prerequisites: MAC 101 Introduction to Programming and MAT 231 Discrete Math, or equivalent.

Student Hours: Wednesday 3:30-4:30 pm on Blackboard Collaborate. This is a time when I am available specifically to work with you. Please join in if you have questions, concerns, insights, or to say hello! I am also always happy to answer questions by email or on the discussion board and will respond within 24 hours.

What Will Class Be Like?: During the week, there will be a wide variety of suggested individual work (activities, videos, readings, discussion board posts) depending on the topic and what you think will help you most. We'll meet synchronously during our Monday and Wednesday class time through Blackboard Collaborate Ultra so that we can explore ideas in small groups and do practice problems.

Assessment

Formal assessment in this course is based around Learning Targets tied to what you should learn. There are 10 Core Learning Targets and 7 Supplementary Learning Targets. Learning Targets are assessed through homework, quizzes, projects, and reassessments as met or not yet met. A Standard is considered to be mastered if you meet it twice.

The types of assignments are:

- **Homework:** 10 homework assignments consisting of Review/Preparation, Practice, and Extension problems, as well as reading assignments. Graded on completion. Complete and correct responses to Practice problems count towards meeting relevant Learning Targets. Attempted problems that don't meet the relevant Learning Target may be revised and resubmitted within one week.
- **Projects:** Explore ideas from class in more depth, often involving implementing algorithms (in a language of your choice). Projects are assessed as successful or not yet successful. They also count towards meeting or mastering relevant Learning Targets. Projects may be revised and resubmitted. There will be 7 choices for projects throughout the term. You may do as many as you would like.
- **Quizzes:** 9 quizzes. These consist of problems on all Learning Targets we have completely covered to that point. I will announce ahead of time which Learning Targets will be on the upcoming quiz, and quiz descriptions will also include this information. You may attempt as many or as few problems as you like. **This means that you need not do every quiz.** Quizzes will be administered through Blackboard and will be available from Wednesday during the class period through the next Tuesday for you to take when most convenient. You may take as much time as you need within one sitting. NOTE FOR QUIZ 8 AND QUIZ 9: This is now 2 hours per section of the quiz, each in one sitting. Sections are specified in Blackboard.

Grading Policy: The table below shows the requirements for each base grade. Plus/minus modifiers are determined by exceeding or not quite meeting single categories.

	A	B	C	D
Core Learning Targets	Meet 10, master 8	Meet 10, master 4	Meet 8	Meet 4
Supplemental Targets	Meet 4, master 2	Meet 3, master 1	Meet 2	
Projects	2 Successful	1 Successful		
Homework/Quizzes	7 HW complete	6 HW or 10 HW/quizzes	4 HW/quizzes	

Important Dates:

Quiz #1 ..... March 25-31
Quiz #2 ..... April 1-7
Quiz #3 ..... April 22-28
Quiz #4 ..... April 29-May 5
Quiz #5 ..... May 6-May 12
Quiz #6 ..... May 13-19
Quiz #7 ..... May 20-26
Quiz #8 ..... May 27-June 2
Quiz #9 ..... June 3-June 11

Course Policies

Integrity and Respect*: Mathematics and computer science as disciplines involve both individual and collaborative work, and this course incorporates both. I expect you to honestly represent your own

work. The course will be conducted in accordance with LaGuardia Community College's Academic Integrity policy.

Significant portions of this course involve groupwork and discussion in class. So that everyone feels comfortable participating in these activities, we must listen to each other and treat each other with respect. Diversity and individual differences are a source of strength in this classroom and community. Any attitude that one group of people is superior to another is not welcome here. One of the greatest failures of mathematics and computer science, historically and in the present, has been exclusion of voices from the field. Everyone here can learn from each other, and doing so is vital to the structure of the course.

Access and Accomodations*: I am committed to building a learning environment with you where all students can participate fully and succeed. My goal is to provide a variety of experiences and resources so that everyone has access to course content. Mathematics and computer science are human activities, and my goal is always to foster an environment that recognizes your humanity and the inherent value in your ways of knowing, doing, and communicating.

You are always allowed to ask for an extension on a homework assignment or project. In most cases, I will grant such a request, and together we'll decide on a new deadline. This will generally be a date that you think is fair and feasible for you to complete the assignment. Keep in mind that I cannot take assignments after Friday, June 12.

If you have any access needs that I can better support by changing some aspect of my teaching, class procedures, or class culture, please let me know! Even if you aren't sure exactly what you need, that's a conversation I want to have with you. You are welcome to talk to me in-person in public or private or via email. Please talk to me if you need accommodation of your disabilities. I honor self-diagnosis, and I want this course to be as accessible as possible.

Disabled students may also officially register with Disability Services; more information is available on the Disability Services site.

Your well-being is of utmost importance. If you are facing challenges to your mental/physical health or obstacles like food or housing insecurity, please don't hesitate to get in touch so that we can find appropriate resources and put you in the best position to succeed.

*Some language borrowed or adapted from Drs. Lydia X. Z. Brown, Melissa Cheyney, Joshua Bowman, Spencer Bagley, and Aunchalee Palmquist.

Learning Targets

Core Learning Targets

☐ ☐ C1: Show that an algorithm is correct, complete, and finite.

☐ ☐ C2: Analyze the complexity of an iterative algorithm.

☐ ☐ C3: Use the Euclidean algorithm to find the greatest common divisor of a pair of large positive integers.

☐ ☐ C4: Prove properties of recursively defined structures using structural induction.

☐ ☐ C5: Analyze the complexity of a recursive or divide-and-conquer algorithm.

☐ ☐ C6: Determine when a relation is an equivalence relation, and determine the equivalence class of an element or whether two elements are equivalent.

☐ ☐ C7: Give an example of a graph having a given set of properties.

☐ ☐ C8: Describe the basic properties of a given graph.

☐ ☐ C9: Trace the traversal of a tree according to pre-order, in-order, post-order, or level-order algorithms.

☐ ☐ C10: Solve shortest-path problems using Dijkstra's algorithm and explain the process.

Supplemental Learning Targets

☐ ☐ S1: Sort a list by mergesort or quicksort and explain the process.

☐ ☐ S2: Determine when a set with a relation is a partially-ordered set and a well-ordered set.

☐ ☐ S3: Write a proof showing that a family of graphs has a certain property.

☐ ☐ S4: Explain why two graphs are or aren't isomorphic.

☐ ☐ S5: Determine whether a given graph has an Eulerian path or cycle or a Hamiltonian path or cycle.

☐ ☐ S6: Search/traverse a graph using breadth-first search or depth-first search and explain the process.

☐ ☐ S7: Find the minimum spanning tree of a connected graph and explain the process.