Syllabus for MATH 374 (Discrete Structures), Spring 2023, Section 003, face-to-face instruction

The Provost issued last minute new policies after I published the syllabus for this course. Therefore the old syllabus is null and void, this is the replacement.

Instructor: Laszlo A. Szekely **e-mail:** szekely@math.sc.edu

Office: LeConte 316

Text: Judith L. Gersting, Mathematical Structures for Computer Science, Sixth Edition, W. H. Freeman and Company, New York, NY, 2007.
ISBN 978-0716768647
Do not buy a different edition. Different editions list different problems under the same problem numbers, it is your responsibility to submit the assigned problems from the sixth edition. Submitting solutions to problems that I did not assign bring zero score. You might run into a reprint of this edition by a different publisher. If the reprint has the same ISBN number, then it is the same book for our purposes.

Prerequisite: grades of C or better in both CSCE 146 and MATH 142.

Health policies: COVID vaccination is strongly encouraged. Make sure that you are familiar with and follow the university COVID policies as they evolve.

General information: Use the Blackboard page of the course regularly. On the "Course content" page you will find the syllabus, course information on what was covered in classes, reminder to homework assignments, practice tests, announcement of upcoming tests, etc. Check the "Course content" page regularly. You will find your test, homework, and quiz grades on Blackboard. Do not pay attention to the “Total” grade. It is a Blackboard function that cannot be turned off, but it has no relation to your grade.

Schedule of the semester:

Classes Begin	Jan 9 M
MLK Day	Jan 16 M (no classes)
Spring Break	March 5 Sun -March 12 Sun (no classes)
Drop Date	March 27 M
Last day of class	April 24 M

Reading Day April 25 T
Final test April 26 W 9:00 a.m .

Class meetings: MWF 10:50-11:40 in Petigru 213.

Office hours: **MWF** 12:00-13:00 in my office, LeConte 316. You may make an appointment by e-mail for other time slots. For your convenience, office hours made by appointment may be performed through the Blackboard Collaborate Ultra classroom. There are no office hours during the first week of the semester.

Use of e-mail: Use your university assigned e-mail address, otherwise I may not see your message. Mention MATH374 in your message, otherwise I may not know which class you talk about.

Attendance policy: Students are expected to attend classes and actively participate in classes. Your semester grade depends on your results, not on your attendance. Experience shows, however, that gap in attendance leads to gaps in knowledge, which hurts the semester grade.

Absences: The University allows for students to petition for an excused absence due to any of the following circumstances: medical conditions or illness or injury, death or severe illness of an immediate/dependent family member, military duty, legal obligation, religious holiday, or participation in an authorized university activity. In order for an absence to be recognized as ``excused", the student must submit the appropriate form on the website of the office of the Undergraduate Student
Ombuds: https://www.sc.edu/about/offices_and_divisions/student_affairs/our_initiatives/academic_success/ombuds_services/index.php. Make sure the instructor is notified by the Student Ombudsman. Additionally, in the case of absences due to religious observance or participation in authorized university activity, the student must notify the instructor in advance, as early the student is aware of the conflict. This helps at scheduling the midterms. If you are to miss classes (especially a test), let me know in advance *as early as you can*.

Tests: Two midterm tests will be taken during the semester. At least a week's notice will be given on test days.

A comprehensive final test is scheduled to April 26 W 9:00-11:30. The final test is not optional.

Homework assignments: Homework will be assigned regularly through the Blackboard "Course Content" page, with about one week to complete them before the deadline. Only solutions of designated problems will be collected in

class. If you cannot come to class on the day when the homework is collected, you may bring the homework earlier to my office and push it under the door, or let a classmate bring your homework to class. If none the above are feasible in an emergency, then you may e-mail to me your solution by the time of the class.

Quizzes: During classes, 10-minute quizzes will be solved regularly. Quizzes force you not to fall behind.

Complaints on grade scores: for questions on scores earned on a test problem, show me your test within a week after it was returned to you. Quizzes and homework assignments will graded by the grader, who handles all papers by the same standards. So only the grader may know how much credit is due for a problem or subproblem, or how much partial credit is due for something. In case you have issues with a grade score on a quiz or a homework assignment, write up your point for the grader, attach it to the page in question, and send it back to the grader at the next submission opportunity.

Make-up policy: (The difficulty is that I cannot give out the same test problems for a make-up test, but in case I twist the problems, there might be claims that the new problems are harder.) Students have the choice between make-up policy A and make-up policy B. I consider policy A as default, you have to let me know by January 22 by e-mail if you choose policy B.

Make-up policy A:

No late or make-up homework for any reason. Homework is due in class. You may submit your homework earlier or send it with someone else. In exceptional circumstances you may submit your homework as an e-mail attachment – still by the due time of the class meeting. About 20%-25% of your weakest or missed homework submissions will be dropped before computing the homework grade. No late or make-up quizzes for any reason. About 20%-25% of your weakest or missed quizzes will be dropped before computing the quiz grade. No make-up tests can be written for any reason. If you miss a test with an ``excused" reason, as justified by the Student Ombudsman, the grade of the final test will substitute the missed test grade. If you have extended illness during the semester, then contact me on how to catch up on the material and to discuss deviation from the grading policy to compute your grade.

Make-up policy B:

Late and make-up homework, make-up quiz, and make-up test opportunities are available for absences for legitimate reasons, if those are justified by the Student Ombudsman. Lacking Student Ombudsman justification, a missed homework, quiz or test brings a zero score. In Policy B, there are no dropped homework and quiz scores, all of your homework and quiz scores contribute to your grade. If you have extended illness during the semester, then contact me on how to catch up

on the material and to discuss deviation from the grading policy to compute your grade.

Grading: 500 points can be collected at the course, 100 from the each of the two tests, 100 from the final exam, 100 from the homework, and 100 from the quizzes. The final grade is based on the ranges:
A [92,100] B+ [87,92) B [82,87) C+ [77,82) C [72,77) D+ [67,72) D [60,67) F [0,60).

Work expectation. You will find this course very intense, with not enough time for examples and practice in class. The topics to be covered are dictated by accreditation requirements. I hardly have enough air time to mention all of the required topics. Some schools teach discrete structures in two semesters, some schools teach it in one semester. Your degree program allows one semester for discrete structures. Be prepared to learn material enough for two semesters in one. This requires on your side a minimum of 9 hours of work in addition to attending classes. You have to work through extra examples and practice problems from the textbook on your own. Mathematics is not a spectator sport. You may follow everything in class, but it is always a different experience if you have to do it yourself. Make sure that you use the notation of the text and/or of the lectures. Use of proper notation and terminology is an indispensable part of mathematics and computer science. I often get questions that somebody wants to use the notation of a different class he had before. This is not acceptable. I do not know and do not need to know notation of all kind of courses and texts. Do not invent your own notation either. Inventing new notation usually has the point that you do not want to write too much. New notation could be acceptable if you properly define it. If you define it properly, it will take more writing than the textbook notation would. (You cannot use ad hoc commands in a code either!) Make sure that you apply equality sign (=) between terms that are of the same type (and indeed are equal). The equality sign is not for punctuation. Statements, numbers, sets are different concepts, and consequently a number is not equal to a statement, or to a set, etc.
You may get partial credit, if your solution is not correct, but your work shows correct steps towards a correct solution. If a wrong answer does not show details, of which some steps are correct, there is no basis for partial credit. Always try to give details of your argument. Writing is your friend, not your enemy. Learning proof writing is part of this course. Without a proof, you never really know if your algorithm solves your problem or not. Considering that basic proof writing is a course in itself (MATH300), we do not get very deep into proof writing. But keep in mind that proofs are written in *sentences.* A sequence of formulas is not a proof, unless the logical connection among the formulas is elucidated.

Graduate requirement. Graduate students are expected to gain deeper knowledge than undergraduate students. Therefore students, who are enrolled

as graduate students, in addition, must submit correct solutions to 5 graduate problems. This is a condition to pass, otherwise it has no effect on the grade letter. An alternative way to satisfy the graduate requirement is to write a 10-pages informative essay on the history of discrete mathematics. Graduate problems will be collected only from students, who are enrolled at this course as graduate students. Undergraduate students may get *extra credit* for solving graduate problems, but not for essay writing. (Currently no graduate student is enrolled in this course, but inclusion of a paragraph on graduate requirement in the syllabus is a condition of accreditation.)

Students in need for special accommodation: Contact the Student Disability Resource Center and let them issue a letter about your needs. Contact the professor early in the semester to discuss the ways of accommodation. If you are allowed to use extra time for tests, schedule the proctoring of your test with SDRC as early as possible, to a time slot that overlaps with the test time in class. If you have accessibility issues, contact the professor early for accommodation. (Mathematics is written in TeX or LaTex mathematical text editors to create pdf files, which look like mathematics books. Those pdf files are formula rich and contain figures. Technology turning those into sound is not available yet.)

Academic integrity: The professor is required to report academic integrity violations. Working together with some classmates and discussing homework problems is perhaps the best way to learn mathematics. What you write down and submit, however, must be your individual work.
Do not discuss with anyone test or quiz assignments and only use tools that were allowed for that test or quiz. If you have questions about the assigned problems in tests or quizzes, ask me; send me an e-mail if you have a question about a homework problem.

Phone policy: Your phones and other devices should not ring or play music during class. During tests and quizzes, you are not allowed to touch any electronic device (phone, laptop, iPad, smart watch, calculator, etc.) or wear headphones. No observer can tell what they are being used for.

Use of text, handouts, and solutions: Students are, of course, expected to read the text after class. However, reading the next section—or whatever was assigned to read---*before* class is also expected. This will help the students in seeing the outline of the current topic, giving familiarity with the basic concepts, and getting the fine points of the lecture easier. Solutions to quiz and test problems (and select homework problems) will be provided. Read them and learn from them, understand your errors, if there were any. On your graded work you find some feedback. Read those comments to avoid repetition of mistakes.

Course content: Course content: propositional and predicate logic; proof techniques; recursion and recurrence relations; sets and combinatorics; functions, relations, and matrices; graphs and trees; algebraic structures. These mathematical concepts are absolutely necessary for students of computer science.

Learning outcomes: The goal of studying logic is to make the basis for proofs and program correctness. Students are expected to translate English sentences into predicates and vice versa. They will be able to evaluate truth values and verify tautologies using methods of logic and will be familiar with the principles of declarative programming languages.
The goal of studying proof techniques is to convince others (and ourselves) about the correctness of our algorithms. Mathematical induction is the basic tool to follow what a loop is doing in an algorithm. Emphasis will be on the applications of proofs, not on subtleties of writing proofs. Recursion is basic technique for simple definition of complicated objects – however, proving anything for recursively defined objects requires mathematical induction! Students will be able to write recursive algorithms, prove the correctness of simple algorithms, solve simple recurrences, and use mathematical induction, in particular to show the correctness of loops. Relations, graphs and matrices are the basic structures for mathematical modelling of discrete real-world phenomena. They provide a natural language for posing and solving problems, or for writing programs to solve them. Students will be able to use these concepts to model real-life situations and to operate within the models they created. Graphs occur naturally in many discrete real-life optimization problems. It is important to gain command of the terminology of graph theory. Trees are important objects in graph theory, since they make the simplest interesting class of graphs. However, our interest in graphs is supported by their significance in computer science. Students will be familiar with basic concepts of graphs and trees and use them in modeling.

Disclaimer: It is not possible to make a rule in the syllabus for all situations that might emerge. Furthermore, all federal and state laws and regulations, and all university rules apply during the course – irrespective of whether you know them or I know them. As those rules may change, the syllabus will change with them.