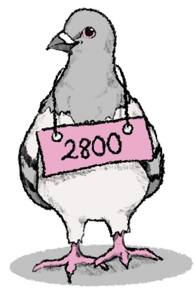
Mathematical Foundations of Computing

CS 2800, Fall 2024

Course Syllabus



Course Overview

Discrete mathematics serves as the theoretical backbone of computer science. From the design of logic gates in low-level hardware to the development of verifiable correct code in large software systems, computer scientists rely on tools from discrete mathematics to design and reason about computational tools. Beyond this, developing mathematical maturity – being able to read and understand new definitions, trace through and develop multi-step procedures and arguments, and verify or disprove claims – is crucial for programming, research, and scientific communication.

In this course, we will study the mathematical foundations of computer science starting at the most basic level. The focus of the course will be on many *discrete structures* (i.e. objects that are characterized by distinct, separated values as opposed to the *continuum* of real numbers) such as logical formulas, sets, functions, relations, and automata. An emphasis of this course, in contrast to other math classes that you may have taken, will be writing rigorous justification (or a *proof*) to explain your mathematical reasoning fully. As such, this course aims to introduce you to important definitions and theorems *and* teach you how to think critically and gain confidence when solving new mathematical or computational problems.

Course Objectives

At the end of this course, you will be able to :

- Use the foundational mathematical objects (sets, relations, functions, automata, and probability spaces) to model problems arising in Computer Science.
- Read and understand *unfamiliar* mathematical definitions, generate examples and counterexamples, formulate precise definitions, and determine whether a given object satisfies a definition.
- Write clear and precise mathematical proofs using standard techniques such as mathematical induction. Find and correct errors in faulty proofs.
- Apply counting techniques and probabilistic reasoning.

Logistics

Credit Hours:	4
Instruction Mode:	On campus
Class Format:	Three 50-minute lectures, one 50-minute discussion per week
Supplemental:	Approximately 10 hours of independent work/study per week. This will vary depending on your background, desired grade and studying efficiency.
Lecture Time:	MWF 11:15 AM-12:05 PM
Lecture Dates:	August 26 - December 9, 2024
Location:	Statler Hall 185 (Aud)
Instructor:	Matt Eichhorn meichhorn@cornell.edu Office: 452 Gates Hall Office Hours: T 2:00-4:00 PM F 10:00-11:00 AM
Course Coordinator:	Kim Budd kj37@cornell.edu
TAs:	We are fortunate to have around 50 teaching assistants, including undergrad, Masters, and PhD students. More information about our TAs is on Canvas and Ed.

Prerequisites/Corequisites

There are no prerequisites for CS 2800. The official corequisites are an introductory calculus course (at the level of MATH 1110) and an introductory programming course (at the level of CS 1110/1112). Mathematical familiarity at the level of a high school curriculum is assumed; you should be comfortable working with familiar functions (e.g., polynomial and exponential) and performing arithmetic without a calculator. If you have concerns about meeting these requirements, please email Matt.

Course Textbook

You are not required to purchase any text for this course. Rather, a set of comprehensive and self-contained course notes will be available on Canvas. These notes are broken down by lecture and include many worked examples and additional exercises. Written solutions for all of these exercises will not be provided. Rather, we encourage you to collaborate with classmates, engage in discussion on Ed, and speak with Matt and the course TAs to guide you through these exercises. Harder exercises are marked with a $\star$ and often surpass the expectations of the course. Exercises that require knowledge from other courses are marked with a $\diamond$.

Optional Texts

Many excellent discrete math books offer an alternative presentation of the material, which may be helpful in your studies. Some recommended references are:

- *Discrete Mathematics and its Applications*, by Kenneth Rosen.
- *Essential Discrete Mathematics for Computer Science*, by Harry Lewis and Rachel Zak.
- *Mathematics for Computer Science*, by Eric Lehmann, Albert Meyer, and Tom Leighton.

The topics covered by these texts differ (to varying degrees) from our course. Also, the levels of formality and difficulty of these texts vary, and some of the notational choices differ from or conflict with ours. For consistency, we expect you to use the notation presented in lectures and the course notes.

Online Resources

We will use three websites to help manage the course. You should monitor these sites regularly.

Canvas: canvas.cornell.edu/courses/66272

Canvas will serve as the main course website. Our Canvas course page includes an updated course calendar with links to all lecture notes, slides, discussion worksheets, assignments, and posted solutions. We will also use Canvas to track attendance and manage our gradebook.

Gradescope: gradescope.com/courses/809018

We will use Gradescope to release, collect, and grade homework, and to upload and grade exams. Gradescope also has a companion mobile app that you can use to take pictures of and upload your assignments.

Ed Discussion: edstem.org/us/courses/61089/discussion/

Ed provides a discussion board where we will post announcements for the class, and where you can ask and answer questions about the course material, assignments, and logistics. These posts can be public (viewable to the entire class) or private (viewable only to the instructors).

Lectures

Each week, we will have three 50-minute lectures where we introduce new concepts, definitions, and proof techniques and work through example problems as a class. The content of the lectures will mostly follow the typed course notes, which will be made available in advance. Annotated slides for each lecture will be made available on Canvas within 24 hours after class.

You are expected to participate in the lecture, ask and answer questions, and engage in class discussions. To facilitate this, we will use Poll Everywhere, which will allow you to answer questions that are posed in lecture and interactively see the responses of other students. Participation in these polling activities will be graded, with full credit awarded for each correct answer and 80% credit awarded for each incorrect answer. To provide flexibility in the case of missed classes, internet issues, etc., the overall score for this portion of the grade will be determined out of 80% of the total possible points (truncated to full credit).

For example, if there are 120 questions over the course of the semester, you answer 60 of these correctly, answer 40 of these incorrectly, and miss class for 20 of these questions, your total score will be

$$\frac{60 + 40 \cdot 0.8}{120 \cdot 0.8} = \frac{92}{96} \approx 95.8\%.$$

Answering a polling question when you are not physically present in lecture constitutes a violation of academic integrity and will result in a score of 0 for this grade component.

Discussion:

Each week, you will attend one 50-minute discussion section. The focus of discussions will be to work through curated exercises in small groups to reinforce and augment concepts from lectures. Attendance and participation (completing assigned worksheets and reflection activities) in discussion is mandatory. To earn full credit, you can miss at most *two* discussion sections. You must attend the section at the time and location specified in Student Center. If you need to make a one-off arrangement to attend an alternate discussion section, please make a private post on Ed.

Homework:

The goal of homework is to make you think more deeply about and work with the material in the course and understand which concepts you need more help or practice with. Most weeks, there will be homework assigned on Wednesday and due the following Wednesday at 11:59 PM. Assignments are released, turned in, and graded on Gradescope. Homework assignments will usually consist of three problems.

Homework Policies:

1. Your two lowest homework scores (potentially including a 0 earned for an unsubmitted assignment) will be dropped; they will not factor into the calculation of your course grade.
2. While you may discuss problem approaches with your classmates, the work you turn in must be your own. You may not copy anyone's written work; as soon as you start writing your solutions, collaboration should stop. In particular, writing out ideas or solution sketches on a whiteboard is not allowed.
3. The only resources you may use for your homework are those linked on Canvas (course notes, slides, discussion worksheets, past assignments) and the optional texts listed in the syllabus. Do not consult outside sources (e.g., online forums, tutoring sites, generative AI) to help prepare your solutions.
4. A cornerstone of proper academic conduct is the proper attribution of one's information sources. You must include a collaborator statement on each homework assignment, whether or not you have collaborated with anyone. This should include a list of all students who you discussed the homework with and office hours that you attended. If you worked alone on the assignment, you must write this explicitly in your collaborator statement. Failure to do this may result in a small grading penalty.
5. Academic integrity violations (i.e. non-adherence of the above points) will be handled seriously and may result in penalties such as receiving a score of 0 (that cannot be dropped) on that assignment.
6. The submission deadlines are handled automatically by Gradescope, and it is your responsibility to ensure that your submission is turned in on time. As technical difficulties (network connection, errors on the website, problems with scanning or file conversion, etc.) may arise, it is advised that you submit your homework well in advance of the deadline. Submission issues that are brought to our attention after the deadline will *not* be accommodated.
7. When you submit your homework, it is your responsibility to tag each page of your assignment with the appropriate question labels; Gradescope will prompt you to do this automatically. Failure to do this may result in a small grading penalty.

Exams:

The goal of the exams is to provide an incentive to review the material and for us to check that you understand the material at a sufficient level to be able to use it in future courses. Our course will have two preliminary (prelim) exams on Tuesday, October 1, and Thursday, November 21. The time of the final exam will be released later in the semester.

Exam Policies:

1. Exams in this course are closed-book. You may not bring any reference materials, course notes, electronic devices, or "cheat sheets" to the exam room.

2. If you have a conflict with an exam, you must fill out the appropriate conflict form (that will be linked in a pinned post on Ed) at least one week in advance to arrange attendance at a make-up exam. On this form, you must describe the nature of the conflict which should be one of the following: (1) an in-person exam in another course occurring on the same day (2) university-approved travel, for example, to attend a conference or athletic event; or (3) religious observance of a holiday or festival that conflicts with the exam time.
3. If you find yourself ill on the day of the exam, please email Matt and Kim *before* the start of the exam to tell us that you cannot attend. We will make arrangements for you to attend the make-up.
4. Violations of academic integrity during an exam, such as using electronic devices or copying another student's paper, will result in an F in the course.
5. Details about the coverage and format of each exam and a practice exam to aid in your preparation will be posted on Ed at least one week before the exam date.

Grade Determination

Your final course grade will be calculated as a weighted combination of the course components:

Lecture Participation (Poll Everywhere):	4 %
Discussion Attendance/Participation:	6 %
Homework (2 Lowest Dropped):	25 %
Lower Prelim Score:	10 %
Higher Prelim Score:	20 %
Final Exam:	35 %

For example, if you earn 95.8% for lecture participation as described above, participate in 11 out of your 14 discussions, receive an average of 80% on the homework (after dropping your two lowest scores), 75% on the first prelim, 60% on the second prelim, and 80% on the final exam, your final grade will be

$$95.8\% \cdot 4\% + \frac{11}{14-2} \cdot 6\% + 80\% \cdot 25\% + 60\% \cdot 10\% + 75\% \cdot 20\% + 80\% \cdot 35\% \approx 78.3\% .$$

The cutoffs used to translate final grades to letter grades will be determined and published at the end of the semester. This course is *not curved*, meaning we do not pre-determine the percentage of students who will qualify for each letter grade. Grades will be posted promptly and updated regularly so you can evaluate your performance in the course.

Course Evaluation:

At the end of the semester, you will receive a link to the course evaluation for CS 2800. This is a way for you to provide candid feedback on the instruction, content, and course organization. While we receive a report of who completes the course evaluations, your answers will remain anonymized. Completing the course evaluation will earn you 5 bonus points on the final exam.

Earning an A+:

There will be 5 "Challenge" homework problems given throughout the semester. These questions are optional and will *only* be used to determine which A grades to promote to A+ (i.e., they do not factor into the formula above). To achieve an A+, you must complete at least $3/5$ problems satisfactorily (and earn an A in the course).

Grading Feedback and Regrade Requests:

Transparency and objectivity in grading is critical to ensure that all students are evaluated fairly. To this end, we will release detailed rubrics and grading comments along with your grades on all work during the semester. Please review this information carefully, as this feedback is critical for learning.

In a large course where over 1000 problem solutions are graded each week, some grading mistakes will inevitably be made. If you feel that a rubric was misapplied to any of your work on a homework assignment or exam, you should submit a regrade request.

Regrade Request Policies:

1. All regrade requests are handled on Gradescope. Do not email the TAs or instructors or go to office hours to ask for a regrade request or to ask whether you should submit a regrade request.
2. Regrade requests must be submitted within one week of when the grades are released. This window may be shortened toward the end of the semester.
3. When submitting a regrade request, be specific about which rubric items were misapplied, and how you feel the grading should be corrected. Point to specific parts of your answer.
4. Regrade requests that are rude, threatening, or condescending will not be considered and may result in an academic misconduct hearing.

Bug Bounty:

Precision is paramount in formal mathematics. To prepare for this course, we have written hundreds of pages of lecture notes and assignments. While we have been careful to find and correct any typos, we are sure there are many still out there. To help ensure that you leave the course with the best possible materials, we present the following "bug bounty:" (Please see the pinned post on Ed for more details.)

The first student to point out a typo/mistake (and suggest an appropriate correction) in the lecture notes will receive 1 point added to their total homework score. The first student to point out a typo/mistake in a homework problem will (1) earn full credit for that problem (or the appropriate subpart of that problem) without needing to submit their solution, and (2) receive 1 point added to their total homework score.

Personal Conduct

Cornell expects students to be respectful and professional in all participation and communication. You are expected to maintain professional conduct and speech in all aspects of this course. Professional behavior demands you have a responsible and mature attitude in person and online. Disrespectful, unethical, and/or unprofessional behaviors will not be tolerated and can result in course failure and/or dismissal from the program. This same professional standard is also expected of our TAs, Kim, and Matt. If anyone on the course staff behaves in a manner that you feel is inappropriate, please bring it to our attention immediately so that it can be addressed.

Prohibition Against Buying, Selling, or Sharing Course Materials

Course materials posted on Canvas, Ed Discussions, and Gradescope are intellectual property belonging to the author/instructors. Students are not permitted to buy, sell, trade, or share any course materials without the express permission of the instructors. Such behavior constitutes academic misconduct.

Absences / Late Registration

The course grading policies, which allow you to miss some homework submissions, discussion sections, and lectures without penalty, are designed to accommodate absences due to illness, travel, etc. If a situation arises that requires a prolonged (multiple-week) absence, please email Matt so we can make a plan to complete the missed work. If you join the course late, you may need to apply your homework drops (etc.) to the assignments due before your enrollment.

Academic Integrity

Each student is expected to abide by the Cornell University Code of Academic Integrity. Any work submitted in this course for academic credit must be the student's own work. The complete code is available at deanoffaculty.cornell.edu/faculty-and-academic-affairs/academic-integrity/code-of-academic-integrity/. Violations of the Code of Academic Integrity, especially plagiarism, may result in an F in the course.

In addition, this course is participating in Accepting Responsibility (AR), which is a pilot supplement to the Cornell Code of Academic Integrity (AI). For details about the AR process and how it supplements the AI Code, see the AR website.

Accommodations

Our primary goal is to establish an inclusive learning environment that is accessible, comfortable, and supportive of everyone's physical, mental, emotional, and learning needs. Services and reasonable accommodations are available to persons with temporary and permanent disabilities when conditions cause barriers to equal educational opportunity. Student Disability Services (SDS) determines the eligibility of students to receive accommodations and works collaboratively with the student and university faculty and staff to recommend appropriate accommodations. If you believe that you require an accommodation, please reach out to the SDS office at sds_cu@cornell.edu or by going to their office at Cornell Health as soon as you can. In order to have adequate time to arrange your approved accommodation(s), you should request your SDS accommodation letter no later than the add/drop deadline for the semester.

Beyond formal accommodations, if you ever have concerns about your learning, grades, or progress in the course, if you get sick, or if you are facing other challenges and need help connecting with the right resources, please don't hesitate to reach out to Matt.

Inclusivity

Students in this course come from a variety of backgrounds, abilities, and identities. We support an inclusive learning environment where diversity and individual differences are understood, respected, appreciated, and recognized as a source of strength. It is expected that students in this class will respect differences and demonstrate diligence in understanding how other peoples' perspectives, behaviors, and worldviews may be different from their own. Please contact Matt if you observe any behavior (from other students, TAs, or instructors) that runs counter to our goal of inclusivity.

Attestation

By registering for this class and accessing course materials through Canvas, students agree to abide by University, College, Department, and Course policies.

Being Successful in CS 2800

Many students feel daunted coming into CS 2800 because it is very different than many other courses that they have likely taken. Unlike previous computer science courses, CS 2800 doesn't require any coding, doesn't look at any programming languages, and doesn't use any special software or hardware. Unlike previous math courses, CS 2800 does not center on memorizing and plugging into formulas. Most of the problems you'll do are not calculations, and we are less concerned with the "answers" as much as we are with clear explanations and complete justification. Instead, this course centers on more abstract thinking: how can we use a small set of mathematical tools to put together clear and rigorous arguments? Below, I've collected some advice, both from my thoughts and opinions as well as those from past TAs and students, on how to be successful in the course.

1. **Allot yourself enough time for homework assignments**
    - Many homework problems require you to come up with new ideas that build on examples from lectures and discussions. If you don't see the solution right away, don't get frustrated.
    - Problems in CS 2800 can be challenging. It is better to work through the challenge by struggling on homework than to find "shortcuts" on homework and feel challenged on exams.
    - To get the most out of the homework problems, try them on your own first (you only have yourself for exams) before turning to your notes, then your peers, then office hours. While it may take less effort to go straight to office hours, this will short-circuit the learning process.
    - On the flip side, it can be easy to get stuck or "think yourself in circles" when tackling a hard proof. Don't be afraid to ask for help.
    - After you finish writing your solutions, the learning process from homework is not over. Step back and look at your solution. What tools did you use to get there? Why were those tools useful? How did you know (or would you know) that these tools would be helpful? If you were given a similar problem, would you feel comfortable solving it?
2. **Learn the definitions carefully**
    - A big part of writing, reading, and understanding proofs is having a good grasp of the relevant definitions. Make yourself a list of the new definitions from each lecture.
    - Beyond knowing a definition, there is extra work to do to really *understand* a definition to the point where it becomes a useful tool. If you are trying to prove that an object has a property, how do you show this? If you are given an object with a particular property, how can you make use of this fact in a proof? Can you list some objects that (do not) satisfy a given definition? Can you both explain a definition in words and formalize it with mathematical symbols?
3. **Make use of the lecture exercises**
    - Each set of lecture notes includes around 10 additional exercises to practice and reinforce the material. This adds up to hundreds of practice questions over the semester.
    - Form a small study group to work through some exercises, comparing your answers as you go (no academic integrity concerns for these exercises, since they are not graded work). If you need help forming a group, you can use this form from the Learning Strategy Center.
    - Ask about lecture exercises during office hours and on Ed discussion. This is a great way to expand your learning.

- Doing *all* of the lecture exercises is not a good use of time. Instead, skim over the exercises and choose the ones that will be most useful (Which skills do you need the most practice on?). Start with the unmarked exercises, the most relevant training for homework and exams, before moving on to the $\star$ and $\diamond$ exercises.

4. **Be pragmatic with your exam review**
   - We cover a lot of material in CS 2800, and the course progresses quickly. Many students feel a time crunch when preparing for exams.
   - Start your studying by "triaging" the topics. For each lecture, determine whether you feel completely confident (in which, reviewing won't help much), fairly confident (in which some light review will be useful), or not confident (where more attention is necessary). Are there particular topics or definitions that you need help understanding?
   - Take the practice exams as soon as they are released under exam conditions (on your own without any resources). This will help give a sense of what you need to review.
   - Don't read the practice exam solutions without attempting the questions first. Reading solutions can give a false sense of confidence and can be detrimental to studying. Instead, compare your answers to the solutions. Did you solve the problem in a different (but also correct way)? What did you miss or misunderstand? How would you come up with the intended solution? Do you feel comfortable attempting a similar problem?
   - While practice exams give a good sense of what the real exam will be like, they are not exact replicas. Don't focus too specifically on the practice exam problems. Rather, focus on the concepts and techniques that are used.

5. **Ask questions**
   - The time in lectures and discussions is meant for *your* learning. If something is unclear, don't be afraid to ask for clarification. Other students likely have the same concern, and they will appreciate that you asked.
   - Come to office hours for more than just homework help. The TAs and I love to chat about any of the topics from the course, and we want to hear your thoughts about them. We do *not* prioritize homework questions during office hours.
   - Ed is a great tool to continue the conversation beyond the classroom. Ask questions about things you're confused or curious about. Feel free to respond to other students' questions. Often, explaining things to others is one of the best ways to further your learning.

6. **Don't get discouraged**
   - Everyone comes into the course from a different starting point, but that doesn't limit who can be successful in the end. Focus on what you can do to get the most out of the course rather than on comparing yourself to other students.
   - If you feel that you are struggling, don't hesitate to reach out for help. Addressing concerns early is the best way to stay on track in this fast-paced course. Also, if there's something that the course staff can do to improve the course, let us know. We appreciate your feedback.

Lecture Schedule

The following is a tentative outline for the course. Modifications will be made where necessary and reflected through announcements on Ed Discussion or on the course Canvas page.

Day	Date	#	Topic
M	8/26	0	Course Intro / *Chomp*
W	8/28	1	Propositional Logic
F	8/30	2	Sets and Predicates
M	9/2		Labor Day: No Class
W	9/4	3	Quantifiers
F	9/6	4	Operations on Sets
M	9/9	5	Proof Techniques
W	9/11	6	Inductive Reasoning
F	9/13	7	Recursion and Recurrences
M	9/16	8	Strong Induction
W	9/18	9	Graphs
F	9/20	10	Induction on Graphs
M	9/23	11	Binary Relations
W	9/25	12	Partial Orders and Equivalence
F	9/27	13	Equivalence Classes
M	9/30	–	Catch Up / Review
T	10/1		Prelim 1
W	10/2	14	Functions
F	10/4	15	'Jectivity and Inverses
M	10/7	16	The Pigeonhole Principle
W	10/9	17	Cardinality
F	10/11	18	Countable and Uncountable Sets
M	10/14		Indigenous Peoples' Day: No Class
W	10/16	19	Deterministic Automata
F	10/18	20	The Myhill Nerode Theorem

Day	Date	#	Topic
M	10/21	21	Non-Deterministic Automata
W	10/23	22	Regular Expressions
F	10/25	23	Kleene's Theorem
M	10/28	–	More on Regular Languages
W	10/30	24	Counting Rules
F	11/1	25	Binomial Coefficients
M	11/4	26	The Inclusion-Exclusion Rule
W	11/6	27	Balls and Bins Problems
F	11/8	28	The Stirling Numbers
M	11/11	–	More Combinatorics Practice
W	11/13	29	Probability Spaces
F	11/15	30	Conditioning and Independence
M	11/18	31	Bayes' Rule
W	11/20	32	Discrete Random Variables
R	11/21		Prelim 2
F	11/22	33	Expectation and Variance
M	11/25	34	Joint Distributions
W	11/27		Thanksgiving Break: No Class
F	11/29		
M	12/2	35	Covariance and Correlation
W	12/4	36	Probability Inequalities
F	12/6	37	Continuous Random Variables
M	12/9	–	More on Random Variables
TBA			Final Exam

Learning Outcomes

The following is a list of skills that you will have after taking this course. This list is not meant to be exhaustive, and some topics may need to be reduced or omitted based on our pacing during the semester. A lot of the outcomes use terminology that we will introduce during the course, so the list is most useful in retrospect.

Module 1: Foundations

1. Explain the syntax and semantics of operations in propositional logic
2. Represent English statements using logical symbols
3. Use truth tables to determine whether two compound propositions are logically equivalent
4. Describe sets using set builder notation
5. Demonstrate understanding of predicates with one or more variables
6. Identify the free and bound variables in a quantified logical statement
7. Establish the truth value of statements containing one or more quantifiers
8. List elements of sets defined with intersection, union, difference, Cartesian product, and power set operations
9. Explain the components of a well-written formal proof
10. Discuss when to use direct proofs, case analysis, contrapositive proofs, and proofs by contradiction
11. Write formal arguments of set containment and equality
12. Construct a counterexample that refutes a proposition
13. Identify and correct flaws in erroneous proofs

Module 2: Induction

14. Explain the Principle of Induction
15. Identify the steps of a proof by induction
16. Use induction to prove statements quantified over the natural numbers
17. Explain why we must sometimes strengthen an inductive statement
18. Compare and contrast ordinary and strong induction
19. Write inductive proofs that require multiple base cases, multiple inductive steps, and/or a strong inductive hypothesis

Module 3: Graphs and Relations

20. Translate between a visual and a formal set description of directed and undirected graphs
21. Identify paths and cycles in a graph

22. Determine whether a graph satisfies common properties such as acyclicity, bipartiteness, planarity, and regularity
23. Explain how to use structural induction to establish a property of a family of graphs
24. Use structural induction to prove propositions involving graphs
25. Identify pairs of elements that satisfy a binary relation
26. Decide whether a binary relation is reflexive, symmetric, anti-symmetric, and/or transitive
27. Construct a relation on a given set that has one or more given properties
28. Justify whether or not a relation is a partial order
29. Visualize the structure of a partially ordered set with a Hasse diagram
30. Justify whether or not a relation is an equivalence relation
31. Find the equivalence classes of an equivalence relation
32. Describe the quotient of an equivalence relation

Module 4: Functions and Cardinality

33. Invoke the formal definition to determine whether a function is well-defined
34. Compute the composition of two functions
35. Determine whether a given function is injective, surjective, or bijective
36. Describe the relationship between bijectivity and function inverses
37. Write formal proofs involving function composition, 'jectivity, and inverses
38. State and apply the Pigeonhole Principle
39. Indicate when one proposition is a generalization or specialization of another
40. Use the Pigeonhole Principle to establish properties of graphs
41. Explain how one can use injective functions to define the cardinalities of sets
42. Determine whether a given set is countable or uncountable
43. Write proofs by diagonalization

Module 5: Languages and Automata

44. Differentiate between alphabets, strings, and languages
45. Give a formal description of an automaton from its graph representation
46. Determine whether a given word is in the language of a given automaton
47. Draw a DFA that recognizes a given language
48. Explain the Myhill-Nerode Theorem
49. Calculate the Nerode congruence classes of a language

50. Use the Myhill-Nerode Theorem to argue that a language is not regular
51. Perform the subset construction to convert an NFA into a DFA that recognizes the same language
52. Write a regular expression that represents a given language
53. Apply Thompson's construction to find an NFA that recognizes the language of a regular expression
54. Use state elimination to write a regular expression that represents the language of an automaton
55. Apply Kleene's theorem to reason about properties of regular languages

Module 6: Combinatorics

56. Describe multi-step processes for enumerating the items in a set
57. Compute set sizes using the Multiplication, Addition, Subtraction, Bijection, and Division rules
58. Prove identities using combinatorial reasoning
59. Explain and apply the Binomial theorem
60. Apply the Inclusion-Exclusion rule to calculate the size of a set
61. Differentiate between arrangements with distinguishable/indistinguishable balls/bins
62. Use the stars and stripes formula to calculate the size of a set
63. Demonstrate understanding of the Stirling numbers

Module 7: Probability and Statistics

64. Identify the components of a probability space
65. Utilize techniques from combinatorics to compute probabilities in discrete probability spaces
66. Calculate the probability of an event conditioned on another event
67. Describe how conditioning on an event changes a probability space
68. Determine whether events are independent
69. Explain and apply the Law of Total Probability
70. Use Bayes' Rule to calculate posterior probabilities
71. Describe how to use Bayesian Inference to update a prior probability based on new evidence
72. Explain how the probability mass function and Cumulative Distribution Function describe the distribution of a discrete random variable
73. Identify the pmf, CDF, expectation, and variance of discrete uniform, Bernoulli, binomial, and geometric random variables
74. Use the definition to compute the expectation of a random variable
75. Calculate and interpret the variance of a random variable
76. Demonstrate understanding of joint and marginal distributions of random variables
77. Describe when two random variables are independent and how this can simplify calculations

78. Explain and apply the linearity of expectation
79. Calculate and interpret the covariance and correlation of two random variables
80. Use Markov's, Chebyshev's, and Hoeffding's Inequalities to calculate tail probability bounds
81. Describe when one probability inequality may be preferable to another
82. Explain the law of large numbers
83. Interpret the probability density function of a continuous random variable
84. Identify three families of continuous distributions and describe their parameters
85. Calculate statistics of continuous random variables