Course instructor: Harry Mairson (mairson@brandeis.edu), Volen 257, phone (781) 736-2724. Office hours 11am–12pm Monday, Wednesday, Thursday, and most any time by arrangement. I especially encourage you to communicate with me via electronic mail, for fastest and most reliable responses to your questions.

Teaching assistants: Karishma Reddy Khan (kreddykhan@brandeis.edu), Damian Lin (dlin@brandeis.edu), Nick Moran (nemtiax@brandeis.edu), Xi Qian (qxx@brandeis.edu), Prakhar Sahay (prakhar@brandeis.edu). Office hours TBA.

Note: We will reschedule office hours if there is a profound mismatch with student needs. You are guaranteed a hearing during the specified hours; you are free to try dropping in otherwise or meeting by arrangement. If you’ve got questions, it’s our job to get you the answers.

Time and place: Class lectures are Monday, Wednesday, and Thursday from 10-10.50am, Mandel G03 (Humanities Center).

Required Textbook

Eric Lehman, Tom Leighton, and Albert Meyer, Mathematics for Computer Science.

The full text of the book is available on-line at the Latte site for the course. There’s also one on the internet, but there may be various versions with slight differences. (Probably you’ve been saved a small fortune by not having to spend in excess of $100 for a Yellow Pages-sized book on discrete math.)

If you don’t like reading a lot of stuff on a computer screen, you should consider printing parts of the text for easier reading. If there’s a big demand, I will consider organizing a print run through the department for a nominal fee ($10 or $20).

What is this course about?

This course is in effect a math course, involving the mathematical ideas that are relevant to understanding concepts in computer science. The plan is to discuss the following topics:

Proofs are in many respects the mathematician’s analogue of a computer program. That’s because lots of proofs have constructive, computational content.

Logic and structures concern hierarchical data, and principled reasoning about them.

Induction and recursion are the mathematician’s and the computer scientist’s dual ways of talking about the same thing. For example, the use of the inductive hypothesis in an induction proof is analogous to a recursively defined procedure call in a program.

Numbers are an obvious thing for a computer scientist to understand. We’ll look at some elementary number theory and its use in the RSA algorithm, a foundation of cryptographic protocols used on the Internet.

Graphs are a ubiquitous data structure. They encode binary relations in math.

Counting permutations and combinations is a foundation for understanding how many computer algorithms work.

Probability is an equally useful tool for analyzing the behavior of algorithms.
What work will be required?

There will be weekly problem sets, midterm examinations on Wednesday, October 14 and Monday, November 16 in the evening, and a final examination tentatively scheduled for Friday, December 18 from 9.15am–12.15pm. I will not allow incompletes for students unable to make exams due to last-minute travel or other logistics constraints. If you have a problem, let me know within the first two weeks of class. On the other hand, I’ll do whatever your academic dean tells me to do.

I encourage you to discuss the problem sets with your colleagues in the class, and to work collaboratively, particularly on difficult problems. Such collaboration must be mentioned explicitly in handed-in solutions, and each student is responsible for writing up and handing in individual problem sets: no “group copies” will be accepted. (See Grading and Homework below.)

Laptop and cellphone policy

You may not have an open laptop in class, unless you have the instructor’s permission.

You may not check text messages during class, unless you have the instructor’s permission. You risk dismissal from the day’s lecture for texting.

Outdoor clothing policy, and slouching rule

You may not wear your hat and coat in class, unless you have the instructor’s permission.

You need to sit up straight in your seat, with your feet placed flat on the floor. (Actually, this is more of a guideline than a rule…)

Note: The outdoor clothing policy and slouching rule are designed to keep you from falling asleep during lecture…

Grading and homework policy

In computing your grade, I expect to take the best of the following: either (1) final [100 percent]; (2) midterms [50 percent], final [50 percent]; or (3) homework [20 percent], midterms [40 percent], final [40 percent]. Everything is scaled: you are ranked in each category, and the ranks are weighed using the percentages. Recall the downside of scaled grades: when two friends are jumped by a bear, one friend says, “I don’t have to outrun the bear—I just have to outrun you!” So if you don’t want to do homework, it’s OK. And if you do badly on a midterm, but can recover on the final, it’s OK. But the only way you will learn enough to solve the exam problems is to do the homework!

I again encourage you to discuss homework problems with your colleagues in the class, and to work together on their solution. This way, you can help teach each other, and correct each other’s crazy ideas.

Submission protocol: Homeworks on multiple pieces of paper must be stapled together! Write neatly. Think about what it must be like to read your own writing!

Homework: Homework will be assigned and collected on Thursdays, in class. Of the 11 homework assignments, the two lowest scores will be discarded. Late homework will not be accepted, except if there are genuine personal difficulties or medical emergencies.

Honor code: Cheating is a very serious business and will not be tolerated at all. We will make every attempt to be reasonable about assignments, due dates, etc., but infractions of the honor code will be dealt with severely.

Responsibility Clause: You are bound by the honor code even if you were not aware of its details.

Responsibility for Responsibility Clause: You are bound by the Responsibility Clause even if you were not aware of its details.

…and so on…
Tentative syllabus
39 meetings overall.

ADMINISTRIVIA AND INTRODUCTION [2 lectures]


August 31: Difference engine (concluded).

PROOFS [3 lectures]

September 2: 1.1 Propositions • 1.2 Predicates • 1.3 The Axiomatic Method • 1.4 Our Axioms • 1.5 Proving an Implication

September 3: 1.6 Proving an “If and Only If” • 1.7 Proof by Cases • 1.8 Proof by Contradiction • 1.9 Good Proofs in Practice

September 9: 2.1 Well Ordering Proofs • 2.2 Template for Well Ordering Proofs • 2.3 Factoring into Primes • 2.4 Well Ordered Sets

LOGIC [3 lectures]

September 10: 3.1 Propositions from Propositions • 3.2 Propositional Logic in Computer Programs • 3.3 Equivalence and Validity

September 16: 3.4 The Algebra of Propositions • 3.5 The SAT Problem • 2-SAT Can be Solved Easily, but 3-SAT Cannot

September 17: 3.6 Predicate Formulas • Interpreting Quantifiers by Games

SETS AND STRUCTURES [3 lectures]

September 21: 4.1 Sets • 4.2 Sequences • 4.3 Functions

September 24: 4.4 Binary Relations • 4.5 Finite Cardinality • 7.1 Infinite Cardinality

September 21: 7.2 The Halting Problem • 7.3 The Logic of Sets • 7.4 Does All This Really Work?

INDUCTION AND RECURSION [5 lectures]

September 29: 5.1 Recursive Definitions and Structural Induction • 5.4 State Machines

September 30: (Tuesday) 5.2 Strong Induction • 5.3 Strong Induction vs. Induction vs. Well Ordering

October 7: 6.1 Recursive Definitions and Structural Induction • 6.2 Strings of Matched Brackets

October 8: 6.3 Recursive Functions on Nonnegative Integers • 6.4 Arithmetic Expressions • 6.5 Induction in Computer Science

October 12: Fibonacci numbers and generating functions.

FIRST MIDTERM [1 lecture]

October 14: Review for first midterm. The first midterm will be that evening, time and place to be arranged.
Numbers [5 lectures]

October 15: 8.1 Divisibility • 8.2 The Greatest Common Divisor (and the Pulverizer) • 8.4 The Fundamental Theorem of Arithmetic

October 19: 8.5 Alan Turing • 8.6 Modular Arithmetic • 8.7 Remainder Arithmetic

October 21: 8.9 Multiplicative Inverses and Cancelling • 8.10 Euler’s Theorem

October 22: 8.11 The RSA Algorithm

October 26: 8.11 The RSA Algorithm (continued) • 8.12 What has SAT got to do with it?

Graphs [5 lectures]

October 28: 9.1 Vertex Degrees • 9.2 Walks and Paths • 9.3 Adjacency Matrices

October 29: 11.1 Vertex Adjacency and Degrees • 11.2 Sexual Demographics in America

November 2: 11.5 Bipartite Graphs and Maximum Matchings: Hall’s Theorem

November 4: Matching in bipartite graphs: Augmenting paths • 11.6 The Stable Matching Problem • 11.9 Connectivity

November 5: Paths: Transitive Closure and Matrix Multiplication • Page Ranking on the Internet

Counting: Sums and Asymptotics [3 lectures]

November 9: 13.1 The Value of an Annuity • 13.2 Sums of Powers

November 11: 13.3 Approximating Sums • 13.5 Products

November 12: 13.7 Asymptotic Notation

Second midterm [1 lecture]

November 16: Review for second midterm. The second midterm will be that evening, time and place to be arranged.

Counting: Cardinality Rules [4 lectures]

November 18: 14.1 Counting One Thing by Counting Another • 14.2 Counting Sequences • 14.3 The Generalized Product Rule • 14.4 The Division Rule

November 19: 14.5 Counting Subsets • 14.6 Sequences with Repetitions • 14.6.3 The Binomial Theorem

November 23: 14.8 The Pigeonhole Principle (including the Greatest Card Trick in the World) • 14.10 Combinatorial Proofs

November 30: 14.9 The Principle of Inclusion and Exclusion • Euler’s Function \( \phi(n) \) (where \( n \) is a product of distinct primes) • The Hat Check Problem

Probability [4 lectures]

December 2: 16.1 Let’s Make a Deal • 16.2 The Four Step Method • 16.4 The Birthday Principle

December 3: 17.1 Monty Hall Confusion 17.2 Set Theory and Probability • 17.3 The Four Step Method for Conditional Probability

December 7: 18.1 Random Variable Examples • 18.2 Independence

December 9: 18.5 Linearity of Expectation • The Buffon Needle Problem