

CS103A

Discrete Mathematics for Computer Science

Instructor: Bob Plummer plummer@cs.stanford.edu

TAs: Brian Eggleston briane@stanford.edu
 Tyler Mullen tmullen@stanford.edu
 Sudeep Tandon sudeept@stanford.edu
 Tiyu Wang tiyuwang@gmail.com

Class Website: cs103a.stanford.edu

Questions for today

What is Discrete Mathematics?

Why study it?

How will the course be run?

How is the CS curriculum changing?

Discrete:
 separate, distinct;
 the opposite of continuous

Discrete Mathematics:
 A collection of mathematical topics that
 examine and use finite or countably infinite
 mathematical objects.

Discrete Mathematics

Formal logic and proof techniques

Number theory—properties of integers

Combinatorics—problems of counting

Analysis of algorithms

Discrete structures—sets, relations, trees, graphs

Computability and decidability

Discrete Mathematics

CS103A	Formal logic and proof techniques Number theory—properties of integers Combinatorics—problems of counting	} More proof techniques
CS103B	Analysis of algorithms Discrete structures—sets, relations, trees, graphs Computability and decidability	

Why?

"The mathematician is an artist
 whose medium is the mind and
 whose creations are ideas."
 -- H.S. Wall

Email Hotline: cs103a@gmail.com

Pre/Co-requisite: [CS106A](#)

Working Office Hours: [Location and schedule TBA](#)

Problem Sets: [Due in class \(schedule in Syllabus\)](#)
 Not accepted more than one class late
 10% penalty if late
 3 free "late days"

Textbook:

[Language Proof and Logic](#)
[Barwise & Etchemendy](#)

Software Tools/Used Books

If you would like a text for the second half of the course, a good supplement to the notes is:

[Discrete Mathematics and Its Applications, 6th Ed.](#)
[Kenneth H. Rosen](#)

Several books are on reserve at the Math/CS Library

Exams

Midterm: [Thurs., Oct. 23, 7 - 9 pm](#)

Final: [Wed., Dec. 10, 8:30 - 11:30 am](#)

Grading

45% Problem sets
 20% Midterm
 35% Final

You must have a passing average on problem sets and on exams to pass the course.

Honor Code

Current Curriculum: Lower Division

AP credit or strong prior programming experience

Programming
 Programming methodology and abstraction
 Programming paradigms
 Object-oriented systems design

Theory
 Discrete mathematics
 Discrete structures
 Automata theory
 Design and analysis of algorithms

```

    graph LR
        AP[AP credit or strong prior programming experience] --> CS106A
        CS106A --> CS106B_X[CS106B/X]
        CS106B_X --> CS107
        CS107 --> CS108
        CS103A --> CS103B
        CS103A --> CS103X
        CS103B --> CS154
        CS103X --> CS154
        CS154 --> CS161
    
```

New Curriculum: Introductory Core

AP credit or strong prior programming experience

CS Major

"Funnel"

CS106A

Systems I (CS106B/X) Theory I (CS103)

Systems II (CS107) Theory II (CS109)

Systems III (CS110) Theory III (CS161)

Courses at appropriate level of difficulty to mature from introductory to depth material in a reasonable manner

```

    graph TD
        CS106A --> SystemsI[Systems I CS106B/X]
        CS106A --> TheoryI[Theory I CS103]
        SystemsI --> SystemsII[Systems II CS107]
        SystemsII --> SystemsIII[Systems III CS110]
        TheoryI --> TheoryII[Theory II CS109]
        TheoryII --> TheoryIII[Theory III CS161]
        SystemsIII --> TheoryIII
    
```

Course Transition Plan in 2008-09

Fall	Winter	Spring
CS106X (Systems I)	CS106B (Systems I)	CS106B (Systems I)
CS107 (last old version)		CS107 (new Systems II) CS110 (Systems III)
CS103A	CS103B	CS154
	CS103 (Theory I)	CS109 (Theory II)
CS161 (current)	CS161 (current)	

- Existing CS103A/B sequence offered one last time
- Students having taken old CS107 can take CS110
- Content of CS161 will transition (slightly) in 2009-10