

COMPSCI 121

Introduction to Theoretical Computer Science

Term/Year: Fall 2018
Department: Computer Science

Enrollment: 216
Number of Responses: 191
Percent Response 88.43%

Unless otherwise indicated in the question text, the following scale is used for responses:
 1=unsatisfactory; 2=fair; 3=good; 4=very good; 5=excellent.

GENERAL QUESTIONS

	na	1	2	3	4	5	Tot.	Response Rate	Mean
Evaluate the course overall.		22	22	56	62	27	189	87.50%	3.26
Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.)	0	24	37	51	36	38	186	86.11%	3.15
Assignments (exams, essays, problem sets, language homework, etc.)	0	7	25	47	67	40	186	86.11%	3.58
Feedback you received on work you produced in this course	2	16	33	45	55	33	182	84.26%	3.31
Section component of the course	76	5	18	32	25	22	102	47.22%	3.40
On average, how many hours per week did you spend on coursework outside of class? (1=<3; 2=3-6; 3=7-10; 4=11-14; 5=>14)		1	32	76	28	46	183	84.72%	11.78
How difficult did you find this course? (1=very easy; 2=easy; 3=moderate; 4=difficult; 5=very difficult)		2	19	51	66	48	186	86.11%	3.75
What was/were your reason(s) for enrolling in this course? (Please check all that apply)	Elective						18	8.33%	
	Concentration or Department Requirement						172	79.63%	
	Secondary Field or Language Citation Requirement						8	3.70%	
	Undergraduate Core or General Education Requirement						0		
	Expository Writing Requirement						2	0.93%	
	Foreign Language Requirement						1	0.46%	
	Pre-Med Requirement						1	0.46%	
How strongly would you recommend this course to your peers? (1=definitely not recommend; 2=unlikely to recommend; 3=recommend with reservations; 4=likely to recommend; 5=recommend with enthusiasm)		17	26	59	62	22	186	86.11%	3.25



EVALUATION OF INSTRUCTORS

Barak, Boaz

	na	1	2	3	4	5	Tot.	Response Rate	Mean
Evaluate your Instructor overall.		14	25	50	51	44	184	85.19%	3.47
Gives effective lectures or presentations, if applicable	3	24	36	44	49	28	181	83.80%	3.12
Is accessible outside of class (including after class, office hours, e-mail, etc.)	22	9	18	30	47	56	160	74.07%	3.77
Generates enthusiasm for the subject matter	1	11	19	34	47	72	183	84.72%	3.82
Facilitates discussion and encourages participation	54	4	12	28	28	46	118	54.63%	3.85
Gives useful feedback on assignments	100	7	9	16	20	19	71	32.87%	3.49
Returns assignments in a timely fashion	84	2	1	13	25	45	86	39.81%	4.28

Unless indicated in the question text, the following scale is used for responses: 1=unsatisfactory; 2=fair; 3=good; 4=very good; 5=excellent.

EVALUATION OF TEACHING FELLOWS

Teaching Fellows Overall

	na	1	2	3	4	5	Tot.	Response Rate	Mean
Evaluate your Section Leader overall.	0	2	2	5	12	92	113	52.31%	4.68
Gives effective lectures or presentations, if applicable	33	2	0	5	8	63	78	36.11%	4.67
Facilitates discussion and encourages participation	17	2	3	7	11	71	94	43.52%	4.55
Is accessible outside of class (including after class, office hours, e-mail, etc.)	8	2	0	7	8	88	105	48.61%	4.71
Generates enthusiasm for the subject matter	9	3	1	10	14	73	101	46.76%	4.51
Gives useful feedback on assignments	45	2	0	6	4	55	67	31.02%	4.64
Returns assignments in a timely fashion	45	1	0	6	6	52	65	30.09%	4.66



COMPSCI 121
Introduction to Theoretical Computer Science

Comments

What were the strengths of this course? Please be specific and use concrete examples where possible.

Course

Evaluate the course overall.: **3 (good)**

In the end, I did feel like I had a grasp of the material. Piazza was super helpful in this course as staff and Professor Barak consistently responded very quickly.

Evaluate the course overall.: **1 (unsatisfactory)**

Problem sets helped clarify some concepts sometimes

Evaluate the course overall.: **4 (very good)**

It forced me to really learn the material in order to do anything.

Evaluate the course overall.: **4 (very good)**

Overall an interesting class, and I think the material is useful knowledge for every CS concentrator to have

Evaluate the course overall.: **3 (good)**

Wide range of interesting material covered in a novel way.

Evaluate the course overall.: **3 (good)**

The material in the second half of the course- cryptography, quantum computing, algorithms, reductions, etc.

Evaluate the course overall.: **3 (good)**

Interesting content, interesting psets.

Evaluate the course overall.: **5 (excellent)**

Being someone who's really interested in theoretical computer science, this was a great class for me. The approach that Boaz took to teaching the material was really novel; for instance, the choice to use circuits instead of Turing machines for the most part enabled some really slick arguments and demonstrations. Kudos to you all for the effort in developing all the materials we used.

Evaluate the course overall.: **2 (fair)**

Interesting material.



Evaluate the course overall.: **4 (very good)**

The course was interesting and challenging yet acceptable.

Evaluate the course overall.: **3 (good)**

You're going to learn a lot - but you might die.

Evaluate the course overall.: **3 (good)**

It was definitely some interesting material, which really did improve my understanding of theoretical computer science.

Evaluate the course overall.: **3 (good)**

The textbook is a great source.

Evaluate the course overall.: **2 (fair)**

This course gives a great broad introduction to theoretical CS topics

Evaluate the course overall.: **5 (excellent)**

The content of this course was amazing -- it was a great introduction to TCS and I'm proud that I'm able to do proofs by reduction through the practice on the psets.

Evaluate the course overall.: **4 (very good)**

Comprehensible course

Evaluate the course overall.: **3 (good)**

Interesting content and challenging. Great teaching staff and Boaz really cared about his students.

Evaluate the course overall.: **3 (good)**

Covers the basics of CS theory

Evaluate the course overall.: **2 (fair)**

The material is good; everything else has room for improvement. Lectures introduce new topics so fast that the only way to vaguely keep up is by reading very long textbook chapters that often don't help very much. As a mandatory CS class, CS121 should be better thought through.

Evaluate the course overall.: **3 (good)**

Problem sets were a great method of learning the courses material. Sections were helpful.

Evaluate the course overall.: **4 (very good)**

Interesting material. TFs were amazing. Partner psets were manageable and encouraged learning.

Evaluate the course overall.: **3 (good)**

The material and proofs are challenging and important fundamental concepts, but the presentation of that material can definitely be improved. The Piazza forum was definitely a strength, though -- I appreciated how hard Prof. Barak and the TFs worked to answer our questions quickly and thoroughly.

Evaluate the course overall.: **2 (fair)**

Strengths were Boaz's kindness, all the bonus points, and Eric the TF.

Evaluate the course overall.: **4 (very good)**

- The course material is really really interesting once you understand it! - Everything is recorded - Boaz is very open about what you need to know, i.e. how to prepare for the class / top theorems for the midterm / review session for all the tests - Piazza is so helpful and the instructors are so diligent about answering questions on it. Course staff were also all so kind and really tried to support people

Evaluate the course overall.: **3 (good)**

Interesting material

Evaluate the course overall.: **1 (unsatisfactory)**

there really weren't any... i came out of this course knowing how to do reductions, but why do i need to know how to do them?

Evaluate the course overall.: **4 (very good)**

Interesting content. I liked the rigorous proofs, such as that of the uncomputability of the Halting Problem, as well as the more abstract discussions, such as of the philosophy of $P=NP$.

Evaluate the course overall.: **1 (unsatisfactory)**

The material itself is rather interesting and I particularly liked the sections on NP reductions, and reductions in general. I also particularly enjoyed the first programming assignment, where we wanted to create a NAND program for squaring.

Evaluate the course overall.: **3 (good)**

Boaz was enthusiastic and cared about us and tried to make the course as fun as possible. Liked the format of psets.

Evaluate the course overall.: **4 (very good)**

Really interesting content, and I learned a lot from the problem sets and from section. Made me realize theory is super cool and applicable, as I got to combine my interest in math with programming.



Evaluate the course overall.: **3 (good)**

Problem sets and office hours were good.

Evaluate the course overall.: **3 (good)**

I think the professors and TFs were really dedicated

Evaluate the course overall.: **3 (good)**

Taught a lot of useful and relevant content about theoretical CS. Covered a lot of ground.

Evaluate the course overall.: **3 (good)**

The concept of reductions, along with their proofs, were very interesting.

Evaluate the course overall.: **3 (good)**

The course content is quite interesting, even if it will never show up on a technical interview. Also, some of the TFs (like Eric Lu) are very dedicated to helping students do well in the course.

Evaluate the course overall.: **3 (good)**

Gave a good basic understanding of theoretical CS. Had a lot of resources for those struggling.

Evaluate the course overall.: **3 (good)**

Interesting material at times. Psets are thought provoking.

Evaluate the course overall.: **1 (unsatisfactory)**

Taught you a lot about computer science theory.

Evaluate the course overall.: **5 (excellent)**

Boaz is a badazz, and his approach to theoretical CS is I think uniquely cool. Also his textbook is great. I went from never having done a proof before to doing them comfortably for lots of applications.

Evaluate the course overall.: **4 (very good)**

The material was very fascinating, and the approach of the class is very rigorous, which is very good for a theory class.

Evaluate the course overall.: **2 (fair)**

Very comprehensive, in-depth overview of theoretical computer science topics.



Evaluate the course overall.: **4 (very good)**

Helping students understand computer theory from the fundamental building blocks of computing

Evaluate the course overall.: **4 (very good)**

This course managed to cover a wide range of material in reasonable depth. The readings were amazing and really educational. Problem sets, in general, were interesting and puzzling.

Evaluate the course overall.: **4 (very good)**

It was difficult for me personally, but I felt it was a fair course which I really appreciated.

Evaluate the course overall.: **4 (very good)**

Takes on very interesting material.

Evaluate the course overall.: **2 (fair)**

breadth of material covered was good--not a lot of intro theory courses touch upon zero-knowledge proofs or cryptography, so these aspects were nice. CFG/regex also interesting

Evaluate the course overall.: **4 (very good)**

The strengths of this course is the material itself, which is i found incredibly engaging, interesting, and fundamental to math and cs. Resources like piazza, the textbook, and section/office hours (which I didn't attend) also seemed great.

Evaluate the course overall.: **3 (good)**

The strengths of this course were all the options offered to get help. The recorded sections with Hikari were amazing and the TFs (especially ERIC) were so helpful. I couldn't have completed this class without their help!

Evaluate the course overall.: **4 (very good)**

Great homework and exam problems to help reinforce material.

Evaluate the course overall.: **5 (excellent)**

Really

Evaluate the course overall.: **1 (unsatisfactory)**

hmmm



Evaluate the course overall.: **5 (excellent)**

Solid covering of theory with a rigorous math background.

Evaluate the course overall.: **1 (unsatisfactory)**

Broad and informative survey of concepts surrounding theoretical computing.

Evaluate the course overall.: **5 (excellent)**

The book is excellent; the problem sets are actually fun; the TFs and Boaz are amazing, knowledgeable and helpful (they always gave detailed answers to my piazza questions).

Evaluate the course overall.: **5 (excellent)**

The course materials and problem sets (see question 2 comments for more detail).

Evaluate the course overall.: **5 (excellent)**

The textbook is a great (and free!) resource. Barak understands the course material ten times over, and can answer students' questions well. Also, I am a fan of the flipped classroom model; it is a more strenuous way to learn, and that's why it's more effective.

Evaluate the course overall.: **3 (good)**

Despite its flaws, the textbook was rather complete and provided sufficient background for the lectures. By being introduced to the material three times (once in the reading, once in the lecture, and once in the psets), I felt that I efficiently learned the material.

Evaluate the course overall.: **3 (good)**

I think the course gets the job done. It gives a broad understanding of very important concepts in theoretical computer science, such as uncomputability, universality, complexity classes, etc. The workload, homework assignments, and exams were very reasonable.

Evaluate the course overall.: **2 (fair)**

It covers a lot of material

Evaluate the course overall.: **5 (excellent)**

This course gave me a thorough understanding of the material by building it up from the very basics (NAND), showing how the different topics were connected, and making sure we understood what the definitions meant and why the theorems were true.

Evaluate the course overall.: **1 (unsatisfactory)**

The only strength of its course was that the material is fundamental to computer science, and that Boaz is a really nice guy. The problem sets were generally fair but occasionally burdensome and uninteresting.



Evaluate the course overall.: **3 (good)**

This course is a good survey of issues of complexity theory and uncomputability. The text, written by Prof. Barak, was very helpful, and obviously followed the course nicely.

Evaluate the course overall.: **3 (good)**

Everything was fair, and nothing was unexpected or unfairly difficult on exams and problem sets.

Evaluate the course overall.: **4 (very good)**

Boaz is a very enthusiastic teacher, and I liked the material that the course covered.

Evaluate the course overall.: **4 (very good)**

P-Sets are good (even if the early ones are a bit annoyingly trivial). Textbook is good. Boaz obviously cares a lot about the course. Almost all the flak the course gets on being too hard is due to Harvard CS's failure to make its students take proof-based math courses, not any failing on Boaz's part.

Evaluate the course overall.: **4 (very good)**

Had a good TF support and lots of office hours where students could get help.

Evaluate the course overall.: **1 (unsatisfactory)**

Material is interesting.

Evaluate the course overall.: **3 (good)**

Good material and problem sets.

Evaluate the course overall.: **4 (very good)**

This course did a good job at providing a support network for students. I always felt like when exam time came around I had plenty of materials available to study and I had a good idea what I was expected to know.

Evaluate the course overall.: **4 (very good)**

I really appreciated the frequency of office hours, quick piazza responses, and general support throughout the course.

Evaluate the course overall.: **1 (unsatisfactory)**

lots of theory learned! cool subject



Evaluate the course overall.: **4 (very good)**

The topics were really interesting, and I enjoyed the problem sets.

Evaluate the course overall.: **3 (good)**

Interesting material

Evaluate the course overall.: **1 (unsatisfactory)**

Really didn't like this course so I'm not sure I have a great answer here

Evaluate the course overall.: **3 (good)**

The course went over a lot of interesting material that is relevant to my future studies of Computer Science.

Evaluate the course overall.: **4 (very good)**

You can tell Boaz cares about the material and having a strong support network for students who may be struggling at times with plentiful office hours and active piazza boards.

Evaluate the course overall.: **4 (very good)**

The subject material was very interesting and presented in a way that showed its impact in the world (although it would have been nice to say that we were using NAND to build up to Turing Machines because it seemed kind of arbitrary). Course staff and problem sets did a good job of making reductions as understandable as possible when the topic is so confusing. Also Boaz is funny.

Evaluate the course overall.: **3 (good)**

Definitely does a good job of preparing students to write and read mathematical proofs. I also think that the homework helped students think about the classes concepts in new and creative ways.

Evaluate the course overall.: **4 (very good)**

The big results, like proving that 3SAT is in NP, that HALT was uncomputable, and consequences of if P=NP were very interesting, even outside of this course.

Evaluate the course overall.: **5 (excellent)**

The content selection is just so incredible. Best content curation of any class I have taken so far at harvard (except for maybe sls20; that shit was good too)

Evaluate the course overall.: **4 (very good)**

Theoretical CS is such an interesting topic. Prof. Barak really cares about student success and it shows through the course. The workload in general is really manageable and the course does a good job in covering a large amount of material.



Evaluate the course overall.: **3 (good)**

- The material is designed in a way that makes you feel like you are actually learning something. - Poll Everywhere questions in lectures were great (even though I watched recorded lectures). - Nice extra credit problems everywhere. - Good jokes that made us remember a lot. - Appreciate all of the effort put by Boaz. I think that in a couple of years the course can be much much better if Boaz is still so motivated to improve it.

Evaluate the course overall.: **4 (very good)**

Very solid progression of topics through the first half of class. Built up very nicely.

Evaluate the course overall.: **3 (good)**

The material is very strong and the textbook provides lots of information while challenging readers. This class does not hold a lot back in an attempt to make things easier to understand which was a positive, but also meant that strong teaching would be necessary.

Evaluate the course overall.: **4 (very good)**

The problem sets and written materials do a really good job of reinforcing understanding.

Evaluate the course overall.: **4 (very good)**

It taught some interesting theoretical material in regards to Computer Science.

Evaluate the course overall.: **5 (excellent)**

Content

Evaluate the course overall.: **3 (good)**

availability of office hours

Evaluate the course overall.: **1 (unsatisfactory)**

none.

Evaluate the course overall.: **4 (very good)**

Interesting material, well structured, lots of resources, reading quizzes actually made you do the reading

Evaluate the course overall.: **2 (fair)**

Interesting Material



Evaluate the course overall.: **5 (excellent)**

Boaz knows his stuff! The materials are exciting and Prof. Barak knows what he's talking about. Once you get into it, it gets super fun :)

Evaluate the course overall.: **1 (unsatisfactory)**

The tfs are all very nic people

Evaluate the course overall.: **3 (good)**

Learned new and interesting concepts, new ways of thinking

Evaluate the course overall.: **4 (very good)**

I learned about very theoretical things

Evaluate the course overall.: **4 (very good)**

I thought that the textbook aided in learning the material and that having student do readings prior to lecture helped learn the material. In addition, I think having Piazza was key to helping the students navigate difficult material.

Evaluate the course overall.: **1 (unsatisfactory)**

This class is hard to like, but if you can manage to see through the cloud of confusion and mathematical jargon, some of the concepts discussed in this class are actually really interesting and fascinating, such as Turing-completeness, uncomputable functions, and the P vs NP problem.

Evaluate the course overall.: **1 (unsatisfactory)**

The course teaches a number of very interesting topics and covers a wide-ground of relevant theory topics in modern CS.

Evaluate the course overall.: **4 (very good)**

Gives good theoretical cs foundation, interesting material surrounding computability, function, and programs.

Evaluate the course overall.: **4 (very good)**

Interesting Content

Evaluate the course overall.: **3 (good)**

Thought it was interesting material



Evaluate the course overall.: **2 (fair)**

active piazza and you could tell boaz wanted to help

Evaluate the course overall.: **3 (good)**

I loved the middle part of the material in this course (uncomputability and P vs NP).

Evaluate the course overall.: **2 (fair)**

Good support and the teachers / teaching fellows really care about your academic career.

Evaluate the course overall.: **4 (very good)**

The material was excellent and very interesting!

Evaluate the course overall.: **5 (excellent)**

I found the material super dry but the course staff definitely made learning it as intriguing as possible.

Evaluate the course overall.: **4 (very good)**

(1) Interesting material (2) Boaz is a great lecturer and a good teacher!

Evaluate the course overall.: **3 (good)**

The strengths of this course were that the exams were well-designed and Professor Barak really cared about the course and the students

Evaluate the course overall.: **1 (unsatisfactory)**

Very interesting material

Evaluate the course overall.: **4 (very good)**

I think the results proven in this course are extremely interesting.

Evaluate the course overall.: **1 (unsatisfactory)**

None, I hated this course, stressed me out, and not sure what I learned.

Evaluate the course overall.: **1 (unsatisfactory)**

Interesting material, and some of the section leaders put in a lot of time and effort making sure the students might actually get what's going on.



Evaluate the course overall.: **1 (unsatisfactory)**

None.

Evaluate the course overall.: **3 (good)**

Very interesting subject-matter and a clear and coherent concept on the part of the instructor on how to teach it.

Evaluate the course overall.: **4 (very good)**

Very good introduction to theoretical concepts in Computer science regarding computability, P vs NP, and others.

Evaluate the course overall.: **2 (fair)**

There were office hours every day, a very fast response rate on Piazza, Professor Barak is very friendly and approachable.

Evaluate the course overall.: **4 (very good)**

I enjoyed the subject matter a lot

How could this course be improved? Please use concrete examples where possible and provide constructive suggestions.

Course

Evaluate the course overall.: **3 (good)**

The textbook for this class, for someone with not much math background, was particularly hard to read and understand. Additionally, lecture did not do much to help with parsing the textbook as we were expected to grasp all the details from the textbook itself and weren't given much support if we found it difficult to do so. Lecture was oftentimes also quite confusing, and there wasn't much support for feeling lost but not knowing what questions to ask to begin. Overall, I found a lot of the course materials really inconsistent (some problems/examples were extremely difficult while others seemed too easy) and it was difficult to know which content was actually important and which were small details.

Evaluate the course overall.: **1 (unsatisfactory)**

Better lectures that actually go over example problems and teach the material. Better teaching PLEASE

Evaluate the course overall.: **4 (very good)**

The guidelines detailing student interaction with psets could be lessened some, allowing for more open collaboration.

Evaluate the course overall.: **1 (unsatisfactory)**

Course not very organized

Evaluate the course overall.: **4 (very good)**

Not nearly enough sections. Maybe have two sections be recorded instead of one? That would have been some good extra review, because I couldn't make it to any of the sections. Hopefully there will be more review materials for the midterms and final next year.

Evaluate the course overall.: **3 (good)**

Varying depth and difficulty between topics made for semi-inconsistent problem set experience. Also, novel approach (aka. NAND) covered a severe cost of being unfamiliar with more "traditional" models used everywhere else (e.g. in papers)

Evaluate the course overall.: **3 (good)**

The lectures could be honed in and move at a slower pace so students will actually gain from them.

Evaluate the course overall.: **3 (good)**

The way the content was presented during lecture, and in the textbook.

Evaluate the course overall.: **2 (fair)**

It would be extremely helpful to have more practice resources made available and in an organized format.

Evaluate the course overall.: **4 (very good)**

Textbook readings could have more "plain English" summaries against which we could check our understanding of the technical descriptions, because I later realized I had significantly misunderstood some theorems and proofs and descriptions.

Evaluate the course overall.: **3 (good)**

Needs more resources, the jump from CS20 to 121 is brutal and the only workaround seems to be spending 12 hours a week in OH.

Evaluate the course overall.: **3 (good)**

It was often hard to follow, and there is no support system in the class. There are no answer keys and limited practice material. The class was made much harder than it needed to be by the total lack of support the students received.

Evaluate the course overall.: **3 (good)**

Lectures were pretty boring and repetitive of the textbook.

Evaluate the course overall.: **2 (fair)**

I found lectures to be unhelpful. Even if I had done the readings, the examples Boaz went over in class went completely over my head. I stopped going to lecture after the 4th week and just read the textbook which was fine.

Evaluate the course overall.: **5 (excellent)**

The lectures were sometimes confusing, so that I would understand the material from the reading but then get confused after leaving lecture. I think it would be great to work with the Bok center to get feedback on lecturing!

Evaluate the course overall.: **4 (very good)**

Pretty reliant on an understanding of proof based mathematics, even having taken CS20 wasn't always sure about how to structure a proof as there weren't very many examples

Evaluate the course overall.: **3 (good)**

Lectures and readings are overly confusing - some very simple ideas are explained in extremely bizarre ways. Lecture is not that helpful as it is explaining the reading in just as confusing a way.

Evaluate the course overall.: **3 (good)**

Topics are approached in a not traditional sometimes inconvenient manner that seems

Evaluate the course overall.: **2 (fair)**

Less material covered - more sections that are smaller.

Evaluate the course overall.: **3 (good)**

The readings were very dense at times and confusing. Having a specific outline with important concepts would improve this. A reading guide that you fill in as you read may be more effective than the quizzes. Lectures were fast at times and I didn't know what questions to ask even when I was confused. I did not feel like I had enough time for either midterm. I think a 24-hour take home midterm would be a way better measure and opportunity to learn for this course.

Evaluate the course overall.: **4 (very good)**

NAND and NAND++ should be equated with FSMs and Turing Machines more clearly. In talking with other people about theoretical CS, it seems like we don't know anything but really the language and terms we use are just totally off from what is used in real CS.

Evaluate the course overall.: **3 (good)**

I found the textbook, lectures, and review materials very confusing (and I know others did, too). I often didn't know where to turn when I didn't understand something -- the review packet for the second midterm, for example, contained several problems that even the TFs didn't know how to do. I also felt like I didn't have access to enough practice problems, so I couldn't test my own understanding -- we were instructed to go to the Sipser textbook when we wanted extra practice, but some guidance would have been really helpful. Also, I know the textbook is difficult for everyone, but sometimes I felt that the sections and lectures didn't explain the trickiest concepts from the textbook that I didn't understand.

Evaluate the course overall.: **2 (fair)**

I didn't understand a lot of the lectures and thought sometimes the course went into depth on proofs that weren't relevant.

Evaluate the course overall.: **4 (very good)**

- It's a hard class if you don't have a super strong math background. There is sometimes a divide (inevitable) between the extremely strong math people and the not super strong math people, for whom this might be their first proof class - Textbook was confusing sometimes and very dense, but also extremely helpful (provides a lot of examples, theorems, etc.) after I reread it several times and finally understood it. It would be helpful to provide excerpts from other textbooks (like Sipser) in the course materials as a possible alternative sometimes. - Releasing problem set solutions would be really helpful

Evaluate the course overall.: **1 (unsatisfactory)**

I have no clue why this course is required. If we learned something useful then maybe i would like the course and boaz as a teacher, but i just felt like 100% of the people in the class only took it because its a stupid required course.

Evaluate the course overall.: **4 (very good)**

A greater distinction between the elegant proofs, such as that of the uncomputability of the Halting Problem, and the uninteresting messy proofs, such as of the Turing Completeness of NAND++.



Evaluate the course overall.: **1 (unsatisfactory)**

The course materials are a weak point in the course design. Until the course text is in suitable condition, the course should be taught using other widely available and embraced texts. The course also lacked guidance on problems -- specifically, at the very least the staff should produce solutions for all problem sets.

Evaluate the course overall.: **1 (unsatisfactory)**

Unfortunately, the way that this class was taught left me rather unsatisfied with the material. Textbook readings were extremely dense and while that can be fine, there was no sense of practicality to the theorems. In the sense that, I would often think, "ok, this is true, but what does it REALLY mean?". For a mandatory class for all computer science students that makes it extremely difficult for many students to really appreciate the material itself. Furthermore, some of the proofs may be rigorous, but completely incomprehensible. For instance, the very first diagonalization argument was, while technically rigorous and correct, also the most convoluted explanation of the argument I had ever seen, and I had seen this proof times before (for instance in Math 23).

Evaluate the course overall.: **3 (good)**

Still seem a bit disorganized, course quizzes would be uploaded late, and there were multiple versions of the textbook going around.

Evaluate the course overall.: **4 (very good)**

Lecture was often really confusing, especially the proofs for major theorems we learned in class. I would really like to learn the reason why theorems are true, but they are usually presented in a weird/rushed way that is difficult to understand.

Evaluate the course overall.: **3 (good)**

I found lectures were often not very engaging and I also felt that the time pressure on the midterms didn't give me chance to truly express my knowledge.

Evaluate the course overall.: **2 (fair)**

very hard tests, no curve, no study tips

Evaluate the course overall.: **1 (unsatisfactory)**

The teaching in the class could be improved. The lectures shed little to no light on the information, and the only real way to understand is to spend massive amounts of time doing close readings of the textbook. The textbook could be very much simplified to be readable and easily digestible.

Evaluate the course overall.: **3 (good)**

Better teaching materials - in particular, a less verbose textbook. It was very difficult to parse, and often would just use mathematical symbols to describe concepts when it would have been very helpful to have a big-picture overview or example to give us an idea of what the concept is actually trying to say rather than parsing the math to understand it.



Evaluate the course overall.: **3 (good)**

This course needs to be less ambitious and cover less material. Lectures need to actually reinforce the reading and vice versa.

Evaluate the course overall.: **3 (good)**

As of the writing of this review, I would say that Boaz's decision to use NAND as the primary computational model is a net-negative decision. While it does lead to a more intuitive understanding of computation for the student (since it makes the focus of the course about computing functions as opposed to deciding languages), it also makes it harder for them to seek out additional resources, since most of those resources are centered around Turing Machines. This means that the student has to do the additional work of translating the findings regarding Turing Machines from the resources to the equivalent findings regarding NAND.

Evaluate the course overall.: **3 (good)**

Having a bit more introduction to proof writing.

Evaluate the course overall.: **3 (good)**

Lectures could be clearer.

Evaluate the course overall.: **5 (excellent)**

More complete textbook, less time wasted in lectures with polls and stuff (more math and productive stuff).

Evaluate the course overall.: **4 (very good)**

When describing proofs and whatnot, it would be better to also have some intuition so it is easier for the students to replicate the examples.

Evaluate the course overall.: **2 (fair)**

Too detailed for an intro-level course.

Evaluate the course overall.: **4 (very good)**

More visual examples in the text. It is a bit too word-dense

Evaluate the course overall.: **4 (very good)**

Lectures were sometimes boring and slow, especially because they repeated so much of the reading.

Evaluate the course overall.: **4 (very good)**

The textbook (although I understand it's a work in progress) was a bit confusing and filled with typos, which made reading more difficult and time-consuming.



Evaluate the course overall.: **4 (very good)**

Less focus on NAND.

Evaluate the course overall.: **1 (unsatisfactory)**

Boaz is not an effective lecturer, and the textbook presents the material in a very convoluted, indirect manner that makes it hard to learn the material by yourself (which is basically what is required of you in this class). I honestly would have preferred using a more traditional textbook like Sipser. I thought I was interested in theoretical CS, but this class has turned me away from pursuing classes that relate to TCS.

Evaluate the course overall.: **2 (fair)**

PLEASE for the love of GOD consolidate your information! if I want to find a file or a piece of information I have to check canvas, the course website, piazza, GitHub, that random google link at which slides were posted for a second, and the google calendar. this is absurd!

Evaluate the course overall.: **4 (very good)**

The actual lecture was generally pretty confusing. Boaz isn't the best at explaining mathematical concepts, and it was almost always unclear what was a tangent and what was actually needed for the psets and tests.

Evaluate the course overall.: **3 (good)**

I wish there were more sections just for practicing applying the material. For example, Hikari's section was excellent for going over material, but if there were another section to specifically apply the material, that would have helped tremendously.

Evaluate the course overall.: **4 (very good)**

Lecture notes and lectures sometimes miss the forest for the trees: a lot of the formalism makes significantly more sense once the main idea is grasped, but doesn't particularly help in understanding the main idea. It may help to have a few editors brush up on the lecture notes and fix typos/revise wording to make a few things clearer.

Evaluate the course overall.: **1 (unsatisfactory)**

there is no reason the course should be this hard, the material actually wasn't that difficult, but the teaching/lectures and psets were just not at all good.

Evaluate the course overall.: **5 (excellent)**

Not going to be super useful in industry or going forward at all



Evaluate the course overall.: **1 (unsatisfactory)**

1) The textbook could be improved to be more clear, less dense, with fewer confusing errors. Although most of the explanations or proofs tried to motivate the "idea of the proof," I found these to be too brief or quick to be helpful in really motivating the idea. I feel that this motivation is KEY to understanding the proof, since it provides a roadmap to the ensuing logic. This is particularly important for this textbook, because many of the proofs involve DENSE mathematical notation, which makes it very easy to get lost. (Although mathematical notation is of course unavoidable in a theory course, perhaps it may be minimized or laid out more clearly instead of in an onslaught of paragraphs?) Another idea to help with this in addition to the roadmap is to provide "stop and regroup" breaks in the middle of very long and involved proofs to very clearly summarize what has been shown and the big-picture route of how that has been accomplished, with a tie-in to what follows. The general goal should be to make the textbook much more readable and much more helpful as a companion to the course lectures. Consider taking example from Professor Blitzstein's STAT 110 book and lectures. 2) More efficient lectures. It seems that lectures were supposed to be a more detailed look at the textbook readings, but often the readings were dense and the lectures were just as dense. In this sense, it seemed that lectures were redundant to the book material. I would recommend making the lectures more like STAT 110's lectures, where the material is introduced and explained in lecture, and then supplemented with more detailed examples/proofs in the book. Reading quizzes are fine if they are used to emphasize material in this regard; but otherwise if they seek to nitpick small facts from the chapter, it can get annoying. I did not like the "discussion timer" used in lecture. I felt that this was a waste of valuable lecture time; I can discuss ideas with my peers at any time, but rarely can I hear course material explained by the professor. Furthermore, I felt that the times set were sometimes too long for the question (5 minutes for a trivial idea, for instance, or an idea that is much better explained by the professor), which gave me a further impression that I was wasting my time. In summary, I feel that, although it is valuable to discuss ideas with peers, this can be done outside of lecture; it is much more efficient to hear a clear explanation from the professor and then dissect this explanation with discussion outside of lecture. 3) Why NAND, NAND++, NAND<<?? Why not use the language of circuits, finite-state automata, and Turing machines that are universal to every other computer science theory course? I feel that much of the introductory time of the course was spent figuring out how to use NAND, when it could have been better spent understanding the relative computing power of the underlying models (circuits vs. finite-state automata vs. Turing machines). We even "admitted" that NAND machines corresponded to these, and yet we decided to go back to the veil of thinking in terms of NAND. This seemed like an unhelpful complexity that only clouded the understanding of the course material. Summary) Perhaps this course could be improved by taking example from CS 124 or STAT 110, which are similar theory-heavy courses but have managed to couple lectures with textbooks/lecture notes to very efficiently and engagingly teach the course material.

Evaluate the course overall.: **5 (excellent)**

Boaz is great and knowledgeable, but I really think he should restructure lecture. It really isn't helpful to just ask students to work out problems on the screen with each other. It's more useful to allocate all of lecture to explaining the difficult parts of readings and problem sets.

Evaluate the course overall.: **5 (excellent)**

See question 2 comments.

Evaluate the course overall.: **5 (excellent)**

A little less on NAND and more on Turing Machines---but Barak already knows this.

Evaluate the course overall.: **3 (good)**

I felt that the textbook just dumped too much code on us and used unnecessarily awkward notation; the uncountability of the reals proof was the first and most egregious example of this. Grading also felt inconsistent sometimes. For example, one time me and my partner accidentally submitted the same assignment independently rather than as a pair, but our grades differed by twenty points.

Evaluate the course overall.: **3 (good)**

For me, one of the weakest components of the course was the lectures and readings. A lot of the time, I felt like lecture wasn't really focusing on stuff that would help my understanding of the material. For example, although a crash course in 19th century breakthroughs in computer science, Python functions for generating NAND code/lambda calculus, and giant NAND circuits are all well and good, they won't help me grasp the material.

Evaluate the course overall.: **4 (very good)**

Okay -- I have a lot to say about this course. I loved it. But, there are a lot of issues still with this course. There should be a "translated to English" pdf or video series done by the TFs where every important definition and equation is translated to English with absolutely no mathematical symbols. The exams shouldn't be the one determinant of our grades. Everyone gets 100s on the psets and quizzes -- fine. But that final was ridiculously hard. There should never be a question on a final that gets 0/20. If a student puts down an answer they should at least get some credit. A standard deviation of 22 is ridiculous. The collaboration policy is so so so so so so so so so stupid. I get where you are coming from Boaz, I really do. I did a lot of the psets myself and it REALLY helped me understand the material. But you are missing a key point -- if you can't figure out an answer, you never end up understanding the material. People should submit their own psets, and there should be open collaboration. I am saying this because this will really help the students better learn the material. I guarantee if you switched to this format you would see a massive increase in how students do on the exams. Students told you this last year, I'm telling you again this year. We aren't stupid, we aren't saying this just to bump our grades in psets -- you already give so much bonus that's not the issue. I'm telling you this will help people better understand the material. THE TFs ARE NOT ENOUGH. You have to sit at OH for two hours to get under 2 minutes of help. That is not a good enough system. It doesn't work. I get that you have piazza but a lot of people just need things to be verbally explained to them. I am begging you on behalf of all future students who take this course and really just want to leave understanding the material, let us collaborate.

Evaluate the course overall.: **2 (fair)**

Realistically, there are a lot of ways this course could be improved, an especially important task given that the course is required for students to concentrate in computer science and has in many ways become the "weed-them-out" class. It is extremely math intensive, and even with CS20, many students are not adequately prepared for the mathematical background. The solution here is probably twofold: increase the rigor of CS20, and lower the level of CS121. At the very least there needs to be more communication between the two courses so that expectations are clear. The textbook is the next aspect where there is major room for improvement. To its credit, it is written in a fairly lighthearted manner and incorporates a good amount of humor. However, this does not make up for the fact that often it assumes a much higher level of understanding than most students have, and then as a result, leaves complicated topics and proofs as "exercises to the reader." For a student who has already not understood the very rough outline of a topic, it is very unlikely that he or she will be able to complete this "exercise," and so as a result, the concept is not well learned. There is nothing wrong with being explicit in the explanations and descriptions; one might even argue that this is the primary purpose of a textbook. There is room for improvement in lecture as well. At least in this iteration, there was very little additional learning that happened in lecture. If anything, lecture was in many ways very disheartening, for example when exercises would be presented, answered by one of the five same (brilliant) students, and then not explained. Exercises like these are probably best left for section (see the next paragraph), but if they must be included in lecture, they should certainly be explained. More than anything, lecture suffered from an at time insensitive use of language, labeling complex topics as "easy" or "not that hard," which merely has the effect of alienating or discouraging students who are struggling. A more conscious effort on the part of the professor to be mindful of language such as this would be a huge and much appreciated improvement. Finally, section was probably an underutilized yet invaluable tool. It might be a good idea to make section mandatory, perhaps at least until the first midterm, so that students are forced to make use of the resources given to them. Given that this is a gateway class to the cs concentration, taken primarily by sophomores (who really have only had two semesters of college under their belts), extra guidance can only be a benefit.

Evaluate the course overall.: **5 (excellent)**

I think the midterm exams could have been slightly shorter, although the final was a reasonable length.

Evaluate the course overall.: **1 (unsatisfactory)**

In almost every way. As a senior who studies computer science, I can confidently say this class offers little in the way of quality instruction. Boaz is too smart to teach this class; his textbook is awful and his lectures are (somehow) worse because he has no idea what it means to learn theoretical computer science from scratch! The problem sets weren't 124 brutal but they were unnecessarily complicated and overall requires way too much in the way of extrapolation of concepts learned in the course. The biggest problem is that BOAZ CANNOT TEACH. He relies too heavily on understanding from the textbook, which would be fine if it was decipherable, and his lectures are just a waste of 75 minutes.

Evaluate the course overall.: **5 (excellent)**

1) Make sure the final is not the same day as STAT 110. 2) Provide solutions to the homework or at least provide practice problems similar to the ones in homework with solutions for each topic. Sometimes readings proofs gives you the right framework to think about things and make sense of the material.

Evaluate the course overall.: **3 (good)**

I thought that the collaboration policy on problem sets was a little strict given how difficult they were, and they could also have been condensed. Also, I feel that more review of what was in the reading during lecture would have been helpful. Yes, I did do the reading, but that doesn't mean I understood everything. Finally, I think it could be helpful for Prof. Barak to use the chalkboard more often, that was he's writing as quickly as we can take notes. It's tough to write down everything from the slides.

Evaluate the course overall.: **3 (good)**

The textbook was slightly disorganized and a bit unfinished it seemed. Lectures weren't regurgitations of textbook material, but the explanations of material was sometimes confusing.

Evaluate the course overall.: **4 (very good)**

The textbook had a lot of typos, and some of the proofs were unnecessarily confusing (like the Cantor's theorem proof for example)

Evaluate the course overall.: **4 (very good)**

Lecture quizzes don't make this course better. It's possible to get 50% consistently with good multiple choice skills and occasionally command-Fing through the downloaded textbook. It adds an extra recurring deadline students have to remember without contributing anything pedagogically. Grading was at times shoddy; as a TF myself, I understand when graders miss things, but I shouldn't have to explain class concepts in my regrade requests. If anything, TFs should be explaining concepts to me.

Evaluate the course overall.: **4 (very good)**

The lectures could have been made more clear and not just another reiteration of the textbook.

Evaluate the course overall.: **1 (unsatisfactory)**

Cleaner presentation of material. More organization and rigor.

Evaluate the course overall.: **3 (good)**

Organization. More outside resources besides office hours which were usually crowded

Evaluate the course overall.: **4 (very good)**

Less NAND. In the beginning, I found anything to do with NAND very confusing. However, once we got beyond the point of doing things with NAND, NAND++, NAND<<, and converting between them, I found it to be much clearer at later points that we simply use NAND++ as a way to talk about Turing complete models.

Evaluate the course overall.: **4 (very good)**

I wish we had the broader overview of lecture before reading the details about each topic.

Evaluate the course overall.: **1 (unsatisfactory)**

many concepts were unnecessarily difficult to learn because of how they were taught! Man, I love Boaz but

Evaluate the course overall.: **4 (very good)**

There were many typos in the textbook that obstructed understanding the concept, and I often had to reread sentences.

Evaluate the course overall.: **3 (good)**

Less reading or make it more understandable (slightly less mathematical rigor where it doesn't really matter)

Evaluate the course overall.: **1 (unsatisfactory)**

The way things are graded is awful: I've had friends who TF CS124 multiple semesters and the thing they would always complain about is people writing enormously long proofs for pset questions... and would wonder where that habit came from. Now I know it comes from 121: the amount of pedantic nonsense 100% impeded my ability to enjoy any of the material (especially since I took 121 after 124), only some of the more mathy stuff was interesting like diagonalization but otherwise no.

Evaluate the course overall.: **3 (good)**

The course could have provided more resources in terms of practice problems and other exercises to help students prepare for exams.

Evaluate the course overall.: **4 (very good)**

The course progresses at such a fast pace that it never felt like I ever had mastery in any topic until reading period when I had more time to digest it all. The weekly psets, quizzes, readings, lectures, and midterms felt like I was barely hanging on at most times.

Evaluate the course overall.: **4 (very good)**

The textbook could be made easier to read. Often there are walls of text with a bunch of complicated notation that is decipherable with enough effort but not easily understandable. However, the proofs that are gone over in lecture are well-explained and more understandable (probably because the format is easier to digest). I especially liked the proof that BPP subset P/poly that we went over in class. Also, it would be helpful to be more clear about which topics are the most important that we should focus on learning, for example it was unclear how much probability/cryptography/proofs/quantum we needed to know, and earlier in the course there were less important topics like the NAND++ to Turing Machine conversion and details about NAND<<.

Evaluate the course overall.: **3 (good)**

I think that the first mid-term could have been a bit more fair in terms of length. The second midterm was much better.

Evaluate the course overall.: **4 (very good)**

The grading of the homework was inconsistent at the beginning; they didn't seem to know how formal they wanted things to be, so they would give you full credit for a high level description, but then other times slight you for not EXPLICITLY proving things that you mention in a sentence.

Evaluate the course overall.: **5 (excellent)**

just fix up the errors in the textbook - that's literally all. There are a few parts that have some significant errors in mathematical syntax that impede understanding.

Evaluate the course overall.: **4 (very good)**

At times, the degree of mathematical formalism in the textbook and in lecture can be excessive and can detract from an understanding of the material. It's not always clear what degree of formality one is expected to use when answering questions on the homework or on exams.

Evaluate the course overall.: **3 (good)**

- The textbook is bad. Not terrible but needs a lot of work to be done on it. Give more concrete examples, get rid of excessive notation, and actually explain what is going on in these theorems. I think a very nice improvement of the textbook would be to add another colored space where every part of a given theorem is explained - what does given notation mean, why do we use it this way, what does the theorem tells us as a whole. There were some sections of the textbook that I just gave up on understanding - they were so abstract and so poorly written that I just couldn't (e.g. the optimal PRG conjecture). - Lectures were mostly okay but even there, I would have loved to see more examples of the material from the textbook. Just give us examples of how to use those theorems. The poll everywhere questions in lecture were probably one of the best parts of the lecture. - I don't think we need NAND to understand the second half of the semester. NAND was great to explain things in the beginning but then it just became too much. Trying to explain cryptography through NAND was a pain in the ass. Maybe the problem is in me for not understanding NAND very well, but I think that you should actually consider at least explaining the theorems in other terms (e.g. I liked the definition of BPP since it involved a NAND explanation and a normal explanation). - The final was very well designed and there was plenty of time but the midterms were so time pressured. I knew the whole material for both midterms and I just couldn't finish. I think we should have had more time because I didn't feel like I managed to show what I actually know. - I think the last few sections were rather poorly explained. I know most people didn't get any of the cryptography/zk proofs/quantum stuff. Maybe, there was just too much information. - Also, I didn't like that we have to read the textbook before going to lecture. In my view, the lecture is to teach us and the textbook is to fill the gaps after going to lecture but I may be wrong.



Evaluate the course overall.: **4 (very good)**

Second half could be streamlined a bit. Draw out certain themes more and focus less on others perhaps (proof systems are interesting but hard to understand in a lecture, flesh out quantum instead perhaps?)

Evaluate the course overall.: **3 (good)**

The instructor has trouble explaining basic concepts in a manner that is understandable. He relies heavily on mathematical notation that is not always clearer than a simple explanation in words.

Evaluate the course overall.: **4 (very good)**

The midterms should be 15% each and the final should be 30%

Evaluate the course overall.: **4 (very good)**

Boaz needs to be more purposeful and careful of a lecturer - make sure to emphasize which parts of which proofs we need to understand. The proofs are also presented in the textbook in a way that's counterintuitive and seems to sometimes just use more mathematical language simply for the hell of it when in actuality there's a much more understandable 3-sentence explanation. I'd really appreciate seeing a lot of the proofs being contextualized immediately / as they're introduced, not something that we have to digest and figure out after reading it in class and literally seeing the same complexity of a proof in class.

Evaluate the course overall.: **4 (very good)**

I feel like sometimes the psets contained material from outside of the course that we would need to complete and I didn't have that background.

Evaluate the course overall.: **5 (excellent)**

This course is obviously harder than many if not most other courses at Harvard, but an A is an A anywhere. I wish you could somehow reconcile how intense this route is relative to other courses. It is just 15-25 hours a week and courses like are treated just like any other ~1/4 classes in a semester and it seems like just an uninformed approach to it all. IDK,

Evaluate the course overall.: **3 (good)**

textbook could be more coherent, lectures could be significantly more coherent, sections should be mandatory and in convenient locations

Evaluate the course overall.: **1 (unsatisfactory)**

This shit sucks ass.

Evaluate the course overall.: **4 (very good)**

Very notationally dense (was often hard to understand fairly intuitive concepts because they were hidden under very abstruse notation), too many typos in the textbook (although this was understandable given the textbook hasn't been published yet).

Evaluate the course overall.: **5 (excellent)**

The textbook proofs can be simplified, especially since it's an intro to TCS class. Perhaps have an appendix containing the proofs and just have a more fleshed-out proof idea in the chapter. This is especially hard for the NAND chapters such as memory allocation in the NAND++ states.

Evaluate the course overall.: **1 (unsatisfactory)**

This course could be organized a lot better, with solutions released to problem sets. The lectures could be improved because they are very hard to follow.

Evaluate the course overall.: **3 (good)**

The textbook is still very much a work in progress and can be difficult to understand

Evaluate the course overall.: **4 (very good)**

I thought that the material at the end of the course felt rushed. I think that it would be better to focus on one of the topics of crypto, quantum, or probability. It felt like we went through those topics really fast. In addition, I felt like the python code in the textbook and lecture did not really contribute to my learning and often confused me rather than helping me learn a topic.

Evaluate the course overall.: **1 (unsatisfactory)**

1. EXPLAIN WHAT THE STUFF ACTUALLY MEANS!! Instead of bombarding us with walls of mathematical jargon and an annoying yellow box yelling at us to read over that jargon over and over again, FIRST explain the concepts themselves and their intuitive understanding in plain English, using simple language and using analogies and many examples, until we actually understand what is going on, AND THEN introduce the mathematical formalism, in which case it will come naturally and it will make sense. Math on its own is not intuitive and doesn't make sense. Don't keep telling us to read over overly pedantic and highly technical definitions because that's not how understanding works. That's not how anything works. It literally took almost an entire paragraph of inexplicable math symbols to simply come to the conclusion that the probability of something not happening is $1 - \text{the probability of it happening!!}$ Some concepts that I understood before this class were made overly complicated and confused me for absolutely no reason, such as probability. What are Chernoff bounds?? I have no idea. The math makes NO SENSE. 2. Be clear about the requirements from the homework. I started by writing simple easy proofs that were marked down for not being "concrete" enough, then ended up switching to more pedantic overly detailed proofs that took SO LONG to write! I am almost sure that my proofs by the end were overly detailed and unnecessarily so, but I had no idea of knowing. Either way is not good and inefficient. The solution is to be very clear from the start of what the expectation is. 3. Make homeworks shorter or less about hard detailed proofs. For most problem sets, I spent less than half the time actually understanding the problems and the solutions and the proofs, and the majority of the time simply coding up and writing down the proofs. This is a MASSIVE waste of time. Either make the homework like the midterms where only one or two problems require full detailed proofs while the others are simply quick brief explanations, or make problem sets shorter. 4. Speak in English. Can you just please explain what the math means? I know I said this before but this is really my biggest criticism. Math will come naturally if we actually understand what you're talking about. There really is no need to be overly pedantic with the math jargon.

Evaluate the course overall.: **1 (unsatisfactory)**

The course is taught too fast, TFs admit that the psets are way too hard and sometimes don't even understand them, there's a serious lack of solid example solutions. Examples taken from section can often get marked down if you use the same styling.

Evaluate the course overall.: **4 (very good)**

More Interesting Content

Evaluate the course overall.: **3 (good)**

Thought lecture notes were too complicated and lecture and Psets did not help me understand the material.

Evaluate the course overall.: **2 (fair)**

more practice problems and guidance in sections on how to approach problems

Evaluate the course overall.: **4 (very good)**

more summaries of dense textbook content

Evaluate the course overall.: **3 (good)**

I would love to see the material be covered with more rigor. By rigor I do not mean difficulty, but rather attention to detail. Some of the concepts and proof techniques were hand-waved more than I would have liked. A lot of the beauty of proof based material is in the details.

Evaluate the course overall.: **2 (fair)**

Learning things / concepts in the way that is done outside instead of having a Harvard CS121 way of doing things would be appreciated.

Evaluate the course overall.: **4 (very good)**

The course was not organized effectively. I liked the textbook overall, but it routinely had distracting errors and was unnecessarily dense, which simply encouraged students to give up on it a lot of the time until they had learned the material from a different source, in section or lecture or for the problem sets. At this point we could go back to the textbook to review and it was very helpful, but the textbook was not at all helpful in teaching the material as a first point of contact. Further, there was a dearth of materials to help study for tests and go over psets, which meant that expectations were not always clear (I did well in the course so this is not out of spite, just so the instructors know how the students felt). Pset and midterm solutions should be released - this is not a complicated matter. If the hesitation about releasing such solutions is that students will store the answers and share them with next year's class -- well, a) there is an honor code that should deal with that, b) essentially every other STEM class operates in this manner and this concern does not seem to be an issue and c) having them would greatly, greatly enhance the learning experience of the vast majority of students. Postmortems are on the right track but just don't cut it.

Evaluate the course overall.: **4 (very good)**

(0) The main problem is that I don't feel like mathematically advanced students get too much out of this course. (1) The pset questions, especially in the beginning of the course, should be structured to feel less like a "grind" and to emphasize the conceptual material more. (2) The lecture quizzes are not useful.

Evaluate the course overall.: **3 (good)**

The course could be improved with better reading materials. The lecture notes had a lot of typos and the proofs were a bit hard to understand at times

Evaluate the course overall.: **1 (unsatisfactory)**

Please organize the course more. The textbook had too many errors, and the lectures went too far into unnecessary details.

Evaluate the course overall.: **4 (very good)**

Most facets of this course were great. I would appreciate a more polished textbook.

Evaluate the course overall.: **1 (unsatisfactory)**

Write a new textbook that clearly explains it material instead of the instructor simply stating "read it multiple times."

Evaluate the course overall.: **1 (unsatisfactory)**

The pedagogy, at least from the professor, is terrible (see the satire v article on this class, it's surprisingly accurate). The textbook is still in progress but is the only course reference, the proofs are full of overcomplicated notation and little intuition, and lecture is essentially a repeat of the textbook material in 75 minutes. The problem sets are too easy and there are so many bonus points given on those that it's entirely possible to have a full problem set average while not understanding a fair amount of the material (and likewise for the tests); the tests themselves have an opaque grading scheme and fluctuate in difficulty, to the point that I'm not sure if anyone's average is a good reflection of their understanding of the material unless their average is extremely good or bad. The professor ostensibly pays attention to student feedback, but he responds in absurd ways (for example, after the midterm people said they were having trouble understanding the proofs, and in response he told people to tune out in lecture while he was going through a proof). The attitude of the class to "rigor" is insulting - although the class claims to pay attention to rigor, often the textbook's "rigorous proof" is a mess of too-formal notation and meaningless symbol-pushing that ignores the fact that one can be "rigorous" without resorting to writing an equation every line; thus problem set solutions are often graded on a scale that equates length with rigor rather than actual proof (it turns out that if you write a page or more of BS that sounds vaguely correct, you can get full credit on a problem, probably because the TFs saw some words they were looking for and didn't want to read through the entire thing). The structure of the course also needs work: taking the course seemed like going through a grab bag of topics in computability theory, and it seemed unlikely that there was some unified purpose to the class. It didn't help that basing things on NAND meant that if we had questions, the internet could not help us.

Evaluate the course overall.: **1 (unsatisfactory)**

This course was clearly unorganized and thrown-together. The HW were impossible without help, the content was trivial at most times, and the lectures were difficult to follow.

Evaluate the course overall.: **3 (good)**

More tuning of exam difficulty and length (which was done with some success between the midterms and the final).



Evaluate the course overall.: **2 (fair)**

The website, Canvas, Piazza, IntroTCS site, and slide website need to be consolidated all into one place. I frequently found myself feeling out of the loop because I didn't see a particular Piazza post or a Canvas message. It took a long time to navigate between all the sites and find the correct URLs for everything- they all need to be available on ONE website (preferably Canvas). If possible, Professor Barak should eliminate his Office Hours that are "by appointment only" and just open them up to everyone. If a student needs to make an appointment, he or she can email the professor and they should meet at an alternate time because other there is only one single office hour every week to attend. I also think lecture should really explain the readings more. Even when I did the readings, I had a hard time understanding the concepts and it felt like once we were in lecture, Professor Barak often assumed we really understood the concepts and built upon those concepts which made it difficult to keep up.

Evaluate the course overall.: **4 (very good)**

The way in which it is taught. Also, I found the partner pset system not to be useful, as for most students they just dived the pset between them two and learned only half the material. Is there a reason this course using partners rather than collaboration while listing collaborators? Additionally, I found the focus on NAND as a language to detract a bit from the theory of computation aspects.

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.) — Add Comments?

Course

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

The textbook was really hard for me to understand, especially with so many notational details that I often confused with the process of the proof itself. Homework was also difficult for me and it took going to office hours to clarify problems to get started on them.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

I know this is theoretical CS, but I still found the textbook super hard to follow. Why do such simple problems have to be explained in such a convoluted way the first time around?

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Textbook is clearly an unfinished product (not a bad thing) but typos and TODO were more frequent than ideal. Many of the definitions given seemed to go out of their way to phrase things in the most convoluted way possible. The "Proof idea" sections before each proof in the text were SUPER HELPFUL in subsequently reading and understanding the proof.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

The textbook was very confusing at times. It also would have been nice to see more examples and explanations.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

The textbook, although some explanations could certainly be abbreviated and improved, was essential to understanding the material of the class. Lecture effectively presented summarized textbook readings and gave context for how this material was useful.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Textbook did not give enough examples of how to apply the concepts that were covered, but we were expected to do that in the pset. An exception was the section on reductions, for which there were a lot of examples. There were sections in the textbook that were too detailed and convoluted, so it was hard to know what the main takeaways were supposed to be and what we were expected to know.



Evaluate the course overall.: **2 (fair)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

The textbook was difficult to understand at some points.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

Continue to fix typos in the textbook

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

It's a lot of Boaz, external materials would be super helpful :)

Evaluate the course overall.: **2 (fair)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

Textbook is good but has a lot of excess information that can be hard to work through. I found that just reading, and re-reading, important theorems and their proofs was most helpful

Evaluate the course overall.: **5 (excellent)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

The textbook was a dense at times. I found it hard to follow the huge amounts of notation in the proofs.

Evaluate the course overall.: **5 (excellent)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

The textbook was very good at explaining the material, though I would often need to re-read sections to understand them fully. I wish it had fewer TODOs.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Textbook was confusing and archaic/useless things were sometimes overemphasized.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

A lot of confusing typos



Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

The textbook is not incredibly useful or easy to understand, and the course website is basically piazza, which is unorganized and not the best place to find materials.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Textbook was dense and confusing at times. Section notes were helpful. Lectures were good.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

I found the textbook and review materials to be inaccessible and hard to follow.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

Textbook is dense and requires a lot of rereading to understand, but after that I appreciated it more. Lecture quizzes were good for keeping me on track. Lecture slides are very good!

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Textbook can be hard to read and has a lot of typos

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

fix the typos in the textbook. also, the textbook simply isnt clear and doesnt explain things intuitively at all.

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

The course textbook was almost unusable and littered with typos. I strongly urge moving to an alternate and completed text.

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

The textbook was extremely tedious to follow.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

Reading is pretty dense, pretty difficult to understand by yourself.



Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

Textbook was often confusing, although the diagrams and definitions/theorems were laid out really well

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

I love Boaz's book!

Evaluate the course overall.: **2 (fair)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

hard to comprehend at times to get the big picture

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

The textbook is very helpful but poorly written.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

It was very difficult to parse the textbook, and often would just use mathematical symbols to describe concepts when it would have been very helpful to have a big-picture overview or example to give us an idea of what the concept is actually trying to say rather than parsing the math to understand it.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

Typos everywhere. Lack of focus.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

The lecture notes still need to be improved.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

The book is fine, still in progress but good nonetheless.



Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

The textbook is a decent place to get a first pass of the new material. The problem comes from the fact that the author and the lecturer are one and the same: if an explanation in the textbook didn't make sense, the lecture will likely give you the same confusing explanation. The only comment in the textbook regarding this problem is that you should re-read it more times.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

The textbook is still a bit rough, but this is to be expected from something which is still a work in progress.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

Way too many typos in the book

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

The readings were very good.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

The textbook isn't quite finished

Evaluate the course overall.: **2 (fair)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

entire chapters missing from the textbook and no numbers on the theorems in the online version, which was pretty confusing

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

Piazza, the textbook, recorded lectures, etc, all great.

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

Textbook has tons of typos, is hard to read and understand

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

Textbook was dense and unclear, which occasional confusing errors. See my suggestions for improving it in the general feedback above.

Evaluate the course overall.: **5 (excellent)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

The course textbook was thorough and interesting. The proofs could get a bit dense, but often times there were pictures or diagrams to help in the understanding of these proofs. The reading guidelines were especially helpful for focusing in on the important parts of the chapter and knowing when we should be re-reading until we fully understand the concept. It may be useful to more frequently note when understanding the proof is important versus understanding the conclusion. Most of my learning came from the readings and was cemented by the problem sets. Lectures were helpful for focusing in on critical concepts in the reading and for when I was still struggling to grasp a topic.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

The textbook was dense and sometimes the abundance of mathematical notation detracted from my ability to parse and understand the material. The proof ideas were helpful. The fact that the slides, homework/section notes, and syllabus were split across 3 separate websites (canvas, cs121.boazbarak.org, introtcs) was also confusing. In the future please pin a Piazza post with the locations of all the lecture slides. Finally, the fact that the textbook was continually being updated and re-compiled was not made clear to the class. I appreciate that Boaz was improving it as the course went on! But make sure to let people know to update their version of the textbook, because many will just use the pdf they downloaded at the beginning of the year.

Evaluate the course overall.: **5 (excellent)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

The textbook was very good because it was very thorough, and it provided formal definitions and theorems but also made sure to explain them in a way that was intuitively understandable.

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

So I appreciate that the material is all in one place. Boaz clearly put a lot of time into his textbook, and he's clearly a brilliant individual. That said, THE TEXTBOOK IS UNREADABLE. IT'S SO AWFUL. The proofs are a waste of time (they're literally not in the same language that I know to be English), his definitions are extremely clunky and often unnecessarily so, and the images/examples are so esoteric so as to make me question his grasp of an undergraduate's reality.

Evaluate the course overall.: **5 (excellent)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

A lot of people didn't like the book, but I personally loved it. It's a work in progress for sure but it is a great work.



Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

The textbook is pretty rough at the moment. I think it needs a bit more polishing before use in the classroom.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Textbook needs fewer walls of text with complicated notation.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

The materials are good, but there needs to be more work to differentiate between what is important and what is not. Specifically, many of the main "theorems" of each chapter were not mentioned again, ever (I'm looking at you, Schwartz-Zippel lemma). It would be helpful if the chapters that we only care about the general "flavor" of the result (think Godel, restricted models, etc.) were more strongly differentiated from chapters where it is important we understand the details (NP, Halting, BPP, etc.)

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

The book is good in general but readings can be pretty dense. Some of the proofs would have been more understandable if treated less formally.

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

- The textbook is bad. Not terrible but needs a lot of work to be done on it. Give more concrete examples, get rid of excessive notation, and actually explain what is going on in these theorems. I think a very nice improvement of the textbook would be to add another colored space where every part of a given theorem is explained - what does given notation mean, why do we use it this way, what does the theorem tells us as a whole. There were some sections of the textbook that I just gave up on understanding - they were so abstract and so poorly written that I just couldn't (e.g. the optimal PRG conjecture).

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

Book was nice, numbering of theorems was annoying sometimes (not in web)

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

Typos in the textbook

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

The textbook is difficult to navigate and often not enough

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

420 :(

Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

the textbook can be difficult to understand and still has a lot of errors

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

I think that once the textbook's bugs are worked out it will be great

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Lots of typos in the textbooks and most visualizations were hard to read with bad handwriting.

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

The book is, if you're not used to CS theory, garbage even if you try to read it again, and again, and again

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **2 (fair)**

Textbook was riddled with errors, question marks, incomplete sentences and typos, even in important theorems! This made it an ineffective way to first encounter the material. Overall, the textbook was an incredibly impressive endeavor and has great potential. It just needs a lot more work before it is ready to see the light of day.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **5 (excellent)**

I actually liked the book

Evaluate the course overall.: **1 (unsatisfactory)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

Way too many typos and hard to parse textbook.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **1 (unsatisfactory)**

It is frustrating that the course textbook contains errors and sometimes lacks full proofs.



Evaluate the course overall.: **3 (good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

The textbook is essential; the lectures complement it. It still has a lot of typos but generally they do not pose too much of a problem.

Evaluate the course overall.: **5 (excellent)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **4 (very good)**

The book is great but it wasn't always clear what exactly we were expected to know. Some proofs went into too much detail/where beyond the scope of the course. It's great to have extra details but it should be better emphasized what exactly we needed to learn.

Evaluate the course overall.: **4 (very good)**

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.): **3 (good)**

Textbook needs to be finished, lots of typos and missing sections in the pdf.

Assignments (exams, essays, problem sets, language homework, etc.) — Add Comments?

Course

Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **3 (good)**

Problem sets were often where I did most of my learning, but I did feel that I needed to go to office hours to clarify what the questions were asking.

Evaluate the course overall.: **4 (very good)**

Assignments (exams, essays, problem sets, language homework, etc.): **3 (good)**

Problem sets were good to learn the material, but many proofs assigned had truly inelegant and time consuming solutions that were not a productive use of learning, at least from the student perspective. Exams were fair.

Evaluate the course overall.: **5 (excellent)**

Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**

Problem sets were challenging but definitely helped consolidate the material.

Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **3 (good)**

Problem sets were a bit overly difficult in some areas. Some problem sets were utterly incomprehensible (misused words/punctuation), and TFs were required just to understand the questions. In the end, those questions were very simple.

Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

Problem sets were great tools to learn. Midterms were not effective ways to learn because there was not enough time to thoroughly think about the problems.

Evaluate the course overall.: **4 (very good)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

- Homeworks were very reasonable for the most part, but it would be good if we were given more context about how they fit in with the course material. For example, I didn't understand that the NANDHARD question was referencing how to show P is in P/poly until way later in the course. Having a couple sentences to provide context would be so helpful for contextualizing all the material, especially since questions on the tests are often very similar to the psets - It would be great to see solutions to the psets. I loved having HW 10 be review for the final! - Would like to see more review problems with solutions for the midterms and finals, but I think the course staff did a great job of coming up with solutions on the go. MAJNOTSAT was a little triggering but it was an educational experience trying to solve it:) - There were some things that were not made clear grading wise (e.g. I didn't know we were not supposed to use the same witness for one of the questions on the final) because the psets were graded very generously and tests were not. Standardizing this in the future would be good



Evaluate the course overall.: **1 (unsatisfactory)**
Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**
They were fair given what we were supposed to learn.

Evaluate the course overall.: **1 (unsatisfactory)**
Assignments (exams, essays, problem sets, language homework, etc.): **1 (unsatisfactory)**
The exams were arbitrary in difficulty and focus. I found the problem sets did not adequately test the material being taught.

Evaluate the course overall.: **1 (unsatisfactory)**
Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**
Some of the problems were nice, but it's never clear if bonus problems are extra credit or required???

Evaluate the course overall.: **3 (good)**
Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**
I liked the format of psets, pushed you to do more because of the bonus questions

Evaluate the course overall.: **3 (good)**
Assignments (exams, essays, problem sets, language homework, etc.): **3 (good)**
Timing needs work with the midterms but the final was much better!

Evaluate the course overall.: **1 (unsatisfactory)**
Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**
The psets are quite arbitrary and almost impossible to do without advice or without a partner.

Evaluate the course overall.: **3 (good)**
Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**
The bonus problems structure is sound.

Evaluate the course overall.: **1 (unsatisfactory)**
Assignments (exams, essays, problem sets, language homework, etc.): **3 (good)**
The problem sets are a good application of what you learned, but the exams had a large time pressure component to them.

Evaluate the course overall.: **4 (very good)**
Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**
The psets were very enjoyable and helpful for learning the course material.



Evaluate the course overall.: **4 (very good)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

Problem sets were usually interesting, but often pretty easy. They could have prompted more hard thinking along new lines from what was in the reading, in my opinion.

Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

Problem sets were very helpful for understanding the material. Sometimes, I felt that completing them was so stressful that I didn't understand anything, but when I went back to review I realized that I really understood so much and was getting the material more and more as I kept writing proofs and practicing.

Evaluate the course overall.: **1 (unsatisfactory)**

Assignments (exams, essays, problem sets, language homework, etc.): **1 (unsatisfactory)**

they did not really test whether or not we understood the concepts. we were tested on intuitive cleverness, which was definitely not taught in the class. ridiculous.

Evaluate the course overall.: **1 (unsatisfactory)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

Homework problems were interesting, and prepared us well for the exams. Having two midterms was fine, since it alleviated the stress for the final. However, there could have been more preparatory material for the second midterm, although it is understandable this year since it is the first time.

Evaluate the course overall.: **5 (excellent)**

Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**

I felt that the problem sets were what really cemented my understanding of the concepts in this course. The problems were well thought-out and usually appropriately challenging. There were some ideas that I would not have gotten without going to office hours (homework 9 problem 3.1, for example, required the use of a "common approximation for $1/e$ " which I didn't know), and avoiding this kind of assumed knowledge (or adding hints as was often done) is definitely helpful. I thought the second midterm was quite fair but felt the first one had too much of an emphasis on uncomputability when our homework covering uncomputability wasn't yet due. I thought that the final was quite fair as well, and appreciated that the bonus probability problem was there in case students weren't quite able to nail down the reduction problem.

Evaluate the course overall.: **5 (excellent)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

Midterms, especially the first one, were too short in duration. Ideally I should leave an exam feeling that it tested not my knowledge and my ability to score points quickly, but my knowledge alone. I felt that the final exam did the latter.



Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

Overall, I thought the homework assignments were fair and helped me develop my understanding of the material. I thought the midterms were also fair and appreciated how they were more high level than the homeworks (for example, no need to recall specifics of the mechanics of NAND++/NAND<</Turing machines) and focused on big picture concepts. My one complaint, which I'm sure has been said before, is that the first midterm was unnecessarily time-pressured, which was especially disappointing given that we were reassured multiple times that it would not be.

Evaluate the course overall.: **5 (excellent)**

Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**

The problem sets were composed of very interesting and challenging problems that prepared us well for the exams.

Evaluate the course overall.: **5 (excellent)**

Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**

I enjoyed a lot of the problems in the problem sets.

Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**

The problem sets were longer than they needed to be, and the collaboration policy was strict. Also, the exams were longer than they should have been given the amount of time given.

Evaluate the course overall.: **5 (excellent)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

There were inconsistencies in solutions across exams and homeworks, which made it difficult to gauge how much rigor was necessary in answering a question.

Evaluate the course overall.: **4 (very good)**

Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**

Pset problems were very good practice with the material and the bonus problems added extra practice. Tests were fair and similar to homework problems.

Evaluate the course overall.: **4 (very good)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

The problems themselves are good.



Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

- The assignments helped with the understanding of the material but they were soooo boring. Honestly, there were some problems that were just pain in the ass to write down - you know that the statement is obviously true, you know theorems to use, you know everything but it is just terrible to write down and oh god so boring. Please make them a little bit more interesting even if that means making them a little bit more challenging.

Evaluate the course overall.: **4 (very good)**

Assignments (exams, essays, problem sets, language homework, etc.): **5 (excellent)**

I like the idea of bonus problems on the pset. Bonus programming was less necessary.

Evaluate the course overall.: **5 (excellent)**

Assignments (exams, essays, problem sets, language homework, etc.): **4 (very good)**

Very Very Hard.

Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**

The exams were extremely harshly graded; the psets were a lot fairer although there was very little guidance on how they could be tackled without staff help

Evaluate the course overall.: **1 (unsatisfactory)**

Assignments (exams, essays, problem sets, language homework, etc.): **1 (unsatisfactory)**

69

Evaluate the course overall.: **2 (fair)**

Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**

Exams have large margins; i.e. one small word choice can cost 10 points either way on an exam.

Evaluate the course overall.: **1 (unsatisfactory)**

Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**

Exams don't accurately assess mastery of material

Evaluate the course overall.: **3 (good)**

Assignments (exams, essays, problem sets, language homework, etc.): **3 (good)**

the problems sets were fair and offered lots of extra points



Evaluate the course overall.: **1 (unsatisfactory)**

Assignments (exams, essays, problem sets, language homework, etc.): **2 (fair)**

Problem sets were very long, overly pedantic, and took way way too much to simply write up.

Evaluate the course overall.: **1 (unsatisfactory)**

Assignments (exams, essays, problem sets, language homework, etc.): **1 (unsatisfactory)**

The quizzes often feel like trick questions, the final is worth 40 percent of your grade, and the psets are HARD and without mercy (I attended one-on-one tutoring weekly even, and I still believe this)

Evaluate the course overall.: **1 (unsatisfactory)**

Assignments (exams, essays, problem sets, language homework, etc.): **3 (good)**

The problem sets were pretty engaging and reasonably well put together.

Feedback you received on work you produced in this course — Add Comments?

Course

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

Not much feedback on psets or midterms. Often time the feedback was not super helpful or really inconsistent.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **3 (good)**

It was sometimes hard to interpret my mistakes from the responses.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **3 (good)**

It was hard to know specifically what I did wrong on homework due to the limited descriptions in Gradescope.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **4 (very good)**

Grading was clear, and important mistakes were pointed out.

Evaluate the course overall.: **5 (excellent)**

Feedback you received on work you produced in this course: **4 (very good)**

It would have been awesome if official solutions were published!

Evaluate the course overall.: **5 (excellent)**

Feedback you received on work you produced in this course: **5 (excellent)**

I liked the use of gradescope because the rubric made grading more transparent and objective. Grading was often done in a timely manner.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **4 (very good)**

Post-mortems were very helpful

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **2 (fair)**

Some TF's had better feedback than others. Written answers to the problem sets would have been super helpful. Even though there is a postmortem homework section, there is a difference between hearing someone explain the proof and seeing it on paper.



Evaluate the course overall.: **3 (good)**
Feedback you received on work you produced in this course: **4 (very good)**
Feedback on problem sets were good.

Evaluate the course overall.: **4 (very good)**
Feedback you received on work you produced in this course: **4 (very good)**
Feedback on psets was good but should be standardized compared to grading on final/midterm

Evaluate the course overall.: **1 (unsatisfactory)**
Feedback you received on work you produced in this course: **1 (unsatisfactory)**
i never understood what the feedback was about because i never really understood what we were supposed to learn.

Evaluate the course overall.: **1 (unsatisfactory)**
Feedback you received on work you produced in this course: **1 (unsatisfactory)**
Problem set comments were negligible.

Evaluate the course overall.: **3 (good)**
Feedback you received on work you produced in this course: **3 (good)**
It was through gradescope, mediocre

Evaluate the course overall.: **3 (good)**
Feedback you received on work you produced in this course: **3 (good)**
Mostly good but sometimes comments on gradescope are too brief and need clarification

Evaluate the course overall.: **1 (unsatisfactory)**
Feedback you received on work you produced in this course: **1 (unsatisfactory)**
The teaching fellows seem to know absolutely nothing, and several times have given faulty information.

Evaluate the course overall.: **3 (good)**
Feedback you received on work you produced in this course: **4 (very good)**
Timely grading and feedback. Thank you!

Evaluate the course overall.: **3 (good)**
Feedback you received on work you produced in this course: **2 (fair)**
Often had to get regrades. TFs are probably rushing through to get through all the psets.



Evaluate the course overall.: **1 (unsatisfactory)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

Releasing homework answers would provide students with a great study tool, especially those who are less familiar with writing proofs.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **2 (fair)**

A lot of the grading was too strict, though considering how many bonus points there were, this may not actually be as bad as I think it is.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **3 (good)**

Problem set scores sometimes felt like the graders had only cursorily read the solutions. Feedback was sometimes insufficient to determine what was wrong about a problem.

Evaluate the course overall.: **2 (fair)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

I had to request a regrade for over half of the pssets and every exam. While I appreciate the TAs' responsiveness and willingness to accept my regrades, it raises questions of why the graders are detracting points so sloppily in the first place--half the time opening gradescope incites in me unnecessary panic! This is coupled with the fact that the feedback on the problem sets is subpar--"point adjustments" with no explanation, for instance--which was frustrating because the homework didn't feel like a place where I could really track my progress.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **2 (fair)**

Because of how homework grading was automated online, comments on homework/tests often were pathetically incomplete, confusing, and occasionally wrong. A few times my requests for regrades weren't taken seriously, so I just gave up.

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

I thought problem set comments were not helpful in helping us improve and neither were the midterm comments.

Evaluate the course overall.: **1 (unsatisfactory)**

Feedback you received on work you produced in this course: **2 (fair)**

Feedback on homework mistakes was detailed...but sometimes it felt that the grader had not actually read my homework response, since at times the parts they had marked as "missing" were clearly there. One time, the counterexample they gave simply did not work.

Evaluate the course overall.: **5 (excellent)**

Feedback you received on work you produced in this course: **5 (excellent)**

Overall, feedback was usually very detailed and helpful, the only exception being (understandably given the timeframe) the last homework. Postmortem sessions were also great for reviewing problems if we didn't understand the feedback on the homework.

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **2 (fair)**

While overall homework feedback was timely, there were multiple instances when I received full bonus points for a bonus problem that I didn't even attempt, or for a very obviously partial answer (e.g. no inductive step written for an inductive proof). While I can't complain about the free points, it does make me concerned that the graders are not reading my work thoroughly. In addition, sometimes the feedback would not apply to my answer, or would seem like feedback for a different question on the homework (e.g. feedback for 2.1 seems like it applies to 2.2). Finally, sometimes on Gradescope, a problem would be marked down with "incorrect proof" but no (or very minimal) explanation would be given, which is frustrating. I understand that you want us to watch the homework postmortems but if a student puts effort into writing a long proof, then it would be nice to at least get a few words of specific feedback.

Evaluate the course overall.: **2 (fair)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

Most of the time I wasn't sure what I had done wrong. On gradescope, the point deduction comment left me wondering if I lost points because I didn't do the thing it specified, or because I did do the thing it specified.

Evaluate the course overall.: **5 (excellent)**

Feedback you received on work you produced in this course: **5 (excellent)**

The feedback I received on assignments was very specific as to what I should improve.

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **2 (fair)**

I didn't receive very much feedback on the problem sets.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **4 (very good)**

There were some inconsistencies in the beginning, but the bonus points make you forget about all of that.

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **4 (very good)**

A lot of wrong grading on the midterms.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **5 (excellent)**

Very fast, gradescope is good

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **4 (very good)**

Hw corrections could be more explicit

Evaluate the course overall.: **1 (unsatisfactory)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

With the exception of Pranay Tankala's grading. He is the best CA CS121 has ever known.

Evaluate the course overall.: **5 (excellent)**

Feedback you received on work you produced in this course: **3 (good)**

The graders sometimes dwell on tiny things (prefix free encoding when making it prefix free doesn't change anything). It gets annoying ;|

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **4 (very good)**

the feedback was timely and generally helpful

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **3 (good)**

I felt that the feedback provided on gradescope for psets was very generic and didn't speak to my individual submission. I understand that there are a lot of psets to grade, but it would be nice to get more specific feedback for improvement.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **4 (very good)**

Sometimes graded wrong, lot of errors

Evaluate the course overall.: **1 (unsatisfactory)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

GIVE US COMPREHENSIVE PSET COMMENTS, OR GIVES US SOLUTIONS PLEASE. WE WANT TO GET BETTER



Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **5 (excellent)**

Generally good, but instructors have several times marked very correct proofs in this course as "completely wrong" with little to no explanation. (Other instructors later verified that the proofs were, indeed, correct). I recognize that it's hard to grade proofs though, which is why I marked the feedback as "excellent" anyway :)

Evaluate the course overall.: **1 (unsatisfactory)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

The grading seemed spotty and often was inconsistent with friends' results.

Evaluate the course overall.: **4 (very good)**

Feedback you received on work you produced in this course: **2 (fair)**

There was sometimes feedback that was incorrect

Evaluate the course overall.: **1 (unsatisfactory)**

Feedback you received on work you produced in this course: **1 (unsatisfactory)**

Regrade requests were all too common.

Evaluate the course overall.: **3 (good)**

Feedback you received on work you produced in this course: **2 (fair)**

At the start of the semester a lot of my work was misgraded and later corrected. Otherwise, fine.

Evaluate the course overall.: **5 (excellent)**

Feedback you received on work you produced in this course: **5 (excellent)**

I like the piazza list where it tells you exactly what you lost points for.



Section component of the course — Add Comments?

Course

Evaluate the course overall.: **3 (good)**

Section component of the course: **4 (very good)**

I found section really helpful as there usually weren't too many people there and we could ask clarifying questions on anything gone over in the book or lecture.

Evaluate the course overall.: **1 (unsatisfactory)**

Section component of the course: **5 (excellent)**

Section videos were the only things that were able to teach me the concepts

Evaluate the course overall.: **4 (very good)**

Section component of the course: **1 (unsatisfactory)**

It was hard to find a convenient section time. Also, the one section I went to didn't seem very helpful

Evaluate the course overall.: **4 (very good)**

Section component of the course: **4 (very good)**

I watched the recorded section, and Hikari was an effective presenter and went over the material well. Section handouts were useful summaries of the week's material, even if they often simplified material.

Evaluate the course overall.: **3 (good)**

Section component of the course: **2 (fair)**

More practice problems and work-throughs at the beginning would have changed this course for the better.

Evaluate the course overall.: **5 (excellent)**

Section component of the course: **3 (good)**

It would be awesome to have the advanced section follow the course material but go more in-depth into the current topics. It seemed useful for the students who had taken advanced math courses, but I felt that it moved too fast and ultimately stopped attending -- which is disappointing, since I felt like I understood the course material and wanted to go further but did not have this math background.

Evaluate the course overall.: **3 (good)**

Section component of the course: **4 (very good)**

Sections were helpful if you went to the right TF.



Evaluate the course overall.: **4 (very good)**
Section component of the course: **4 (very good)**
people should go to section more!!

Evaluate the course overall.: **4 (very good)**
Section component of the course: **5 (excellent)**
CS121.5 was very good. I Love Alec Sun.

Evaluate the course overall.: **1 (unsatisfactory)**
Section component of the course: **N/A**
I only went once and it was not helpful to me, but I see why it could be helpful.

Evaluate the course overall.: **3 (good)**
Section component of the course: **3 (good)**
Not enough section times.

Evaluate the course overall.: **3 (good)**
Section component of the course: **3 (good)**
Sections are OK, seem to be at a more surface level than the pssets/exams. Would be good to have a proofwriting section with some relevant practice problems they walk you through. They had one but it wasn't very helpful.

Evaluate the course overall.: **4 (very good)**
Section component of the course: **N/A**
I went to a couple advanced sections, which were great. I never attended section though.

Evaluate the course overall.: **4 (very good)**
Section component of the course: **4 (very good)**
CS121.5 was awesome.

Evaluate the course overall.: **2 (fair)**
Section component of the course: **3 (good)**
s/o Alexis Ross for saving my life

Evaluate the course overall.: **4 (very good)**
Section component of the course: **N/A**
There was a section, but I never went.



Evaluate the course overall.: **3 (good)**

Section component of the course: **5 (excellent)**

Hikari was amazing in her section. I'm so grateful she was the recorded section leader!!!

Evaluate the course overall.: **1 (unsatisfactory)**

Section component of the course: **3 (good)**

the tf's did work pretty hard

Evaluate the course overall.: **3 (good)**

Section component of the course: **3 (good)**

Unfortunately, I wasn't able to attend more than one section because all the ones I was available for were transformed into office hours. I understand why this happens, because students under-utilize section, and so TFs believe they're wasting time teaching section to an empty classroom or only a few students. However, speaking from my own past experience as a TF/CA, I find teaching small sections to be very rewarding, and usually there are a handful of students that show up every week and seem to really benefit from section. Therefore, I would request that if there are TFs willing to do so, then let them teach section despite small numbers.

Evaluate the course overall.: **5 (excellent)**

Section component of the course: **5 (excellent)**

The sections really helped me have a more thorough understanding of the material by answering specific student questions and giving us exercises to help us understand how things worked.

Evaluate the course overall.: **3 (good)**

Section component of the course: **5 (excellent)**

Section was nice even though I never went.

Evaluate the course overall.: **4 (very good)**

Section component of the course: **5 (excellent)**

I feel like this is a course where everyone should be assigned to a section with a pretty lenient attendance policy but still a strong push to attend. For a lot of people this is their first theory course and I feel like they'd really benefit from being held accountable for going to section.

Evaluate the course overall.: **1 (unsatisfactory)**

Section component of the course: **N/A**

What section?

Evaluate the course overall.: **4 (very good)**

Section component of the course: **3 (good)**

I felt that too many sections were converted into OH



Evaluate the course overall.: **1 (unsatisfactory)**

Section component of the course: **2 (fair)**

The TFs really try, and the example sometimes are helpful

Evaluate the course overall.: **4 (very good)**

Section component of the course: **5 (excellent)**

I really appreciate the CS 121.5

Evaluate the course overall.: **2 (fair)**

Section component of the course: **3 (good)**

Sometimes section was a little bit too general and didn't focus on what students were struggling with.

Evaluate the course overall.: **4 (very good)**

Section component of the course: **1 (unsatisfactory)**

Sections were cancelled early in the year and everything felt unstructured. I was unable to find section useful.

In your opinion, what preparation or background is necessary to take this course?

Course

Evaluate the course overall.: **3 (good)**

Background in proofs is so, so helpful. I took Math 101 in the spring, and although I don't think taking Math 101 is necessary, having some background and intuition on how proofs are structured is essential. I was concurrently taking Stat 110, which was really helpful when we got to probabilistic computation, but I think it's doable to learn the necessary statistics on the fly.

Evaluate the course overall.: **3 (good)**

A lot of math#####know proof-based math by heart before taking this

Evaluate the course overall.: **1 (unsatisfactory)**

None

Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **4 (very good)**

CS20 is helpful.

Evaluate the course overall.: **3 (good)**

None

Evaluate the course overall.: **4 (very good)**

A math background is more important than a CS background. The material is also very college-focused, i.e. high school students hoping to study ahead typically won't learn about this.

Evaluate the course overall.: **4 (very good)**

Good reading comprehension skills.

Evaluate the course overall.: **5 (excellent)**

CS50, 51



Evaluate the course overall.: **4 (very good)**

This class is very proof heavy, so you either need to be familiar with proofs or be prepared to catch up.

Evaluate the course overall.: **4 (very good)**

I took the course without official CS50 background (but with some coding experience from high school) and without Stat110. I have a pure math background, which was sufficient, but having taken 124 beforehand would have been immeasurably helpful.

Evaluate the course overall.: **3 (good)**

Knowledge of mathematical proofs and basic probability and set theory.

Evaluate the course overall.: **5 (excellent)**

A background in proof-writing, or natural aptitude for abstract mathematics, or both. Also, previous exposure to basic discrete mathematics, including modular arithmetic, elementary graph theory, etc. Although most of the problems on the exams and problem sets aren't actually that hard, they do require understanding of the often dense material.

Evaluate the course overall.: **2 (fair)**

How to write formal proofs

Evaluate the course overall.: **4 (very good)**

Ability to read math or have taken some sort of math class (CS20)

Evaluate the course overall.: **3 (good)**

take CS20 !!!

Evaluate the course overall.: **3 (good)**

I found Math 23a and c to be extremely helpful, mostly because a proof-based mathematical background can be very useful in much of the work.

Evaluate the course overall.: **3 (good)**

Nothing

Evaluate the course overall.: **2 (fair)**

CS20



Evaluate the course overall.: **5 (excellent)**

Some probability and combinatorics and some knowledge about functions (what it means to be injective, surjective, etc.)

Evaluate the course overall.: **5 (excellent)**

CS 20 for knowing how to write proofs and exposure to graphs. Probability background would be good because the probability review lecture was very rushed.

Evaluate the course overall.: **4 (very good)**

Proof-based mathematics or CS20

Evaluate the course overall.: **3 (good)**

CS 20 was marginally helpful (it was good to have written formal proofs before).

Evaluate the course overall.: **3 (good)**

The more CS background you have the better

Evaluate the course overall.: **2 (fair)**

Much more exposure to theoretical CS concepts than most students already have.

Evaluate the course overall.: **4 (very good)**

Proof based math is quite important.

Evaluate the course overall.: **3 (good)**

Some background in linear algebra, statistics, and writing proofs would all be helpful for this course.

Evaluate the course overall.: **4 (very good)**

Interest in CS. Basic proofs.

Evaluate the course overall.: **3 (good)**

Discrete math background helps, but self-studying is probably just as good as taking CS20

Evaluate the course overall.: **2 (fair)**

CS51. CS20 slightly helps (but not necessary)



Evaluate the course overall.: **4 (very good)**

Experience with discrete math, either by self studying or some previous class. OR proof-based math knowledge

Evaluate the course overall.: **3 (good)**

You should ideally take Math 23 or higher in order to develop rigorous proof-writing skills. Otherwise, be prepared to catch up.

Evaluate the course overall.: **1 (unsatisfactory)**

Proof-based math is nice, but as long as you do your psets with someone who is good at proofs, you will get the hang of it.

Evaluate the course overall.: **4 (very good)**

CS 124 was a help. Mathematical intuition and experience is important.

Evaluate the course overall.: **3 (good)**

mathematical proof background

Evaluate the course overall.: **1 (unsatisfactory)**

Intro CS, Probability (though not listed as prereq), Some proof based math

Evaluate the course overall.: **5 (excellent)**

mathematical maturity

Evaluate the course overall.: **1 (unsatisfactory)**

Proof math background absolutely! I think every student should have taken Math 23+ or 101 or have had proof experience in high school. You don't need to have taken CS classes though.

Evaluate the course overall.: **3 (good)**

Some theoretical math background

Evaluate the course overall.: **4 (very good)**

Knowledge of proofs -- induction, contradiction, etc, and familiarity with mathematical notation.

Evaluate the course overall.: **3 (good)**

CS50 and CS20. I know it says CS20 is optional but I found it difficult to catch up at the start.



Evaluate the course overall.: **3 (good)**

A lot of Math - A whole lot of Math and Proofs

Evaluate the course overall.: **1 (unsatisfactory)**

Stat 110, CS 124

Evaluate the course overall.: **3 (good)**

Some background in proofs.

Evaluate the course overall.: **3 (good)**

Taking concurrent with stat110 is helpful - otherwise CS20 definitely isn't a must if you are willing to work a little harder to catch up the first few weeks.

Evaluate the course overall.: **3 (good)**

Knowledge of probability (STAT110) and facility with mathematical reasoning and logic (CS20, MATH152, or a similar course).

Evaluate the course overall.: **3 (good)**

An understanding of how proofs work and how to write them.

Evaluate the course overall.: **3 (good)**

It can easily be done by someone who is at least at the level of multivariate calculus.

Evaluate the course overall.: **5 (excellent)**

Either some proofy background or going through the pre-course stuff at the end of the summer.

Evaluate the course overall.: **4 (very good)**

Know how to write proofs

Evaluate the course overall.: **2 (fair)**

Comfort with proofs and probability.

Evaluate the course overall.: **4 (very good)**

Proofs



Evaluate the course overall.: **4 (very good)**

None. In my opinion, this course should not be taken after CS124. It is too repetitive.

Evaluate the course overall.: **3 (good)**

Understanding more statistics would have been nice, also taking cs20 is basically necessary-it gives you good enough proof background to start off fine in the class.

Evaluate the course overall.: **4 (very good)**

Some CS and math experience.

Evaluate the course overall.: **1 (unsatisfactory)**

A lot of maturity with proofs and coming up with proofs. For reference, I got an A in Math25 and still found this class difficult just because of the way the class is taught and how the textbook is written.

Evaluate the course overall.: **2 (fair)**

Stat 110 and CS124 seriously helped

Evaluate the course overall.: **4 (very good)**

No coding necessary, nothing technically necessary but familiarity with proofs goes a long way.

Evaluate the course overall.: **3 (good)**

Having experience in discrete math, and perhaps even some in algorithms, is very helpful for having the intuition to solve some of the harder problems in this course

Evaluate the course overall.: **3 (good)**

Definitely knowledge of proofs is helpful. I took CS20 and felt underprepared. In CS20, you can get away with memorizing things, but in CS121 you need to focus on understanding all the material because it is impossible to apply things when you just memorize.

Evaluate the course overall.: **4 (very good)**

Experience with proof-based math (23+) is ideal because those classes tend to help people get used to common language and the types of thinking used in math courses.

Evaluate the course overall.: **5 (excellent)**

CS20 or proof knowledge



<p>Evaluate the course overall.: 1 (unsatisfactory) nothing will really prepare you for this course besides maybe having taken it already, or just being a genius</p>
<p>Evaluate the course overall.: 4 (very good) Proofs based math class and some knowledge of python/another coding language</p>
<p>Evaluate the course overall.: 5 (excellent) CS20 or proof experience, programming experience is useful</p>
<p>Evaluate the course overall.: 1 (unsatisfactory) Comfort with mathematical proofs and notation.</p>
<p>Evaluate the course overall.: 5 (excellent) Some higher level math (mostly for understanding basic proof techniques).</p>
<p>Evaluate the course overall.: 5 (excellent) CS 20</p>
<p>Evaluate the course overall.: 5 (excellent) Strictly, almost nothing is required. I knew no computer science prior to this course, and the course text provides an introduction to the mathematics required for the class (which I already had). I recommend above all else an advanced beginner's mathematical maturity. Knowing prior to the course what an injective function is of great benefit to any potential CS121 student.</p>
<p>Evaluate the course overall.: 3 (good) A basic understanding of mathematical proofs, basic familiarity with graph theory.</p>
<p>Evaluate the course overall.: 3 (good) Proof-based math along the lines of CS 20.</p>
<p>Evaluate the course overall.: 4 (very good) math 1b</p>
<p>Evaluate the course overall.: 2 (fair) Students should really take CS20, and even that probably won't be enough. Be very comfortable with proofs and reading mathematical texts (because the textbook is mostly math notation and it is not really at all explained)</p>

Evaluate the course overall.: **5 (excellent)**

Being comfortable with discrete math is necessary. I also think that having experience with formal proofs is helpful.

Evaluate the course overall.: **5 (excellent)**

proof-based math helps a lot

Evaluate the course overall.: **3 (good)**

Coding is not very necessary, and neither is proof writing. I'd never written a proof before, and I feel that I picked it up quickly; this wasn't what made the course difficult. Experience thinking about theory in math could be helpful, though I'm not sure that CS 20 is necessary (which I didn't take).

Evaluate the course overall.: **3 (good)**

Maybe CS20, but the online MIT course was sufficient background.

Evaluate the course overall.: **5 (excellent)**

Background in discrete mathematics or proofs is extremely helpful.

Evaluate the course overall.: **4 (very good)**

Some experience with proofs would definitely be helpful

Evaluate the course overall.: **4 (very good)**

Math 23A/B or similar level of proof-based math experience (maybe Math 101?). Anyone who has taken CS 124 should consider themselves prepared also (though I would recommend the same preparation for that class). Math 21A/B + CS 20 is NOT enough preparation and the main reason why everyone complains about this class being too hard.

Evaluate the course overall.: **4 (very good)**

CS20, any background on proofs and logic is helpful, and discrete math.

Evaluate the course overall.: **1 (unsatisfactory)**

Very little background is required.

Evaluate the course overall.: **5 (excellent)**

124 or cs20



Evaluate the course overall.: **4 (very good)**

Some sort of higher level math class such as Math 101 or CS20.

Evaluate the course overall.: **4 (very good)**

comfort with algebra, some exposure to proofs helpful but not necessary

Evaluate the course overall.: **4 (very good)**

Proof-based math would be helpful (I took CS20 and found it helpful.)

Evaluate the course overall.: **3 (good)**

Some background in proofs is quite helpful.

Evaluate the course overall.: **1 (unsatisfactory)**

If you've taken 124 or any proof-based math before this course will be a breeze.

Evaluate the course overall.: **4 (very good)**

Any sort of proof-based class (CS20, Math23/25) would have done wonders in preparing me for proofs. I had no proof background and felt grossly under prepared at the beginning.

Evaluate the course overall.: **3 (good)**

Definitely should take CS20 or have some formal training in proof based math. I think linear algebra also helps understanding the quantum stuff.

Evaluate the course overall.: **5 (excellent)**

Some familiarity with mathematical proof writing is helpful.

Evaluate the course overall.: **4 (very good)**

Some amount of proof based math is necessary; I already had that going in, and that was a big leg up in this class, considering that this is really a math class disguised as a programming class.

Evaluate the course overall.: **5 (excellent)**

just a bit of math knowledge. know what a bit is too

Evaluate the course overall.: **4 (very good)**

None, just basic proof writing.



Evaluate the course overall.: **4 (very good)**

An understanding of mathematical proofs at least at the level of CS20.

Evaluate the course overall.: **3 (good)**

Math/proofs background.

Evaluate the course overall.: **4 (very good)**

Proof based math class. Some programming experience

Evaluate the course overall.: **3 (good)**

Strong mathematics background in proofs

Evaluate the course overall.: **4 (very good)**

Math 23 (a/b/c) and/or CS20. Taking CS 124 before this class REALLY helped

Evaluate the course overall.: **4 (very good)**

A lot easier if you have exposure to mathematical proofs. If you've already taken 124 it's also much, much easier as a lot of the content covered here is also covered in 124.

Evaluate the course overall.: **4 (very good)**

I decided to toughen up and take this class without CS20 and zero background with computational theory or any proof knowledge. It was difficult at first but I kept pace with the readings and worked hard on the psets with a good pset partner so everything worked out and was enjoyable.

Evaluate the course overall.: **5 (excellent)**

A whole lot. Not only should you have a strong math foundation in general, but a pretty broad one at that. There are long and complicated proofs and algorithms making all kinds of arguments so you are expected to know quite a bit. For example, very few people know much graph theory (in general?) at this point, but it comes up quite a bit. Things like this happen often, and its part of what makes the course so rich. But don't show up to CS121 as your first math class.

Evaluate the course overall.: **3 (good)**

I took CS20 but it wasn't nearly enough preparation; I wish I was better prepared but I'm not sure how - I think this course just needs to be taught with more structure and explanation

Evaluate the course overall.: **1 (unsatisfactory)**

Serre Duality, Math 55 with Yum-Tong Siu, Lubin-Tate Theory, and math 233a, preferably Arnav Tripathy's version.



Evaluate the course overall.: **4 (very good)**

Lots of proof-based math (23abc or above). You absolutely need this. Some knowledge of and experience with coding (CS50 or equivalent) is useful. Ideally even some discrete math. Taking Stat 110 concurrently is helpful later on. But generally just have to be extremely comfortable doing proof-based math.

Evaluate the course overall.: **5 (excellent)**

Familiarity with proofs and rigor, such as a proof-based math course or CS 20.

Evaluate the course overall.: **1 (unsatisfactory)**

One previous course in computer science

Evaluate the course overall.: **3 (good)**

CS20 was definitely helpful as well as programming knowledge especially in Python

Evaluate the course overall.: **4 (very good)**

none

Evaluate the course overall.: **4 (very good)**

You should have knowledge of proofs from CS20 or a similar course. Proofs and logic are essential to this course, so if you haven't taken discrete math you're gonna have a hard time working through the material.

Evaluate the course overall.: **1 (unsatisfactory)**

You need to have a strong background in proof-based math and it helps to have some understanding of linear algebra.

Evaluate the course overall.: **4 (very good)**

Some background in proofs (summer self-study was fine for me)

Evaluate the course overall.: **1 (unsatisfactory)**

An understanding of CS notation, understanding things like sets, subsets, big oh, and finding algorithm run time are all strongly recommended.

Evaluate the course overall.: **4 (very good)**

discrete math



Evaluate the course overall.: **3 (good)**

TAKE CS20, take a theoretical math class before cs121. This is brutal without this background. I didn't take CS20, just took Math 21ab, DIED because I didn't have the math background. People told me I would be okay without CS20, but they were wrong.

Evaluate the course overall.: **3 (good)**

I think CS 20 helps, but a background in proof and logic is also helpful. Also general CS knowledge can be very useful for understanding material.

Evaluate the course overall.: **2 (fair)**

the required pre req cs courses and potentially cs20

Evaluate the course overall.: **2 (fair)**

Proof based math

Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **3 (good)**

A solid background in proofs, ideally at a higher level than CS 20. You can certainly get by with CS 20 or even without it (grades-wise, that is), but I don't think you'll get as much out of it. The more comfortable you are with proofs/more math background you've had, the better you'll be able to fill in the gaps where the textbook/lecture are either hand-wavy or unclear.

Evaluate the course overall.: **2 (fair)**

None

Evaluate the course overall.: **4 (very good)**

If you are newish to CS and considering taking this class, read this review! I was in your situation. The tl;dr is as following: you do not need CS20 and if you do most of the background prep, you will be ok. The longer version is this. The stuff you do in CS121 is quite particular to 121, so having a super strong background in discrete math for example from CS20 actually will only translate to your work in the class to a small degree. What you do need to know, from the background prep page, is basic familiarity with graph theory (just like terms and basic theorems), set notation, logical expressions notation, proofs by induction, and probably a couple more things that I am forgetting now, but honestly, if you want to take this course without having any background in anything, I really believe that is doable (that was what I did). I would say you should allot about a week to doing the background prep if you can. If not, you will still probably be fine but there will be a more steep learning curve. And don't freak out about seeing crazy math notation from day 1 in the course! Everyone is in the same boat as you and will feel similarly overwhelmed, but it's not actually so bad when you get down to it. Enjoy it!



Evaluate the course overall.: **5 (excellent)**

Proof knowledge but accessible to people only moderately mathematically inclined.

Evaluate the course overall.: **4 (very good)**

Proof-based math

Evaluate the course overall.: **1 (unsatisfactory)**

Lots of programming experience. The math was not too bad, the big thing was having intuition about how programs work.

Evaluate the course overall.: **4 (very good)**

I believe this course does a great job of being accessible to those of any background. That said, a background in proof-based math or computer science will make the coursework substantially easier.

Evaluate the course overall.: **1 (unsatisfactory)**

CS20 kinds of helps with proofs.

Evaluate the course overall.: **1 (unsatisfactory)**

Optimally, some math/proof/tcs background so the course doesn't give you the wrong ideas.

Evaluate the course overall.: **1 (unsatisfactory)**

Definitely some proof experience. To be honest, you need all the preparation you can get.

Evaluate the course overall.: **3 (good)**

General exposure to CS ideas and a significant degree of mathematical maturity and experience with proofs.

Evaluate the course overall.: **5 (excellent)**

a 100 level math class or math > 21

Evaluate the course overall.: **4 (very good)**

CS20 is helpful, but didnt really help prepare much for CS121

Evaluate the course overall.: **2 (fair)**

You need a LOT of experience with proofs and probably CS 51 or higher



Evaluate the course overall.: **4 (very good)**

math background, rather than cs.

What would you like to tell future students about this class?

Course

Evaluate the course overall.: **3 (good)**

The best thing about this course is that Professor Barak is really trying his best to make this course the best that it can be. That being said, I found much of the pedagogy quite frustrating. Reading and understanding the textbook is central to this course, but if you don't have much math background the textbook is quite hard to parse, and I found that there wasn't much support to help with this in lecture. Sometimes Professor Barak assumes that a lack of questions means that you understand everything, and so I recommend asking questions whenever you have them, as Professor Barak is usually really good about answering them once questions are on the table. The pssets are also done in partner pairs, and so finding a good partner is crucial and will save you so much time.

Evaluate the course overall.: **1 (unsatisfactory)**

Be prepared for really bad teaching by the professor and not much help from TFs

Evaluate the course overall.: **4 (very good)**

This course is very difficult, but it is do-able if you put in the effort. The course material is really interesting, but at times it is really hard to understand.

Evaluate the course overall.: **4 (very good)**

This class wasn't too bad, and that was in large part because I had a wonderful partner and collaborators. Having other people to talk about the problems with can make it so much easier and more enjoyable. Boaz is a good lecturer, I just wish he could present the information in a more accessible way because it was often really hard to follow. There are lots of opportunities to make up for lost points on the homework, and you only need to get 50% of the questions on the reading quizzes correct to get full credit. I really appreciated this because it took the pressure off.

Evaluate the course overall.: **3 (good)**

Well, it's required, but here goes. Starting with the textbook. It's unapologetically an unfinished product (in a bad way); typos and "TODO" are frequent. That could be forgiven except the text seems to go out of its way to explain something in the most convoluted way possible at times, assuming widely varying levels of background for each topic. It's only saving grace is that it tries to give "easy-to-understand" "Proof ideas" before many proofs and these are usually at least somewhat helpful. Next, you've probably heard of NAND by now. The text and this course build up an entire theory/model of computation around NAND under the premise that such a theory/model is easier to comprehend or more intuitive than traditional Turing machines. That's fine (whether you agree or not). What's not fine, is that it comes at the cost of getting familiar with the traditional model which is used literally everywhere else (in papers, at conferences, etc.) Sure, Turing machines are covered, but to a very little extent. Regarding problem sets, they're weekly and they're difficult. You do get to work with a partner, but the course's collaboration policy is severely out-of-whack, necessitating "declaring" partners beforehand and only allowing any collaboration whatsoever between two pairs. Most other CS courses here have much more relaxed policies encouraging a much more collaborative and supportive atmosphere that is sorely lacking from CS 121. Probably the worst aspect of this class is that it expects you to treat it differently (e.g. somewhat expecting you to "prepare" over the summer with "problem set 0", etc. etc.). This sort of mentality unfortunately permeates and gives the course a bad vibe.

Evaluate the course overall.: **3 (good)**

You have to take it

Evaluate the course overall.: **4 (very good)**

The subjects covered are extremely interesting and I had not encountered most of them in any other computer science classes I have taken. However, the concepts are very difficult and often confusing. Be sure to give yourself lots of time to review the material and do the pset every week. I found it necessary to review the textbook over and over again. Also, there's almost no programming involved whatsoever.

Evaluate the course overall.: **4 (very good)**

If you don't need to take this class for the CS concentration, only take this if you are really interested in the material. It is a fair class but there is room for improvement with regards to elegance/productive use for assigned problems. Readings take longer than you may at first expect, and doing them is important for success in this class. Watch the HW postmortem reviews!

Evaluate the course overall.: **5 (excellent)**

It's useful to know that this class is very bimodal. Some people are in it just because of the requirement, and if they haven't done any formal math before, then they can tend to struggle. Others are in it because they're actually really interested/experienced in theoretical math and they do fine. In general, you should always do the readings, because most of the problems on the problem sets and exams are more about parsing the definitions of all the terms and figuring out what you're being asked to prove. Once you understand exactly what the problem is asking you to show, typically you only need to do a small amount of actual manipulation or work with the concepts. Again, don't be discouraged if the lectures and readings are difficult. If he proves some sophisticated theorem, he's not going to ask you to reproduce the proof on a problem set, you just have to be aware that the theorem is true and maybe know a couple of trivial consequences.

Evaluate the course overall.: **4 (very good)**

In the problem sets, your proofs probably don't need to be as detailed as you think. When they say assume an intelligent reader they mean it, you can gloss over many simple/obvious steps. Figure out how thorough you actually need to be. Also, always check over your graded work and make use of regrade requests when applicable.

Evaluate the course overall.: **4 (very good)**

Boaz is a good professor who is committed to making the class accessible for students. Unlike CS124 professors who try to make things as hard as possible, his exams literally take questions from the homework or review, so if you study hard, you can do well. Only caveat is to actually abide by the math pre-reqs, it is HARD to do well in this class if you cannot read the mathematical notation in the textbook and understand it intuitively.

Evaluate the course overall.: **3 (good)**

I have so many emotions about this class. Shoutout to Eric Lu, absolute angel and love of my life. Really really ready yourself to find help in this class because NO ONE will have any clue what is going on. Honestly though, I am leaving the class with more positive emotions than expected. I suffered, but I learned a lot :)

Evaluate the course overall.: **3 (good)**

I can't say I would recommend it to those who aren't studying CS.

Evaluate the course overall.: **3 (good)**

The coursework is interesting but can get tedious at times.

Evaluate the course overall.: **5 (excellent)**

This class provides a nice introduction to theoretical CS. Boaz has made some large modifications to the "classic" style of teaching theory which I think overall were improvements. We use NAND machines to simulate computation instead of Turing machines, which I thought was much more applicable to the real world since NAND is a better model of real computers. As someone who mostly tries to stay away from theory, I thought CS121 was a relatively enjoyable way to learn some foundational concepts. The course infrastructure was great and the problem sets weren't overly demanding. At times though the notation was too heavy and the topics at the end weren't given enough time so I felt like I didn't really learn them. Overall a solid class and I'm sure it will continue to improve as Boaz seems very dedicated to his teaching.

Evaluate the course overall.: **5 (excellent)**

The content in this course was really interesting, and the TFs really try their best. Reading the textbook is a better way to understand the material than lectures, which were often confusing. The psets are manageable if you go the office hours early in the week (especially the weekend night ones) since they're a lot less packed. Overall, the course is hard but you get a lot out of it

Evaluate the course overall.: **4 (very good)**

Get a pset partner and always attempt the extra credit problems!

Evaluate the course overall.: **4 (very good)**

I know everyone hates this class, but this was the first computer science class (I took cs50 and cs51) where I felt really excited about the material. I didn't have a strong math background before coming in (didn't take any proof based math, or cs20), so I wasn't looking forwards to the class, but it ended up being my favorite class this semester (then again, I also had a great pset partner I worked together really well with). Boaz is very funny (imo) and responsive, especially on piazza, so don't be afraid to reach out for help, and ask clarifying questions. Reading the textbook made me interested in other math subjects as well.

Evaluate the course overall.: **3 (good)**

It's clear that Boaz really cares about this course and he is actively working to make it better. The textbook can be super confusing in some places, and re-re-reading is sometimes needed. The same goes for lecture and problem sets - they can be super hard to understand sometimes. TF's really shine here - they are more than happy to go over something in lecture and put it in different words. Personally, lecture was not that helpful for me as I would understand the material better through reading multiple times. Tips for lecture and reading - make sure you do it, read slowly, and ask questions. Problem sets are definitely difficult, and sometimes very hard to understand what the question was asking. Don't be afraid to ask for help here - even in just what the question is asking. What makes some of these problem sets difficult is that they are hard to understand. Tips for problem sets - don't be afraid to go to office hours and ask questions. Overall - theoretical CS isn't easy, but it is weird aspects of the class that make things harder. The odd pset collaboration policy, indecipherable readings, lectures, psets, and piazza posts don't help the situation; hopefully, Boaz works to improve these aspects in the future.

Evaluate the course overall.: **2 (fair)**

Avoid taking this class if you can (it is a requirement for CS concentrators) and definitely not as an elective - the material is interesting but the structure and pace make the class an unenjoyable learning experience. Boaz is a nice person but not an excellent lecturer.

Evaluate the course overall.: **3 (good)**

The material is interesting, but the readings are dense and can be hard to get through. It might be helpful to start a reading guide while you read that you can study from for the midterms. The problem sets really help clear up the important topics. Sections and section notes are helpful. Ask questions.

Evaluate the course overall.: **4 (very good)**

Good course, a lot of people don't like it because it's required and not practical. I came in thinking it was purely for a requirement and a way to learn proofs before taking 124, and fulfilled those goals but also found it super fun and learned a lot. There are few professors who care as much about their courses as Boaz does, and it shows in the amount of time and effort he puts into the course. Given that, lectures are boring and I probably retained close to 0 from them despite always going. Read the book and understand the theorems, that's what actually helps. Definitely reach out to him / TFs if you need help. Don't take CS20 before this, it's not needed, you'll be fine coming in even if you meant to study over summer but didn't (that's what happened to me but proofs really aren't as hard as you might expect if you put in the effort to learn at the start).

Evaluate the course overall.: **3 (good)**

Keep up with the readings and go to section -- you will be in much better shape for the exams that way. The textbook, lectures, and review materials are confusing and hard to follow, but TFs can be really helpful if you reach out. I found this course conceptually difficult because it's so abstract, and because there's honestly no way to be "ready" for a proof on an exam -- they're all different, so you just have to think things through on the spot. It's really important to spend time on the psets and practice problems so that proofs come more naturally. But I think it's still a tough course no matter what, and unfortunately the textbook and lectures don't help much :(

Evaluate the course overall.: **4 (very good)**

CS 121 is really hard, but rewarding because the content is extremely interesting once you start understanding it. This class is about what computation is (did you know we can express every finite computation using just simple computational gates? That's so cool!) as well as what algorithms are, how to define runtime, what problems are solvable in certain runtimes, etc. You also learn about more advanced but still fascinating topics like randomized computation, cryptography, proofs, and quantum computing. Long story short there - the concepts are really cool, and very foundational to computer science. Machine learning will probably be extremely different 10 years from now, but theoretical CS won't have changed much. Other pieces of advice - ACTUALLY do all the readings. Put in the work EARLY on and rethink concepts until you understand them. It takes time to understand things, and before hating on theoretical CS you owe it to yourself to put in your best effort to understand things. Go to office hours, go to section, go to lecture, and ask any questions you have on Piazza. The course staff in this class are some of the most talented, kind, people in CS and they care about helping you learn. Also, do NOT rely on office hours or your partner when it comes to doing psets! I made this mistake in the beginning, so please trust me on this. Try to do all of the problems yourself before you meet with your partner, and before going to OH try discussing and struggling over the pset with your partner first. It's so rewarding when you pinpoint exactly what your confusions are, and then get those specific confusions answered at OH instead of just asking the TF to tell you the answers. Keep at it! The moment you finally understand a difficult concept, or the moment everything you've been learning finally comes together, will be the moment you fall in love with this class.



Evaluate the course overall.: **1 (unsatisfactory)**

Do not take this course unless you want a degree in CS... if you want a degree in CS, buckle up because its going to be one hell of a ride.

Evaluate the course overall.: **4 (very good)**

Try to make sure you are understanding the material - it is possible to get by without really learning it, which is bad.

Evaluate the course overall.: **1 (unsatisfactory)**

I would suggest not taking this course, but unfortunately it is required for concentrators. The concepts are actually very interesting, so I would suggest trying to focus on that and exploring the material apart from the course text.

Evaluate the course overall.: **5 (excellent)**

I love this course! Boaz is a smart Jewish guy, and CA's are helpful. The course covers fascinating material in a non-traditional and modern manner. You may face some difficulties talking to outsiders about the NAND<< language because they probably don't know how those results correspond to traditional Turing Machine based proofs. But take this course to know why computation is cool, and how our universe even allows us to simulate/compute!

Evaluate the course overall.: **1 (unsatisfactory)**

I was extremely disappointed with this class. I looked at the material that would be covered and thought, "This looks really cool! I can't wait!" Some other students had mentioned they didn't like the class, but I just assumed that was only because they weren't math people. To put it bluntly, despite coming to the class expecting it to be my favorite this semester, I left with it as my absolutely LEAST favorite class. It's a huge time sink to understand readings and lectures for the sake of "rigor" despite similar rigor being capable with less convoluted notation and explanation. It becomes so easy to get lost in this "computation" world that any resemblance to practicality becomes lost. If you're not a CS concentrator, I would recommend avoiding this class, and for CS concentrators, wait until your senior year so that more years of experience may help this class become better run and taught before you take it. The material is still pretty interesting though.

Evaluate the course overall.: **3 (good)**

The second half of the course is definitely more interesting than the first. Boaz cares a lot about his students, and while there were some bumps this year, I'm sure it'll be better in the years to come.

Evaluate the course overall.: **4 (very good)**

Really interesting topic that bridges mathematical concepts with computer applications. The material is often hard to understand, so it's easy to stop paying attention when some part makes sense, but then new material becomes a huge hassle to understand. It's also really easy to lose points on tests because of small mistakes (because partial credit is hard to get), which can be pretty frustrating. Overall though, a very cool class!

Evaluate the course overall.: **3 (good)**

Okay so you've probably just read a tonne of evaluations moaning about this class and how you have to take it. I kind of agree but at heart I don't really. It's not as bad as it's made out to be. I'm not a CS concentrator so can hopefully give an outside perspective. **THE GOOD:** I enjoy doing the problem sets. Clearly I say that in hindsight but they really do help me to understand the material much better. The subject matter is also really interesting; Can a function be computed? How quickly? Would it be better if it was random? Could I make the result secret? What if I had a quantum computer? Really cool! I also like piazza and the responses are really great and useful. I also think the book is an excellent resource. **THE BAD:** I didn't enjoy the lectures too much. Boaz knows his stuff but I really don't find him that engaging. I also felt that the timing on the midterms made it difficult to finish and also consider your answers. However the final was much better with this so I think Boaz could be fixing this. **VERDICT:** If you're a CS concentrator then you've got to take this but if you approach the course by getting excited by the material I guarantee that you'll enjoy it a lot more :) If you're NOT a CS concentrator then I do recommend that you take this course. It gives a great introduction to doing theoretical computer science and some of the results are really amazing! Be warned it'll take a lot of time and hard work but you'll come out glad that you took it.

Evaluate the course overall.: **3 (good)**

I did not like this course that much. Theoretical Computer Science is not at all engaging to me. I had to take it which I am guessing you will have to do. Just do your best

Evaluate the course overall.: **1 (unsatisfactory)**

If you have to take this course, you're stuck. Otherwise, avoid.

Evaluate the course overall.: **3 (good)**

Course is not that bad overall. The lectures are fairly useless unfortunately, but the textbook is very thorough, and even though there are typos it contains a lot of great information.

Evaluate the course overall.: **3 (good)**

A good overview of important theoretical CS concepts. However, it was very difficult to parse the textbook, and often Boaz uses mathematical symbols to describe concepts without first giving a big-picture overview or example to give us an idea of what the concept is actually trying to convey. So the teaching could be a lot better. Take if you're a strong self learner.

Evaluate the course overall.: **3 (good)**

I really don't see a reason to take this course if you are not planning to concentrate in CS.

Evaluate the course overall.: **3 (good)**

Overall, this course isn't bad. However, the parts that are bad are primarily in the execution of the teaching surrounding the course content. The course content itself is quite interesting and forces the student to learn how to reason about objects with very little information about the objects in question, among other things. However, the bad parts of this course are in the teaching of the course. Namely, the fact that NAND is the primary computational model for the course and the fact that the only course material that uses NAND are Boaz's lecture notes, which are very much in need of development. Beware: this is a course in development, and it shows. While I wouldn't outright say not to take this class, I would advise seriously exploring the option taking the equivalent course at MIT for the same credit. CS121 may become the great course that it can be, but it still needs some time to grow before it gets there.



Evaluate the course overall.: **3 (good)**

You have to take it for a CS concentration, but I wouldn't recommend it to anyone else. If you can write proofs well you'll be fine. The problem I found is that it's very easy for those who can write proofs and very hard for those who can't.

Evaluate the course overall.: **1 (unsatisfactory)**

As a CS major, you will take this course. Although I learned enough to find the content interesting, the manner in which it is presented is often obtuse and difficult to digest. Make sure you look through your graded work as grading errors occur frequently.

Evaluate the course overall.: **5 (excellent)**

Boaz is a badaz. It's a good class--my favorite this semester. You'll get comfy with proofs. I recommend doing the summer pre-course stuff early (though you're probably reading this too late) and doing pssets in your own and only CHECKING with a partner.

Evaluate the course overall.: **4 (very good)**

Easy set theory gen ed.

Evaluate the course overall.: **2 (fair)**

This course is a CS124 workload with almost none of the reward for the standard CS student. If you're genuinely interested in theoretical computer science, you'll have more than enough to satisfy your appetite. If you're not, this course is brutal in terms of content and time commitment and will almost certainly devour your semester. There's no issue with a difficult course, but the fact that CS121 in its current form is required is a sore point. Most of the topics in the course have very specific applications that are not relevant to the majority of work computer science students end up doing, and unfortunately they are not well taught in either lectures or the textbooks. Concepts that are explained quite simply in other theoretical CS courses are mangled and laden with dense mathematical formulas and notation, and to make matters worse are seemingly written intentionally to be distinct and at times incompatible with the wealth of theoretical CS literature already out there. Lecture, which was designed to clarify the textbook readings, ends up doing the opposite, leaving a tremendous burden on the course's teaching fellows who themselves often struggle with the material at office hours. This course is in need of either serious reworks or a change to elective status, but so long as Boaz has any say, it is doubtful that will happen.

Evaluate the course overall.: **4 (very good)**

Some parts of the class are quite interesting, but other parts you just have to get through. Problem sets and exams are pretty fair, but you'll have to do some self-learning since lecture alone isn't enough to get by with. Overall would recommend, even if it isn't a requirement, if you're at all interested in theoretical CS.

Evaluate the course overall.: **4 (very good)**

This course is a nice introduction to a lot of interesting ideas in theoretical computer science. The readings are great and you will probably be exposed to a lot of cool ideas. However, it is a bit slow moving at times. I would not recommend taking CS124 before this course because that sequence causes CS121 to feel quite slow and repetitive.

Evaluate the course overall.: **2 (fair)**

If you are not well versed in math proofs or theoretical computer science, be prepared to be confused for a majority of the semester. The professor and his textbook are not very helpful in making things any clearer either; would recommend asking TFs for help

Evaluate the course overall.: **3 (good)**

Honestly I thought this was a bit over hyped in terms of workload, there is no coding so you can never get stuck on a small bug for multiple hours, and they are pretty lenient with grading on psets. However, the psets are poorly written, and I often found myself needing to go to office hours in order to even understand what the questions were even asking. If you think you are gonna take this course, FIND YOURSELF A GOOD PSET BUDDY. Find someone who you can spend many hours with and not hate, and someone you can keep accountable who can also keep you accountable. Attempting to do this course alone would have been horrible, and also not worth it to "learn the material better" because a lot of the questions on the psets are harder than the exams. If you get the big ideas on the psets, you should be fine because the exams kind of reiterate those. In terms of the class itself, boaz is not a good lecturer, because he spends too much time going off the deep end in terms of technical things that are mostly irrelevant to the main ideas, and will sometimes speed through the big picture so he can spend time on these less relevant details. It's kind of annoying that he required attendance in class, because I found myself zoning out after 15-30 minutes because the lecture became almost irrelevant, or half relevant and half not, but you can't tell which is which because he doesn't tell you when he's going into unnecessary detail. Furthermore, the "lecture notes" aka his textbook, is similarly written, and has some mistakes, and he explained everything very mathily and often leaves out the plain english version of the theorems he is trying to teach, making it hard to understand.

Evaluate the course overall.: **1 (unsatisfactory)**

You probably have to take this class because it's a CS requirement, but just be warned that you'll be doing a lot of self-studying for this class.

Evaluate the course overall.: **2 (fair)**

This class is a bit of a mess, so I wouldn't recommend taking it unless you have to (which all CS majors do). Relevant information is spread across a ridiculous range of sources--piazza, canvas, the course website, the gcal, a random link with lecture slides, GitHub, gradescope--making it quite difficult to find relevant information. Sometimes important information would be posted on piazza and wouldn't even be stickied! Also, be vigilant about the grades you get--I had to request regrades on many of the psets and exams for grading mistakes that I found pretty clear, to which I would have been more sympathetic if better feedback was actually provided as to why points were deducted, versus the many arbitrary "-3, points adjustment"s. To its credit, the course basically lets you create your own curve by supplying a TON of extra credit on both exams and problem sets, so while this creates much more work for you (as since it's very possible to get over 100% in the course, I'm guessing there is no curve applied despite median exam scores being in the 70's) there's also no excuse to not get a good grade in the class. Bottom line, I'd recommend wait until having taken CS124 (and STAT110) if possible, as I found the course manageable after having taken these course but would have been VERY lost without this knowledge (and this class doesn't really prepare you for CS124, as the rigor of that course is so much higher anyway). That being said, much of the material is interesting if you can figure out what the textbook is saying (a nontrivial feat) besides a few seemingly arbitrary points of memorization (I'm still unsure why we care how many NAND gates it takes to create a certain program) and as long as you're on top of the ever-changing office hours schedule (check the Gcal--found myself going to a nonexistent Dunseter office hours two weeks in a row because the stickied post on Piazza wasn't updated) there's no reason why you shouldn't walk away unscathed.



Evaluate the course overall.: **4 (very good)**

The best thing about CS121 is the material-I'm a math concentrator, but I chose to take this class because you learn some seriously interesting stuff. I enjoyed the readings and the homework because it involved solving a lot of fun puzzles. Because I had experience with math, the class was pretty easy, but I can imagine if you only do cs and don't have proof experience, it's a lot harder (especially since Boaz isn't great at explaining proofs and proof methods...) In general, since the class is big there are a ton of resources like piazza, sections, office hours, recorded lectures, etc., which seems really helpful (though I didn't really take advantage of them). The biggest downside are the lectures themselves, which aren't super well taught (and often go over things that aren't crucial to know). You'd probably be better off just reading the textbook thoroughly.

Evaluate the course overall.: **3 (good)**

I went into CS121 not really knowing what to expect in terms of difficulty. I found that it was really difficult and completing the psets were really challenging because I wasn't understanding any of the material. I recommend using sections and office hours with TFs because they are amazing! Also, make sure to understand all of the material over memorizing things because you can't get away with memorization in this course. Even though this class was hard, I learned a tremendous amount and the material was really interesting.

Evaluate the course overall.: **4 (very good)**

This class is very reasonable if you're used to proof based math, although many people who do not have this background have a more difficult time. Anyway, I recommend that you do as many of the homework and exam prep problems by yourself as you can before going to partners and office hours, because I believe doing problems in this class is very very helpful for understanding the material even (especially?) if you find the lecture notes confusing. I would recommend this for most other CS and math classes as well.

Evaluate the course overall.: **1 (unsatisfactory)**

if you're taking this class, you don't have a choice, but don't assume things are being done well or will be done well and that it is just you who is struggling. the class is notorious for a reason, and that is not necessarily in terms of the material

Evaluate the course overall.: **4 (very good)**

If you take the class hoping to get something out of it, you will. Go to section and to lecture; they were some of the most underutilized parts of the class.

Evaluate the course overall.: **5 (excellent)**

You're taking this because is required. It's not going to be useful in industry, but it's very interesting material treated in a very rigorous way. Not the worst class, very manageable with a good pset partner and time to give to the material.

Evaluate the course overall.: **5 (excellent)**

CS121 is one of the best courses I have ever taken at Harvard. The material is genuinely interesting, and the problem sets are fun. Also, the TFs and Boaz are extremely helpful and knowledgeable, and always responded to my Piazza questions. My only suggestions are 1) Don't use lecture time to have students do exercises they can do on their own outside of class. Sometimes the majority of a single lecture consisted of "doing exercises with a neighbor." 2) The book has a unique approach (using NAND) to explaining theoretical CS topics, which I found to be ingenious and very helpful in understanding theoretical CS topics. However, I think there should be slightly more emphasis on connecting this new approach to describing theoretical CS topics to traditional approaches/language (like Turing machines, tapes, etc.).

Evaluate the course overall.: **3 (good)**

I think the latter end of the course was relatively interesting and useful, but the first half of the class (material up until midterm 1) was confusing and too detailed. This class is good if you are interested in the foundations of theoretical CS, but if you're not, then this class will get very tiresome and uninteresting until the last couple of lectures. Also, I think the exams were not representative enough of one's ability to understand the material, but more so the ability to answer those specific questions (mainly referring to the open ended questions, specific reductions, etc.) Problem set grades are generous but the problems seemed very repetitive and time-intensive; be prepared to type up the same proof format many times

Evaluate the course overall.: **5 (excellent)**

This is a challenging class, but if you put in the time it deserves it is fantastic and you will learn a ton (coming from someone who isn't a "theory person"). Boaz really cares about the students and their learning (he's on piazza really frequently, ask any question you have!), and he actually responds to feedback. There are plenty of resources to help you succeed (tons of office hours and sections with recorded sections and homework/midterm postmortems). You should definitely have taken CS20 or have similar background. Regarding course advice, I'd recommend every time you do one of the readings set aside a large chunk of time for it. Doing well in this class really starts with taking the time to closely read and understand the material in the textbook. The problem sets are critical to cementing your understanding of the course material and preparing for exams. My biggest piece of advice is to actively work on EVERY non-bonus problem. You should never split work on these with your partner; that will ultimately hurt both of you when exams come around. After you've tried working on all of the problems, office hours are super helpful to make any last jumps to solving the problems (there are some problems that I would definitely not have solved without some help from a TF). There's so much bonus on the homeworks that you really don't need to worry about getting minor things wrong as long as you understand the core concepts. If you've done all of the homework problems and gone back and reviewed them you should be in good shape for the exams.

Evaluate the course overall.: **5 (excellent)**

This is an extremely well-run class that rewards its hardest-working students substantially more than those who put in moderate effort. Also, a number of students in this class, usually through anonymous Piazza comments, have adopted victim complexes in confronting the difficulty of this course, so be wary of course reviews that bash this class too enthusiastically. I come from the land of physics, so no one forced me to take this class to graduate, and I bear no prejudice toward it. My personal recommendation is that you enter this class with most of the required mathematical background---sets, functions, etc. A little bit of mathematical maturity will go a long way in the class, since it will enable you to focus on the CS-theoretic concepts without getting too bogged down in their mathematical details. Overall, the (free!) course materials, staff availability, and flipped classroom style of CS121 under Barak make this an excellent, even if occasionally painful, way to learn the rudiments of theoretical computer science.



Evaluate the course overall.: **3 (good)**

To be honest, I thought it would be a lot worse based on what I heard last year, which makes me hope it will continue to improve next year. This is Boaz's second time teaching the course. He's not the strongest lecturer, and sometimes presents stuff that seems tangential/doesn't really help us learn the material at all. Even though lectures have the polls that are most likely used for attendance, a sizable portion of the class doesn't attend, which Boaz himself acknowledges, so unclear on whether attendance is mandatory. Section and office hours are fine but I didn't use them too much. Office hours in particular were a bit too hectic for me, especially because under the collaboration policy you're not supposed to ask your classmates to explain what the TF just told them, so you have to wait in a long line for the TF. Pick a good pset partner because you'll probably be spending most of your time bouncing ideas off them. That being said, if you do the homework thoroughly, then you shouldn't have trouble on the midterms or final, because they have very similar problems. Budget your time!! Even if Boaz says there will be ample time, budget anyway!! Overall a decent course but really hoping the lectures and textbook will keep improving, because as they stand they're acceptable but not amazing.

Evaluate the course overall.: **4 (very good)**

Do the psets yourself. I know that sounds ridiculous and overly challenging but trust me. You will understand the material so much better. Use collaborators only to "check answers" write everything up yourself. If you do that. YOu will get an A in this course. I guarantee. Letting your partner do work only hurts you in the long run.

Evaluate the course overall.: **2 (fair)**

Realistically, there are a lot of ways this course could be improved, an especially important task given that the course is required for students to concentrate in computer science and has in many ways become the "weed-them-out" class. It is extremely math intensive, and even with CS20, many students are not adequately prepared for the mathematical background. The solution here is probably twofold: increase the rigor of CS20, and lower the level of CS121. At the very least there needs to be more communication between the two courses so that expectations are clear. The textbook is the next aspect where there is major room for improvement. To its credit, it is written in a fairly lighthearted manner and incorporates a good amount of humor. However, this does not make up for the fact that often it assumes a much higher level of understanding than most students have, and then as a result, leaves complicated topics and proofs as "exercises to the reader." For a student who has already not understood the very rough outline of a topic, it is very unlikely that he or she will be able to complete this "exercise," and so as a result, the concept is not well learned. There is nothing wrong with being explicit in the explanations and descriptions; one might even argue that this is the primary purpose of a textbook. There is room for improvement in lecture as well. At least in this iteration, there was very little additional learning that happened in lecture. If anything, lecture was in many ways very disheartening, for example when exercises would be presented, answered by one of the five same (brilliant) students, and then not explained. Exercises like these are probably best left for section (see the next paragraph), but if they must be included in lecture, they should certainly be explained. More than anything, lecture suffered from an at time insensitive use of language, labeling complex topics as "easy" or "not that hard," which merely has the effect of alienating or discouraging students who are struggling. A more conscious effort on the part of the professor to be mindful of language such as this would be a huge and much appreciated improvement. Finally, section was probably an underutilized yet invaluable tool. It might be a good idea to make section mandatory, perhaps at least until the first midterm, so that students are forced to make use of the resources given to them. Given that this is a gateway class to the cs concentration, taken primarily by sophomores (who really have only had two semesters of college under their belts), extra guidance can only be a benefit. So go to section.

Evaluate the course overall.: **5 (excellent)**

I think a strong math background, especially in proof-based math, is very helpful for this course. The assignments are challenging, but you will learn a lot from the course as long as you take the time to do the assignments by yourself before referring to your partner and read the textbook thoroughly. You should actually do every reading assignment thoroughly if you want to keep up with the course and get more out of the lectures.



Evaluate the course overall.: **1 (unsatisfactory)**

I hate to be the bearer of bad news, since 121 is required, but this class WILL MAKE YOU HATE COMPUTER SCIENCE AT HARVARD. As a senior who studies computer science, I can confidently say this class offers little in the way of quality instruction. Boaz is too smart to teach this class; his textbook is awful and his lectures are (somehow) worse because he has no idea what it means to learn theoretical computer science from scratch! The problem sets weren't 124 brutal but they were unnecessarily complicated and overall requires way too much in the way of extrapolation of concepts learned in the course. The biggest problem is that BOAZ CANNOT TEACH. He relies too heavily on understanding from the textbook, which would be fine THE TEXTBOOK WAS NOT UNREADABLE. IT'S SO AWFUL. The proofs are a waste of time (they're literally not in the same language that I know to be English), his definitions are extremely clunky and often unnecessarily so, and the images/examples are so esoteric so as to make me question his grasp of an undergraduate's reality. Boaz is a really nice guy and I hope he improves the class drastically. I even did well in this class and hate it.

Evaluate the course overall.: **3 (good)**

This course is very interesting at a high level. The issues of uncomputability and complexity mark the limitations of computation period, not the limitations of our current computers, so I think that it's an important class for all concentrators to take. There are also some interesting concepts from the real world such as cryptography and quantum computing (at the very end of the course). That being said, the problem sets and exams are long and hard, so be prepared to put in some work. Prof. Barak's lectures aren't particularly helpful either, but his text is quite good. Frankly, office hours are vital. Having experience with mathematical proofs isn't necessary, in my opinion, as they aren't very strict in this course and therefore this skill can be picked up quickly.

Evaluate the course overall.: **4 (very good)**

I wouldn't necessarily strongly recommend this class, but are you really going to take CS 221 if you're CS? In all seriousness, good class.

Evaluate the course overall.: **4 (very good)**

I actually thought that this was a pretty interesting class. It's not very much work, and you cover some really cool results, like the existence of uncomputable functions.

Evaluate the course overall.: **4 (very good)**

This class gets a lot of criticism it doesn't deserve. This class surprised me with how good (and not actually that hard) it was. If you have taken proof-based math at the level of Math 23A/B (or Math 101?), this class will not be very hard for you. The main reason people complain about this course being hard and painful is that the recommended prep by Harvard CS department is Math21A/B + CS20, which is just not enough mathematical preparation (although if this is you, know that you're in the same boat as a lot of people). Boaz's lectures aren't the most engaging, but the textbook is great, the P-Sets are good, and Boaz is really committed to making this course better.

Evaluate the course overall.: **4 (very good)**

You're probably taking this class because it's for the CS concentration. Although I had a rough time in this class and probably know that a lot of this stuff won't be applicable in the real world, this class actually grew on me a little bit and made me appreciate the theory behind CS a little bit.



Evaluate the course overall.: **5 (excellent)**

It's a requirement

Evaluate the course overall.: **1 (unsatisfactory)**

For those out there thinking about taking CS 121 as an elective, I would suggest finding another course. The subject matter is interesting. And coming from a math background, I did not find the class particularly challenging. That said, we only covered topics on a superficial level and the course lacked any semblance of organization.

Evaluate the course overall.: **5 (excellent)**

(from someone who did CS124 first, which helped, but only somewhat) It wasn't nearly as bad or painful as people hyped it up to be. The exams were very fair, and there were 20 points of extra credit on each one. Psets took about 7 hours to figure everything out, 3 hours to type up. Get a good partner and partner pair to collab with.

Evaluate the course overall.: **4 (very good)**

CS121 is overall an okay class. It was never the most exciting class and if it weren't required for CS students I probably wouldn't have taken it. However, Boaz is a good lecturer and assignments were clear in their expectations.

Evaluate the course overall.: **1 (unsatisfactory)**

dont take unless you need to.

Evaluate the course overall.: **4 (very good)**

CS121 is a great course. Boaz is a really caring professor and tries to give students information in a manageable way. Theoretical CS turns out to be really cool! In terms of advice: Definitely take the time to push through the readings, and report typos! We tried to do it this year, and you should pass on the favor :)

Evaluate the course overall.: **3 (good)**

Pretty decent course overall.

Evaluate the course overall.: **1 (unsatisfactory)**

This is by far my least favorite class I've taken at Harvard so far. Boaz is a good lecture and the class is pretty accessible with lots of office hours and help. But there were so many things wrong that I can't even remember them all. The biggest issue I had was with the strict "grading rubric" style of evaluation of this class. One of the things I loved about other math-heavy courses, like Stat 110 or CS 124, etc., is that the TAs for those classes were given a pretty good level of freedom in how they graded; as a result, since all of them were incredibly proficient w.r.t. the material, it would be clear to them when someone understood a problem vs. when they didn't and could grade appropriately and with discretion; as a result, students who wrote 1-2 paragraph proofs were generally the ones who got full credit since they had focused understanding of the material, while those who wrote pages and pages would do much worse both on the problem sets and on exams. In contrast, 121 has a grading rubric for every problem set and every exam, extremely fine-grained, such that many students that I knew in the class felt pressured to write multiple page proofs for simple questions just so that they didn't miss anything... and this WORKED, so they kept doing it. If you understand the material, you will still do well in the class but it's frustrating that the grading system encourages students not to try to focus their understanding and synthesize a proof to its critical components, which is nothing but harmful in the long run.

Evaluate the course overall.: **4 (very good)**

It's a dense class that requires a lot of work on the weekly psets and readings. It progresses at such a fast pace it barely feels like you have mastery over any of the material until the end. It'd be incredibly beneficial to have taken a proof-based class like CS20 or Math23/25 before this class, otherwise you'll have a rough few psets to start.

Evaluate the course overall.: **4 (very good)**

Pretty good introductory course for learning theoretical computer science, covering things like Turing machines, uncomputable functions, $P=NP$, etc. Note that there is no programming aside from a couple bonus assignments. Problem sets and exams are all proving mathematical statements, so it will help if you are familiar with proof-based math before taking it (though not necessary). The class is not easy, especially the reading (yes there is a ton of reading) but you get a lot of bonus points on the homework and the exams are fair.

Evaluate the course overall.: **3 (good)**

This class is really difficult and time consuming. I would only advise taking it if you need to for a concentration requirement or if you're really into proof-based math.

Evaluate the course overall.: **5 (excellent)**

Do your readings and go to office hours.

Evaluate the course overall.: **4 (very good)**

Overall, it's a very good course. Inconsistent at times, but the larger results are actually very interesting. Boaz is a very enthusiastic person, and very likable, but he occasionally forgets to mention when certain details are important to know (and when they are not).

Evaluate the course overall.: **5 (excellent)**

Incredible class. Just incredible. The content selection is of god-tier standard.

Evaluate the course overall.: **4 (very good)**

This class is a very manageable introduction to theoretical computer science. Prof. Barak really cares about student success and does a great job making the material accessible and understandable. As long as you keep up with the readings and the homework it's a very reasonable class.

Evaluate the course overall.: **3 (good)**

Well, dear cs concentrators, you have to take this class - you don't have a choice. The class is not great - most of you probably wouldn't enjoy the material that much but it is also not a terrible class. It is manageable and I would say it is not that hard (compared to many other classes in the department). Make sure you are comfortable with proofs before coming into class and then make sure you understand the psets. If you do this, you will probably be alright. Also, the class is kinda bad so don't have any big expectations - just get done with it.

Evaluate the course overall.: **4 (very good)**

CS 121 was a pretty good class. There are a lot of juicy ideas and you build up computation from NAND gates to Turing completeness. The class is still young and there are some kinks with learning materials, the exams, and second half content that are getting flattened out. Shoutout to Brian for being a great head TF.

Evaluate the course overall.: **4 (very good)**

Lectures aren't great, but in all this course is a good comprehensive overview of really interesting and essential topics in CS. The problem sets do a really great job of enforcing a standard of rigor / a new way of thinking and proving things that I found was really helpful when doing any other algorithms or theoretical CS problem.

Evaluate the course overall.: **4 (very good)**

I decided to toughen up and take this class without CS20 and zero background with computational theory or any proof knowledge. It was difficult at first but I kept pace with the readings and worked hard on the psets with a good pset partner so everything worked out and was enjoyable. Make sure you understand the pset answers because that is the most important material you will need to know for exams.

Evaluate the course overall.: **5 (excellent)**

CS121 is an incredible course and I would recommend this to anyone, especially those with a taste for quantitative/CS subjects. The work load is heavy, and this should be considered as one of the 'larger' courses in your semester's planning -- do not short yourself time in a schedule including courses like this.

Evaluate the course overall.: **3 (good)**

It's a CS requirement so brace yourself - having friends to work through concepts and psets together is extremely helpful. Also go to office hours as often as you can. Lectures are helpful, but they often don't do much more than confuse you.

Evaluate the course overall.: **1 (unsatisfactory)**

I love Boaz Barak D.

Evaluate the course overall.: **4 (very good)**

CS121 is the kind of class that you struggle with during the semester but look back on fondly. I personally found the material very interesting, and I thought it was a useful course for a CS or math concentrator to take. There are lots of office hours and resources available, and the Piazza forum is usually good for getting quick answers. You need to have the prerequisites (proof-based math at the level of 23abc or above is a must), and it would be nice if Boaz focused more on intuition than the dense notation, but I would still recommend taking this course.

Evaluate the course overall.: **2 (fair)**

dont take it if you dont have to take it! take it at MIT under Sipser!

Evaluate the course overall.: **5 (excellent)**

1. Get a partner. They don't have to know CS well, but must be someone you are happy with. 2. Take this if you are interested in the material. If the first couple lectures son't interest you, drop it - you won't stand it. 3. If you like it, THIS COURSE IS AWESOME <3

Evaluate the course overall.: **1 (unsatisfactory)**

Use sections and homework as an opportunity to learn. You don't need to know everything from lecture. Make sure you understand the format of the test and study accordingly. Many theorems won't ever be tested.

Evaluate the course overall.: **3 (good)**

the concepts in this class can seem removed from applicable computer science but it teaches good ways of thinking and problem solving skills

Evaluate the course overall.: **4 (very good)**

This class is hard. You will struggle with the pssets (but there's bonus points so it works out). Make sure that you prioritize this class in your semester. You will only get the most out of it if you can put the effort in. I would not recommend taking this concurrently with another CS class or even Stat110. You'll just detract from your learning in both classes and do poorly. Make sure that you have seen discrete math in some class (CS20 is great! and tailored for some of the topics in CS121). While I was hesitant about the NAND and NAND++ models, looking back they aren't terrible. We discuss Turing Machines for a bit, but the point is that the model of computation doesn't matter (too much). Overall I think this class is worth the time and a great way to think about CS theory.

Evaluate the course overall.: **1 (unsatisfactory)**

This is a candidate for the worst class I've ever taken at Harvard. If you don't have to take this class, avoid it like the plague, especially if you have no experience with proof-based math. This class is overly pedantic with its math jargon, and it's very math-forward. You don't write one line of code in this class. The textbook is full of typos, and doesn't explain the concepts you're supposed to learn, but instead just presents you with an inexplicable wall of math jargon and asks you to read it over and over again until you understand. It also doesn't help that you're supposed to do the readings BEFORE lecture and you're expected to already understand what's on the textbook. Absolute trash.

Evaluate the course overall.: **4 (very good)**

This was actually a pretty interesting class. It may not be super applicable in your life, but I think that it is important to have understood ideas of computability and efficient computability. Not much coding, and you get good at writing proofs. I liked the math component and found it empowering to be able to prove things were uncomputable or computable or efficiently computable. The psets take a while, but just go to office hours and ask questions! Always start early! Do the textbook readings, you definitely need the readings and the lecture to understand the ideas. Midterms and final were generally reasonable. Section is not helpful.

Evaluate the course overall.: **1 (unsatisfactory)**

I know you have to take this class, and I'm sorry. Try and find people to pick through the readings with, use ALL the office hours, search out tutors and peer fellows and every available resource to help with psets, and STUDY FOR THE TESTS because boy they nearly got me.

Evaluate the course overall.: **4 (very good)**

It's very interesting - the material is very foundational, in a good way.

Evaluate the course overall.: **3 (good)**

Make sure to do the pre-reading. I, personally, could not attend lecture as the professor spoke much too quickly so I could not understand the material. I watched the lecture videos at home at took the notes at my own pace. Find a great pset partner and try to start as early as possible. Also, some TF's are not good, so try not to get discouraged if some office hours/section are not as good, try others!

Evaluate the course overall.: **3 (good)**

If you have to take this as a CS requirement, then you just have to survive this course. If you don't have to take it I wouldn't recommend it. Psets are hard an time consuming and I don't feel like you learn a ton from them.

Evaluate the course overall.: **3 (good)**

This a great class in many respects, but certainly has its drawbacks. My biggest complaint is that the material is not treated rigorously. Because of that, I think you'll get out of this course what you put in to. If your goal is to skate by and check-off the box on your requirements list, you're not going to like the class. What's more, you'll probably walk away with a pretty poor understanding of the material. I encountered several people in the final weeks of class who still didn't understand the proof that the halting problem is uncomputable. The problem is they didn't really have a chance of getting it because they also didn't understand diagonalization (some of them thought they did, but also happened to think that you can diagonalize the natural numbers). Their failure to understand that likely didn't affect their grade, though, because all you're really expected to know is how to apply Rice's theorem.

Evaluate the course overall.: **4 (very good)**

This course is very much ok - somewhat frustrating though not too bad overall. The material in this course is very interesting! The organization of the course definitely leaves something to be desired, though. The textbook is full of errors and is super dense, not a good way to first encounter the material, even though in concept it is cool and a good resource. This class will require time, to be sure, but doesn't necessarily need to be the absolute center of your semester -- for example, you could take Stat110 alongside it and be fine. Also pro tip is really try to do the psets yourself because if you rely on others to carry the weight of the psets it will really take a toll later on.

Evaluate the course overall.: **4 (very good)**

This class is required for CS concentrators, so my response is directed mostly at people who actually have a choice whether they want to take it or not. :) (For reference, I took this sophomore year after taking quite a bit of pure math freshman year - I'm not a CS concentrator, so maybe they like this course better). I recommend this class with reservations to mathy people like me who want to take it. In particular, the material is generally super interesting (particularly Cook Levin Theorem, Rice's Theorem, Quantum Computing), and I think Boaz is actually a very good and engaging lecturer. However, I didn't end up getting as much as I wanted out of this class. The problem is that the course psets and exams are ****not**** geared towards the most interesting material: they often just ask you to make pretty trivial applications of theorems which you are never expected to really question. Worse is that Boaz literally forces you to read his book chapters before you come to lecture (and then quizzes you on the material), making lecture entirely useless, despite the fact Boaz is a good lecturer! As a result, I don't think I got much more out of this class than I would have just by reading the lecture notes. You do end up getting a pretty strong foundation in theoretical computer science, so if you do want a chill class, I would recommend it. However, if you're seriously interested in this stuff (and have experience with proofs and math), I would recommend skipping this and taking a higher level class.

Evaluate the course overall.: **3 (good)**

If you like math, you'll probably enjoy a lot of the content of CS121. Boaz really cares about the course and the students, and his lectures are pretty good. However, I wasn't a huge fan of the organization or focus of the content in the class, and the proofs in the lecture notes were pretty hard to read a lot of times. Still pretty cool content though

Evaluate the course overall.: **1 (unsatisfactory)**

This class has really interesting material, but it is not yet ready to be a class. I think that given a lot more time, this class would be great. Right now, however, the class is just too disorganized, and that just gets in the way.

Evaluate the course overall.: **4 (very good)**

In general, I think this course is very good overall. The textbook does contain annoying errors, but the topics covered are very interesting and the professor/staff is excellent. To those CS concentrators who are required to take this course but are uncomfortable with proof-based math, I encourage you to give this course your all. For the first couple of psets, it may be more difficult to do all the problems with fully rigorous proofs, but I would encourage you to start early and go to OH to get help and really work on proof-writing. I think the ability to discuss problems in a rigorous manner is an important skill that you will really get out of CS 121 if you see the class as an opportunity instead of a requirement.

Evaluate the course overall.: **2 (fair)**

man this class kinda sucked lol

Evaluate the course overall.: **1 (unsatisfactory)**

Only take this class if you have to in order to graduate.

Evaluate the course overall.: **1 (unsatisfactory)**

Unless this course improves drastically next year, don't take this. In fact, if you're planning on majoring in CS or getting an introduction to TCS (as the course purports to do), run away in the other direction now. The pedagogy, at least from the professor, is terrible (see the satire v article on this class, it's surprisingly accurate). The textbook is still in progress but is the only course reference, the proofs are full of overcomplicated notation and little intuition, and lecture is essentially a repeat of the textbook material in 75 minutes. The problem sets are too easy and there are so many bonus points given on those that it's entirely possible to have a full problem set average while not understanding a fair amount of the material (and likewise for the tests). A lot of problems merely involve wading through the mess of notation present in textbook definitions, and many can be solved by some tool that's clearly too powerful because it reduces the problem to a triviality - bad writing on the teaching staff's part. The tests themselves have an opaque grading scheme and fluctuate in difficulty, to the point that I'm not sure if anyone's average is a good reflection of their understanding of the material unless their average is extremely good or bad. The professor ostensibly pays attention to student feedback, but he responds in absurd ways (for example, after the midterm people said they were having trouble understanding the proofs, and in response he told people to tune out in lecture while he was going through a proof). The attitude of the class to "rigor" is insulting - although the class claims to pay attention to rigor, often the textbook's "rigorous proof" is a mess of too-formal notation and meaningless symbol-pushing that ignores the fact that one can be "rigorous" without resorting to writing an equation every line; thus problem set solutions are often graded on a scale that equates length with rigor rather than actual proof (it turns out that if you write a page or more of BS that sounds vaguely correct, you can get full credit on a problem, probably because the TFs saw some words they were looking for and didn't want to read through the entire thing). In fact, after the first few weeks I don't think I did more than read a "proof idea" and skim over the actual proof for most theorems unless I found the proof particularly egregious. The structure of the course also needs work: taking the course seemed like going through a grab bag of topics in computability theory, and it seemed unlikely that there was some unified purpose to the class. It didn't help that basing things on NAND meant that if we had questions, the internet could not help us. The TFs vary a lot in competence, unfortunately. Some know what's going on and are doing their best to improve the student experience - they're honestly heroes. They're capable of clearing up points you can't understand (for various reasons: a typo in the textbook, bad writing in the textbook, no explanation given in the textbook) very rapidly and actually making you think the material is worth learning. Others seem to not know much TCS, can't evaluate proofs well, and are generally unhelpful. The amount of inanity involved in the grading of the course is absurd. There have been numerous instances of TFs taking off points for some component of the proof they claimed was missing but was actually written, of refusing to accept solutions that differ from the standard examples given in section and class, and of giving credit to solutions that would fall apart under any form of scrutiny greater than a superficial once-over. On a more positive note, Boaz is a great guy when he's not teaching - very smart, quirky, and personable. He is, however, so smart that it's difficult for him to impart his understanding of the material onto a class that has a wide range of experience with mathematical concepts, proofs, and computer science.

Evaluate the course overall.: **1 (unsatisfactory)**

Don't take this class without really thinking about it. Unless you are a CS concentrator and you have to.

Evaluate the course overall.: **3 (good)**

The course requires a good amount of mathematical maturity above all else, much more than any CS background. That being said, the material is deeply interesting.

Evaluate the course overall.: **5 (excellent)**

This is one of the most fun CS classes I have taken at Harvard. There are very few classes I have taken that changed the way I see the world and this is definitely one of them. The material is incredibly interesting. Boaz cares so much about his students and the course. There are tons of bonus points on psets/midterms. The psets are so well thought out. If you have taken a 100 level math class or math 23,25, etc then this class won't be challenging for you. If you've only taken math 21 then you might slightly struggle.



Evaluate the course overall.: **2 (fair)**

Do not take this class unless it is required. Textbook and lectures are unnecessarily complicated and difficult to follow. Material is not very useful, and some is covered in CS124 as well. This course is not as hard as CS124, but the workload is still significant.

Evaluate the course overall.: **4 (very good)**

Don't fall behind in lectures and reading, you will suffer. The textbook wasn't finished yet, so missing sections and typos were plenty.

Evaluate the course overall.: **2 (fair)**

It seems as if Professor Barak has made the course a little easier and more forgiving than previous years, so hopefully that grants a little peace of mind! The course is still pretty tough though and requires a lot of work.

What did you learn? How did this course change you?

Course

Evaluate the course overall.: **3 (good)**

I learned about a cool new perspective of CS through CS theory, which I was entirely unfamiliar with before.

Evaluate the course overall.: **1 (unsatisfactory)**

Proofs

Evaluate the course overall.: **4 (very good)**

It hasn't changed me too much.

Evaluate the course overall.: **3 (good)**

It definitely got me to think more like a theoretical computer scientist

Evaluate the course overall.: **4 (very good)**

I learned a lot more about the science part of computer science, rather than just the programming aspect.

Evaluate the course overall.: **4 (very good)**

I was introduced to theoretical CS and I first began to consider theoretical cs as a future (if unlikely) pursuit. It was a good first cs class at harvard for a math student.

Evaluate the course overall.: **3 (good)**

I learned I REALLY don't like theory

Evaluate the course overall.: **3 (good)**

Theoretical Computer Science can be interesting, but probably not an area of CS I want to focus on.

Evaluate the course overall.: **2 (fair)**

We covered a number of interesting topics!

Evaluate the course overall.: **4 (very good)**

I grew in my understanding of computation and my problem solving skills.



Evaluate the course overall.: **3 (good)**

The course definitely was true to its name in that it introduced me to the theoretical aspects of computer science in a way I hadn't really encountered before, which I think was inherently valuable. It also gave me a little more insight into what my future study as a CS concentrator at Harvard may look like, which is also a good experience to have.

Evaluate the course overall.: **3 (good)**

Some functions are uncomputable.

Evaluate the course overall.: **5 (excellent)**

I learned how and why some functions are uncomputable, what it means to be efficiently computable and what the P=NP debate means, how to do proofs by reductions, and a lot more. As a systems person, I'm really glad I took this course -- theory is super interesting and this course was a good introduction.

Evaluate the course overall.: **4 (very good)**

I've deepened my dislike of math

Evaluate the course overall.: **4 (very good)**

Opened my eyes to how much else there is in computer science. Now there are so many more classes I want to take, but so little time!

Evaluate the course overall.: **3 (good)**

I thought this was a good introduction into the core ideas of theoretical computer science - computability, reductions, complexity.

Evaluate the course overall.: **2 (fair)**

I have acquired a decent but superficial understanding of a wide variety of CS concepts.

Evaluate the course overall.: **3 (good)**

I learned about important topics in theoretical computer science.

Evaluate the course overall.: **4 (very good)**

I learned about how to write proofs and major concepts of theoretical CS. I never realized how interesting theoretical CS was and am glad I was forced to take it and expand my interests.

Evaluate the course overall.: **2 (fair)**

I learned how to think about CS more theoretically. I dont think the course changed me.

Evaluate the course overall.: **4 (very good)**

I learned that theoretical computer science is really fun and interesting, and I love the content that we learned. I came into this class without a very strong math background and struggled at the beginning, but over the course of the semester I learned about what makes a proof rigorous and how to contextualize all of the information we had learned over the course of the semester. One of my favorite moments of this whole semester was when I FINALLY understood and could do reductions, and I think this class was great for both my computer science education and my personal growth, as it taught me that even though these topics were extremely challenging, I could still understand them well if I worked really hard and tried my best.

Evaluate the course overall.: **1 (unsatisfactory)**

p=np?

Evaluate the course overall.: **4 (very good)**

I found new ways to think about problem solving the the limits of computation.

Evaluate the course overall.: **1 (unsatisfactory)**

I learned some really interesting things about the theory of computation and sparked an interest in several areas, including cryptography.

Evaluate the course overall.: **5 (excellent)**

Love and appreciation of TOC. I realise that CS is not bound by humans's discovery of a macbook pro: computation is something very fundamental and intrinsic to the fabric of reality. I love math too, so I found this course making good use of that to reach stunning conclusions. I also learned that Boaz is a smart Jewish guy.

Evaluate the course overall.: **3 (good)**

Theoretical isn't as bad as I thought it would be

Evaluate the course overall.: **4 (very good)**

I learned that theory was a very interesting field in CS that I'd really like to study more! Also realized that slacking off in this class was bad, and to not make silly mistakes in future tests.

Evaluate the course overall.: **4 (very good)**

Learned about the limits and frontiers of CS.

Evaluate the course overall.: **1 (unsatisfactory)**

I got a weird smattering of information that I don't feel is truly applicable. I do think that a couple things really stood out and gave me a better understanding, but overall not worth it.



Evaluate the course overall.: **3 (good)**

I learned that I learn best in a 1-1 environment, so I should keep trying to find tutors where possible.

Evaluate the course overall.: **3 (good)**

I can write slightly better proofs

Evaluate the course overall.: **1 (unsatisfactory)**

I learned a new way to think about certain problems in CS.

Evaluate the course overall.: **5 (excellent)**

Great background in proofs and theory.

Evaluate the course overall.: **4 (very good)**

I thoroughly enjoyed this topic and learned a lot, and I think I would like to try a summer of research in related topics.

Evaluate the course overall.: **2 (fair)**

I learned never to touch theoretical computer science again.

Evaluate the course overall.: **4 (very good)**

This course definitely changed the way I think in terms of building thoughts and opinions from axioms

Evaluate the course overall.: **4 (very good)**

I learned new ideas about TCS.

Evaluate the course overall.: **4 (very good)**

Theoretical CS is more interesting than I thought.

Evaluate the course overall.: **4 (very good)**

I learned a lot about theoretical cs (AND math)! Because of this course, I've decided to get a secondary in cs.

Evaluate the course overall.: **3 (good)**

I learned so, so much. I not only learned about theoretical computer science, but also about how to study and improve as a student. This class had really interesting material, and I feel that it gave me a new perspective on life and the things around me. I hated this class for a while, but now that I'm finished I love it. It's one of the best I've taken at Harvard because it taught me so much and really made me love CS.



Evaluate the course overall.: **4 (very good)**

I learned a lot about different complexity classes and models of computation and how they relate to each other.

Evaluate the course overall.: **1 (unsatisfactory)**

I learned that I don't want to do computer science anymore

Evaluate the course overall.: **1 (unsatisfactory)**

I do indeed have a better understanding of the foundations of theoretical computer science...but not without tedium and frustration.

Evaluate the course overall.: **5 (excellent)**

It's difficult to summarize everything that I've learned and taken away from this course. Suffice it to say, I learned a ton about all sorts of topics in computation; many were topics that I'd heard of and knew were important but never actually understood. I'm also leaving with a newfound appreciation of, and interest in, theoretical computer science despite never being a "theory person". I've learned and practiced a new way of thinking, and I'm thankful for this course.

Evaluate the course overall.: **5 (excellent)**

P=NP? Theoretical computer science is cool.

Evaluate the course overall.: **3 (good)**

I've wanted to learn about a lot of the stuff taught in this course for a while (Incompleteness, NP-completeness, Quantum Computing), and this course has satisfied that itch.

Evaluate the course overall.: **3 (good)**

I have a stronger understanding of various topics in computer science theory that I had heard thrown around but never really understood completely, like universality, Godel's Incompleteness, Turing machines, quantum computing, cryptography, P vs NP, etc.

Evaluate the course overall.: **2 (fair)**

Honestly, it just made me feel bad about myself as a student and really steered me away from any sort of proof-based math or theory in cs.

Evaluate the course overall.: **5 (excellent)**

I learned a lot of intriguing new concepts relating to theoretical computer science. I entered the course having learned a bit about things like runtime and complexity classes, but, by the end of the course, I felt like I really understood those things in a way I hadn't before.

Evaluate the course overall.: **3 (good)**

I feel like I learned a side of Computer Science that I had no idea about before taking this course. It was fascinating to learn about problems that computer scientists have to think about every day.

Evaluate the course overall.: **4 (very good)**

I think it definitely improved my creative thinking.

Evaluate the course overall.: **4 (very good)**

I learned (along with my harrowing experience in CS 61) that I am most naturally a CS theorist and much prefer proving theorems to writing code.

Evaluate the course overall.: **4 (very good)**

Gave me a new perspective on CS that I didn't know before.

Evaluate the course overall.: **1 (unsatisfactory)**

Some general computer science knowledge. Basic ideas like computability, function classes, reductions, etc.

Evaluate the course overall.: **4 (very good)**

I am interested in taking more theoretical computer science courses.

Evaluate the course overall.: **4 (very good)**

I learned a lot about theoretical computer science and gained new ways to think about CS problems.

Evaluate the course overall.: **3 (good)**

I learned more about what is computable and what isn't

Evaluate the course overall.: **1 (unsatisfactory)**

Post-124, this class was both uninteresting and (at least, it felt) not applicable to any real life problems.

Evaluate the course overall.: **3 (good)**

I learned the basic theory of computation. I think this makes more attuned to the difficulties associated with solving certain problem.

Evaluate the course overall.: **4 (very good)**

It went through the limits and powers of computers, abstracted away from any particular programming language.



Evaluate the course overall.: **4 (very good)**

This course taught me a lot about the theory of computation and was definitely a really interesting way to think about the ideas underlying the programming I do in other classes.

Evaluate the course overall.: **3 (good)**

I feel much more comfortable with math proofs and different kinds of notation right now. It was also interesting to talk about cutting-edge research and what are the hot topics in theoretical cs. Other than that, the course didn't change me that much - if something, now I know that I don't want to do theoretical cs.

Evaluate the course overall.: **4 (very good)**

Formalized ideas of encoding and computation. Learning about the halting problem was a bit more fruitful than I thought

Evaluate the course overall.: **2 (fair)**

Theoretical computer science is not for me. But it was still a very interesting class overall

Evaluate the course overall.: **4 (very good)**

I learned interesting computational theory

Evaluate the course overall.: **5 (excellent)**

A pretty comprehensive introduction to theoretical computer science topics and proofs (specifically in the realm of computation) and a pretty solid understanding of algorithmic complexity and run-time analysis. Also, a great intro. to lots of hot-button CS topics e.g. crypto/quantum computing at the end of the course.

Evaluate the course overall.: **3 (good)**

I don't know. It made me cry and have a much more miserable semester than you, next

Evaluate the course overall.: **1 (unsatisfactory)**

I learned never to take a computer science class ever again in my life.

Evaluate the course overall.: **1 (unsatisfactory)**

I learned about time complexity classes and computability, but I'm still struggling to figure out its relevance to my life.

Evaluate the course overall.: **3 (good)**

learned good problem solving skills and ways of thinking



Evaluate the course overall.: **4 (very good)**

I learned theory

Evaluate the course overall.: **4 (very good)**

This course changed how I approached CS. Before I would just think of algorithms and solutions to problems. However, this class made me think about what computation means and how some problems are just harder than others.

Evaluate the course overall.: **1 (unsatisfactory)**

Uncomputable functions, Turing-completeness, P vs NP problem

Evaluate the course overall.: **1 (unsatisfactory)**

It made me realize I needed a therapist

Evaluate the course overall.: **4 (very good)**

I understand the underpinnings of computation much better now.

Evaluate the course overall.: **3 (good)**

Understand basic principles of theoretical CS and am better at proofs.

Evaluate the course overall.: **2 (fair)**

I learned to keep working hard even when things don't come easily. Theoretical CS does not come naturally to me so I tried to buckle down and improve.

Evaluate the course overall.: **3 (good)**

This course changed the way a think about computation and just generally expanded my mind. I appreciated the proof/analytic techniques used to prove results in computability/uncomputability and P vs NP (reductions and such)

Evaluate the course overall.: **1 (unsatisfactory)**

I did learn a good amount of computer science concepts.

Evaluate the course overall.: **1 (unsatisfactory)**

I don't know.



Evaluate the course overall.: **1 (unsatisfactory)**

I feel like I learned very random and useless facts about how computers theoretically work. I will try to never cross these topics in my career.

Evaluate the course overall.: **3 (good)**

I learned a great deal more about the theoretical underpinnings of computer science. I learned some about my ability to do math.

Evaluate the course overall.: **2 (fair)**

I learned which functions were uncomputable, the notion of a Universal circuit, and improved my Proof skills

Please comment on this person's teaching.

Barak, Boaz

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Professor Barak is unquestionably brilliant, however I had a lot of problems with his pedagogy. Often times, he would assume that no one having questions in a 100+ person lecture meant that everyone understood the material completely (this was, I wager, never the case). As someone who felt insecure about their math background and was often nervous to speak up and ask a question, this was really discouraging to me. Additionally, throughout the course, Professor Barak gave an unclear picture of which parts of the content were more significant than others. I was never sure which proof details were important, and at one point in the semester Professor Barak was telling us to tune out at certain points if we got confused and to tune back in at other points, which was just more confusing. Nevertheless, Professor Barak is well-intentioned and it was evident throughout that he was doing whatever he could to make the class better.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

So bad. so confusing. his lectures make NO sense.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Sometimes it would be difficult to understand what he was saying in lecture, but other than that I found Boaz to be a funny and somewhat quirky person.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **2 (fair)**

I do not understand what he says 60% of the time, because 1) he moves way too fast for a majority of the class and assumes knowledge in his students that is not really there, 2) his accent, which I do not blame him for of course

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

I think Boaz knows the materials very well and is a very good scholar/person. He cares genuinely about teaching and welcomes student feedback about his style.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

The lectures could be confusing at times, but Professor Barak usually did a good job trying to keep them interesting while moving quickly through the material.



Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz is an excellent teacher and a luminary in his field. However, I think he sometimes struggles to find exactly where students sticking points are when they don't understand the material.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz was very active and patient with questions in review session and on Piazza, and made it clear we could ask questions anytime. His lectures were effective and often very good, although most of the learning was done by reading his (sometimes-intractable) textbook.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz really tries hard to be an engaging professor. 75% of the time the lectures are engaging and well-taught, but the lectures on proving Cook-Levin Thm and others in the second half of class were extremely dry, and not helpful to students trying to understand the material.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz is a very intelligent and friendly person. Unfortunately, his intelligence doesn't always translate to being able to convey his knowledge effectively. His textbook is often difficult to parse due to unnecessarily complicated notation, making it hard to really grasp the material, and his lectures follow a similar pattern. That being said, he is definitely one of the smartest professors I've had, and I really like him as a person.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

His lectures are not engaging because he speaks in a very monotone voice.

Evaluate the course overall.: **2 (fair)**

Evaluate your Instructor overall.: **3 (good)**

I found lectures to be too complicated/move to fast, even having done the readings

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is really nice and it's obvious that he cares a lot about the class. His lectures were often pretty confusing.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

very funny! enjoyed the lectures



Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Lectures, textbook and psets can be hard to understand at times.

Evaluate the course overall.: **2 (fair)**

Evaluate your Instructor overall.: **2 (fair)**

Boaz expects students to understand the material in the readings too well, especially because they are often very dense, and it isn't obvious what the key takeaways of the material should be. Lecture should be slower paced and more introductory - the textbook should be there as a secondary, not primary, resource.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Professor Barak was excited about the material, but sometimes did not effectively communicate the important concepts with ease which made it difficult to grasp and continue building upon. It would be helpful to know what concepts to look out for so that you can understand them and build upon them.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz cares a lot about his students and teaching, and it shows in the way he comes up with his own pedagogical techniques (ie NAND rather than Turing Machines / FSMs, even though imo it'd be better to actually learn what those other things are too). However, his lectures are boring and I didn't retain anything from them. I still appreciate him though!

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **2 (fair)**

I found lectures to be confusing and difficult to follow, and I think others felt this way as well. Sometimes concepts were rushed through in lecture, and we moved on before I fully understood (which may have also been my fault, not sure).

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz has put in a lot of effort to make this class good and I really appreciate that. He seems to be very thoughtful about assignments, answering questions from people during lecture, rewarding participation/trying hard and is very fair. His lectures are well designed and lecture notes, although they are dense, are very important and helpful on the second, third, or fourth reread. He is extremely accessible outside of class via email and on Piazza, and genuinely seems to care about his students. I got an email from him after my first midterm about how to improve for the next midterm, and it really, really helped me and showed that he is genuinely invested in seeing his students improve. Thanks for a great semester Boaz!



Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

The lectures are run such that if you miss any little detail of whats being said, you won't know whats going on for the next 30 minutes. In general, I got the sense that Boaz doesn't realize how to describe something intuitively and doesn't know what he should even be trying to describe well.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

He was very good at making lecture engaging and informative.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

Professor Barak is a very intelligent and talented researcher, but I found this course to be run very poorly. The course text (written by Professor Barak) was one of the weakest parts of the course. I highly recommend a major overhaul of the course before it is taught again.

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is brilliant. I would like for him to make lectures more engaging (not that he doesn't try), but else everything is great

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Very enthusiastic for theory, and has definitely put in a tremendous amount of effort into the textbook / course materials. Often though concepts that easily make sense to him get lost in translation, and sometimes it can be hard to understand, even after asking questions.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

I love Boaz's enthusiasm for this course and attempts to improve it every year. I didn't find the lectures too engaging to be honest though. Boaz clearly knows his stuff but sometimes I think it was a little dry and hard to maintain concentration. I loved CS121 and CS61 equally but if CS121 had lecture's like Eddie's I think it would have been a lot more fun!

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **2 (fair)**

Although I appreciated Boaz's enthusiasm for the subject matter (especially cryptography) and teaching, I really could not follow him much at all during lecture which hurt my own enthusiasm for the class.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **5 (excellent)**

Lectures are engaging and interesting. Boaz teaches well.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

If you get a tutor, you will realize that the way topics are presented are often obscure and unintuitive for an introductory course.

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

Lectures sometimes dry, but the content and textbook are awesome.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Honestly I probably am not qualified to judge because I realized I learned the course more time-efficiently by figuring everything out while solving the pssets, so I only listened to the first few lectures intently.

Evaluate the course overall.: **2 (fair)**

Evaluate your Instructor overall.: **4 (very good)**

Professor Barak is a genuinely pleasant and extremely knowledgeable individual tasked with teaching some of the most grueling content in the undergraduate computer science curriculum. I personally struggled a bit with the transitions between introducing a topic and delving into the details of it; there is often little build-up between the two, and ends up leaving only a surface-level knowledge of some important concepts. I suspect this will be resolved as the course itself matures, however, so I can't fault Professor Barak for that.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: no answer

Boaz's textbook is amazing. His lectures are probably amazing too but they were so similar to the book it felt a bit repetitive sometimes.

Evaluate the course overall.: **2 (fair)**

Evaluate your Instructor overall.: **2 (fair)**

Literally did not understand anything that was ever said in lecture unless he was telling jokes or being witty

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is a fantastic professor. Loves the material, cares about his students. The class is frustrating at times, but not because he isn't fully invested. If anything, he takes on a lot of things to try to teach which makes keeping up sometimes onerous.



Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **3 (good)**

While Boaz is certainly knowledgeable and enthusiastic, he's not the best lecturer. I found his lectures often hard to follow (even when the concepts seemed clear in the textbook). This was less of an issue for me since I have a lot of familiarity with proofs, but in general Boaz wasn't good at explaining proofs (even simple concepts like proof by contradiction or diagonalization arguments) which made relatively easy concepts seem a lot harder than they were.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Boaz is very enthusiastic about the course topics and willing to answer questions.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz is a very thoughtful and diligent professor, as it seems he is always answering people's questions on Piazza. He could improve his lectures and lecture notes by being clearer about the main concepts and parts of proofs, and not being as bogged down by technical details (most of which become more or less obvious once the main ideas are well understood).

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **4 (very good)**

You can tell Boaz really likes the subject and is passionate about it. He lectures well and answers any questions you may have well.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

He is just objectively not a good teacher. There are not a whole lot of good reasons for so many students to have struggled in this class. It was not well taught by prof. Barak

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **3 (good)**

Professor Boaz is clearly knowledgeable about the course material, but his explanations are at times dense and unclear. He is very active on Piazza, which is helpful.

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is a great professor. But I would have appreciated less time spent on "doing exercises with a neighbor" and more time spent on going over difficult parts of the readings. Overall, though, Boaz is amazing.



Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

Professor Barak really cares about the students and their learning. He's easily accessible and very frequently responding to all sorts of conceptual questions on Piazza. In class, he sets aside plenty of time for questions and he is always enthusiastic and engaging.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

I think Boaz is a great guy and really cares about the material. I see evidence that he has worked to address issues brought up in last year's Q guide reviews and even from the mid-semester feedback form. That being said, I think his lectures and textbook still have room for improvement. In terms of lectures, I liked the little concrete examples/characters he creates (for example, when explaining reductions and interactive proofs. Those were helpful for my understanding. The tables of red and green squares for randomized algorithms were also okay, but could have been explained a little more. However, some proofs completely lost me, such as the proof of Sipser-Gacs. In general, the concepts that are presented with visuals and "intuitive" explanations are much easier to grasp in lecture than the ones presented with a lot of mathematical notation. It would be helpful to see more of this in the textbook. Also, sometimes we took too many suggestions/answers from the audience. I get that we want an interactive structure, but if all the answers are along the same lines, then it's time to move on to the next question or concept.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

You need to offer office hours!! A lot of us want to have a chance just to talk with you! Please offer office hours!

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

His lectures helped me understand the material better after having done the reading. He understood the students' questions and explained the material from the textbook in a very clear manner.

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

While the first CS 121 lectures could be made more interesting, I have never met a professor who made me love the subject itself as much as Prof. Barak.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Prof. Barak's text was quite good (though with some typos), but his lectures could be a bit difficult to follow. I would have liked to see more time spent on the key ideas from the reading, which may not have been so straightforward to everyone, and more examples of types of proofs in class (rather than just saying that everyone should be able to do it trivially).

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

He clearly is passionate about the subject.



Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz obviously cares a ton about this class. He is very on top of piazza, and writes good P-Sets and exam review materials. His lectures aren't very engaging, but his written textbook is quite good, and the support structures of this class is well-designed to make it minimally painful.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

His lectures were basically a regurgitation of stuff from the textbook, which he also wrote. Would have been nice if he could explain the concepts more clearly / more simplistic manner and build his way up to be able to parse the terms in the book.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **2 (fair)**

Boaz is a great teacher in a lot of ways. He is clearly an expert on the material, and he genuinely cares about his students. I think he could work a little bit on organization in general. The textbook needs improvement. His graduate textbook on computational complexity is much nicer.

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **5 (excellent)**

Excellent work.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is a great lecturer and even though I personally didn't find the material very interesting, I had to admire how passionate Boaz was about the material.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is a really nice and caring teacher. He was very excited about the material, which was fun, but at times lectures felt a bit fast, and then he would slow down at the end of class. I appreciated how he really made an effort to make the course manageable (reading guidelines, chapter summaries, etc.) and was so encouraging. He was also super active on Piazza, which was great!

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Boaz is a pretty cool guy.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **3 (good)**

Good lecturer!



Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz is very enthusiastic and very likable. I think it would have been better if he more carefully labelled materials as NEED TO KNOW vs. INTERESTING MATH BUT CUMBERSOME AND WE DON'T REALLY CARE AT THE MOMENT.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Professor Barak really cares about student success and it really shows in his teaching.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz seems very interested in what he teaches which is great. He really tries to make the course better which I think everyone in the class appreciated but he still has a lot of work to do to get the class to a better place.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Pretty good lecturer. Sometimes overestimates people's ability to understand proofs

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **2 (fair)**

During lectures he struggles to communicate important information and explain topics in an intuitive manner. As a result, many students struggled to grasp the material that was inconveniently left unexplained besides the convoluted textbook definitions and proofs.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

I really love Boaz - he's an incredible person and cares a lot about his teaching - but I wish that he would more regularly and actively solicit feedback from students on his lecturing style or the ways in which the material is presented. There were several times when I received a short explanation from a TF or a student that I personally found way more intuitive / understandable than the way a concept was present in Boaz's book or in lecture. Learning from them I think would be a great first step in making the course more enjoyable.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is realistic with his expectations of what students should know which is really good because it might initially seem there is way too much we would need to know for this course. He was also a very good lecturer and a fun guy. A couple times we would go over proofs/material which we didn't really need to know but overall he did a stellar job

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Professor Barak is very enthusiastic and clearly cares a lot about the material and his students. His execution, however, is quite poor. His lectures are extremely difficult to parse through, and are not effective in relaying the course material. Also, he holds at most 10-20 minutes of office hours per week, and I wish he had at least been more available. He was very active on Piazza but that's not the best medium for many people.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **5 (excellent)**

I would estimate the teachings of Barak are somewhere in between those of the great sage Mengzi and Confucius.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

Boaz gives generally good lectures, although I feel he focuses too much on notation and definitional intricacies instead of trying to instill intuition (which might be more useful or memorable).

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

Boaz's lectures are extremely hard to follow, and the lecture readings are often not written in plain English that can be easily understood. That left very few avenues through which to learn in this class.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

His diction is colorful

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz's lectures are full of fun. He likes showing off various code that demonstrates NAND and other course concepts. The lectures do a great job building off of the textbook.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **4 (very good)**

He just goes really fast but he is nice.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **1 (unsatisfactory)**

Boaz does not teach examples in the way they are expected on the pset, and some long form examples (or near-pset examples) would help way more. Also, CHILL OUT on how hard the psets are in everything you teach.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Would suggest that he slows down his lectures, I was not able to attend because it was so fast, I watched the lecture videos from home. As I'm sure he's aware, there are multiple errors in his textbook as well as some updates needed to diagrams and additional problems. I would highly recommend that he fixes this before the next class. Lastly, I would recommend a more concerted effort towards diversity in next year's class of teaching fellows. As a required course for CS, many of my friends chose to drop the major because they felt unsupported by CS TFs. If the concern is a lack of diversity in terms of students who reach the grade cutoff to be a TF, that is a big concern that requires stronger support overall that starts with the professor.

Evaluate the course overall.: **3 (good)**

Evaluate your Instructor overall.: **3 (good)**

Very good in person, but lacking in lecture.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **2 (fair)**

.#####, #####, ##### ##### ##### ##### ##### ##### ##### ##### #####
Now English for some suggestions. Boaz's enthusiasm for the course material was infectious! However, sometimes he didn't seem to be on the same wavelength about what the student experience in the course was. This theoretical material is very prone to causing one to have what is called the Curse of Knowledge. What this means is that once one understands the CS121 material, in retrospect it seems totally obvious and one is cursed with thinking that, because it is simple from the 'other side' of the learning process, that the process of learning it is simple as well. This is totally not the case - in fact, approaching the material first through the textbook and from high level lectures were not always effective, and many times obfuscated perfectly simple concepts. What this course needs is, in addition to every seemingly complex math theorem, a bare bones, 8th-grade level english explanation beside it, and in lecture, a constant reiteration of basic facts to get everyone on the same level of basic understanding. Without this, people just tune out and get totally lost until they have to start the pset. All this aside, great job teaching and continue sharing your enthusiasm for CS with the world!

Evaluate the course overall.: **5 (excellent)**

Evaluate your Instructor overall.: **4 (very good)**

Sometimes a bit unclear or moves through topics too fast but always receptive to questions in-class and on Piazza.

Evaluate the course overall.: **1 (unsatisfactory)**

Evaluate your Instructor overall.: **2 (fair)**

The class is just too disorganized at the moment. Boaz gets hard to follow and goes way to far into some details that don't really need to be gotten in to. I think with more time, he can get the lectures polished, and then they'd be great.

Evaluate the course overall.: **4 (very good)**

Evaluate your Instructor overall.: **5 (excellent)**

Boaz is a great professor. His lectures are engaging and interesting.

Evaluate the course overall.: **1 (unsatisfactory)**
Evaluate your Instructor overall.: **1 (unsatisfactory)**
Seems like a really nice guy, just can't teach.

Evaluate the course overall.: **1 (unsatisfactory)**
Evaluate your Instructor overall.: **1 (unsatisfactory)**

Professor Barak is intelligent and an excellent researcher, but a terrible teacher (one of the few people I've encountered at Harvard that fits this stereotype thus far). I think he's just really bad at writing proofs in such a way that students just beginning to encounter this type of thinking will understand him (a friend of mine phrased it like "to him, both ways of proving this theorem are equally understandable and easy, while to everyone else one way is much harder to understand than the other"), and equally bad at presenting them in lecture. He seems very open to feedback and is working hard to improve the class, but I worry sometimes that his good intentions might not have the best results.

Evaluate the course overall.: **1 (unsatisfactory)**
Evaluate your Instructor overall.: **1 (unsatisfactory)**

Professor Barak was very often confusing and hard to understand. Even when he tried to slow down and explain things that most students weren't understanding, he couldn't do so effectively. On the other hand, he is extremely kind and even more intelligent.

Evaluate the course overall.: **5 (excellent)**
Evaluate your Instructor overall.: **5 (excellent)**

Incredible professor. Genuinely cares about his course and students.

Evaluate the course overall.: **2 (fair)**
Evaluate your Instructor overall.: **3 (good)**

Lectures and textbook are overly complicated (heavy reliance on mathematical notation when an explanation in words would be more intuitive).

Evaluate the course overall.: **3 (good)**
Evaluate your Instructor overall.: **1 (unsatisfactory)**

Very difficult to understand what he says. He constantly pauses and his speak is littered with umms and ahhs to a point where the flow of information from him to you is incomprehensible.



Which prior courses or self-study helped you the most in preparing for CS 121? Anything you didn't do but wish you have done to prepare?

Course

Evaluate the course overall.: **3 (good)**

Taking Math 101 was really helpful because I had 0 proof background in high school and Math 101 gave me a solid foundation for understanding the format of proofs. Taking Stat 110 concurrently also helped with the probabilistic computation sections.

Evaluate the course overall.: **1 (unsatisfactory)**

N/A

Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **4 (very good)**

CS20 was helpful

Evaluate the course overall.: **3 (good)**

Math 101

Evaluate the course overall.: **4 (very good)**

Reading Sipser alongside would be a helpful complement for general knowledge purposes. I don't mean ahead of time though.

Evaluate the course overall.: **4 (very good)**

I don't think anything I learned in prior classes was particularly useful.

Evaluate the course overall.: **5 (excellent)**

CS124 was extremely helpful, although I think CS121 would have been more useful had I taken it before 124.

Evaluate the course overall.: **4 (very good)**

It probably would have been good to have taken a CS class prior to this class (other than having taken APCS a long time ago), but I made it through with my math background and a lot of work. I don't think Stat110 is necessary though, as many people wrote in response to this question last year.

Evaluate the course overall.: **3 (good)**

CS20 and Stat 110 (taken concurrently) were helpful. I wish I read more mathematical proofs.

Evaluate the course overall.: **2 (fair)**

CS20

Evaluate the course overall.: **4 (very good)**

Homework 0 and the mathematical background chapter of the textbook

Evaluate the course overall.: **4 (very good)**

I'm a Math major so I've taken 12 math classes by this point, as well as CS124 and CS181. Prior exposure to mathematical notation and struggling with difficult concepts is definitely most important. I'm also a senior which helps. The people I saw struggling the most were those who hadn't seen mathematical notation before and hadn't taken CS20. Would highly recommend people without mathematical background to take it (and encourage this in advising to freshmen to take in addition to CS51) because many don't realize it until sophomore year, when it's too late.

Evaluate the course overall.: **3 (good)**

Math 23a and 23c were extremely helpful, if for no other reason than their proof-based nature.

Evaluate the course overall.: **5 (excellent)**

Cs124

Evaluate the course overall.: **3 (good)**

I don't really think I needed any prerequisite for this course, maybe CS50's genera; CS knowledge was helpful.

Evaluate the course overall.: **2 (fair)**

CS124, CS20. CS20 was helpful for writing proofs but that's about it.

Evaluate the course overall.: **5 (excellent)**

CS20 was definitely useful for preparation. I think also PHIL140 and CS141 as these classes helped me develop good knowledge of logic (especially useful in the first part of the course).

Evaluate the course overall.: **5 (excellent)**

CS 20 helped a lot, but I still struggled a lot even though I took it. While I learned how to write proofs in that class, and learned about graphs, that really only prepares you for pset 0. I wish I had a strong math/proof-writing background as the course seems easiest to those with that background.



Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **4 (very good)**

I self-studied by watching the videos from cs20. Was helpful, but I don't think it was crucial, and other factors such as general math competence are much more important for doing well in the class.

Evaluate the course overall.: **3 (good)**

CS 20

Evaluate the course overall.: **2 (fair)**

CS20 seems like good but not essential preparation for CS121 - otherwise, programming experience for the occasional programming problem

Evaluate the course overall.: **3 (good)**

CS20 helped me learn how to write proofs, but it definitely wasn't necessary to understanding the material.

Evaluate the course overall.: **4 (very good)**

Math for CS resources on Boaz's website are good. Am really glad I didn't take CS20.

Evaluate the course overall.: **3 (good)**

Self-studying using the MIT course and the review guide Prof. Barak provides is definitely useful. I don't think students who took CS 20 were significantly more prepared than those of us who self-studied.

Evaluate the course overall.: **4 (very good)**

Wish I had more proof-based math experience (like Math 23 instead of Math 21) or even just experience taking harder computer science classes (like CS 124 instead of coming straight from CS 51). I self-studied most of the MIT 6.044 discrete math class over the summer and that was pretty helpful, but only for the first pset. I actually think having a strong math/proof background is the most important thing for this class, since discrete math is easier to pick up.

Evaluate the course overall.: **3 (good)**

Math 23

Evaluate the course overall.: **1 (unsatisfactory)**

I did nothing to prepare. Never took proof-based math. I picked it up and it worked out fine.



Evaluate the course overall.: **4 (very good)**

CS 124 was helpful but not necessary. Math experience helped too.

Evaluate the course overall.: **3 (good)**

CS124, Math25

Evaluate the course overall.: **1 (unsatisfactory)**

CS124

Evaluate the course overall.: **5 (excellent)**

Math 25

Evaluate the course overall.: **1 (unsatisfactory)**

Math 23 and Yale's CS 365b

Evaluate the course overall.: **3 (good)**

I took CS20 but it didn't help very much. Some experience with reduction proofs would have been helpful.

Evaluate the course overall.: **4 (very good)**

Math 25 helped so, so much -- made me really proud and glad to have taken that class. Definitely should have treated this class more seriously.

Evaluate the course overall.: **4 (very good)**

CS124

Evaluate the course overall.: **3 (good)**

Helpful preparation: Maths 112 Real analysis gave me some proofs background and that helped in the problem sets. Wish I had done: CS20. I don't actually know whether I would have been better off. However, I think I enjoyed the course much more as time went on and I believe this was because I was slightly out of my depth at the start. However I also probably wouldn't have taken the course if CS20 was required because I'm not a concentrator and couldn't fit both in to my plan.

Evaluate the course overall.: **3 (good)**

Background material on website



Evaluate the course overall.: **3 (good)**

Stat110 concurrent (because the CS121 material on statistics overcomplicated everything with unnecessary over-notation)

Evaluate the course overall.: **3 (good)**

CS124

Evaluate the course overall.: **3 (good)**

STAT110 and CS20. This course will demand a facility with rigorous mathematical definitions and proofs, as well as facility with probabilistic reasoning in order to do well.

Evaluate the course overall.: **3 (good)**

I should have written more proofs. I was not comfortable with them and it hurt me quite a bit in this course. Especially since they didn't do a great job of preparing you to write them. A more relevant proof-writing section would have been great. The one we had was very "CS20" like and too general/basic to be very helpful. The homework postmortems were helpful though.

Evaluate the course overall.: **1 (unsatisfactory)**

CS 121 would have been unbearable if I had not taken CS 20 beforehand.

Evaluate the course overall.: **3 (good)**

I took CS124 last spring and I think that gave me a massive advantage over people who had not.

Evaluate the course overall.: **5 (excellent)**

Self-studying the materials on Boaz's website.

Evaluate the course overall.: **4 (very good)**

I did algorithms at UChicago and learned some stats stuff. There's nothing I wish I had done to prepare.

Evaluate the course overall.: **2 (fair)**

CS20 helped instill some comfort working with proofs, which helped but was not necessary. I think taking CS124 prior to CS121 tremendously helped in being able to adapt to the heavy workload of the course.

Evaluate the course overall.: **4 (very good)**

CS20. I wish I had taken AM106 before CS121



Evaluate the course overall.: **4 (very good)**

CS 124 definitely helped a lot.

Evaluate the course overall.: **4 (very good)**

CS124 was so good for preparing CS121 felt repetitive, which wasn't great.

Evaluate the course overall.: **3 (good)**

cs20!!! wish i had taken a stat course as well, but not nearly as necessary as cs20

Evaluate the course overall.: **4 (very good)**

I didn't do CS 20 but the summer plan you outlined on the website was very helpful!

Evaluate the course overall.: **4 (very good)**

CS-124, but I think the most important thing that's necessary for preparation is just a good amount of experience reading mathematical texts.

Evaluate the course overall.: **4 (very good)**

math, stats. some cs

Evaluate the course overall.: **2 (fair)**

CS124, Stat 110 Really don't understand how CS20 is supposed to help people with this course.

Evaluate the course overall.: **4 (very good)**

Proof-based math knowledge helped a lot, as well as a very base knowledge of coding. Nothing else would really have helped prepare.

Evaluate the course overall.: **3 (good)**

Having experience in discrete math and having taken CS 124 were quite helpful, as 124 helped build a lot of intuition about algorithm complexity and design that was used in the latter half of the course.

Evaluate the course overall.: **3 (good)**

CS20

Evaluate the course overall.: **4 (very good)**

Math 23 was helpful, and being interested in math in general was also helpful.



Evaluate the course overall.: **5 (excellent)**

Self-studying CS20 and taking CS124 was very helpful

Evaluate the course overall.: **1 (unsatisfactory)**

this class was the worst

Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **5 (excellent)**

CS20

Evaluate the course overall.: **1 (unsatisfactory)**

I had taken CS 124 last semester, and have had experience with proofs and notation from a Graph Theory course in high school. With these, I felt adequately prepared for the course.

Evaluate the course overall.: **5 (excellent)**

Math 23 (for proofs)

Evaluate the course overall.: **3 (good)**

Without having taken 124 last spring, I feel like it would be much harder for me to grasp certain concepts about complexity and reductions.

Evaluate the course overall.: **5 (excellent)**

CS 20 was really helpful. Early in the probability unit, I wished I had taken STAT 110, but after grasping the material I don't think it would have been too helpful beyond the initial problem set problems on basic concepts. The concept of amplification really felt like the core of our use of probability, and you don't really need STAT 110 to learn that.

Evaluate the course overall.: **5 (excellent)**

Knowing elementary set theory and having moderate mathematical maturity; I'm a joint physics and math major with no prior CS experience.

Evaluate the course overall.: **3 (good)**

Any heavily proof-based math course will suffice (Math 25, 55, maybe 23).



Evaluate the course overall.: **3 (good)**

CS 20.

Evaluate the course overall.: **4 (very good)**

I did the self-study course boaz recommended, it was helpful

Evaluate the course overall.: **2 (fair)**

I probably would have failed without having taken CS20. I wish I had been born more mathematically inclined, but sadly this wasn't the case. I'm not sure what else would have helped me.

Evaluate the course overall.: **5 (excellent)**

I took several proof-based math classes at the University of Chicago, and I think having had those experiences helped me greatly because I had experience with writing formal proofs, a large component of this course, and also because I was used to thinking deeply about the material and figuring out how to apply it creatively to different types of problems, which was also extremely important on the problem sets in this course.

Evaluate the course overall.: **5 (excellent)**

MATH 25A, MATH 25B. I wish I took CS 124 first. For the reductions to prove NP-Completeness, always tried to use 3SAT because that was the one we spent a lot of time on in class but those who took CS 124 were more comfortable using others (MAXCUT, ISET, etc.).

Evaluate the course overall.: **3 (good)**

I didn't take CS 20, and I don't regret this. I was very worried at the beginning of the course about lacking the mathematical background (I'd never written proofs before), but I don't think this is what made the course difficult. I picked up the math concepts/proofs as I went along, and I wouldn't say that this held me back, especially since the course isn't very strict about proofs.

Evaluate the course overall.: **3 (good)**

I watched all of the lectures from the online MIT course recommended in lieu of CS20 on the CS121 website and did problem sets from that course.

Evaluate the course overall.: **5 (excellent)**

I took CS20, which was moderately helpful in giving a first exposure to some topics like proofs, state machines, etc. However, I think I would have done similarly without taking it.

Evaluate the course overall.: **4 (very good)**

CS 50



Evaluate the course overall.: **4 (very good)**

I've taken a lot of math classes, which I think helped.

Evaluate the course overall.: **4 (very good)**

Math 25A/B, CS 124. After taking these courses, CS 121 was not very hard at all. In my opinion, the CS department should make people take proof-based math or make CS 20 more legit; this is the main reason people suffer in this course. You should also bring back CS 125!

Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **5 (excellent)**

CS 124, Stat 110. In my opinion the required math is easy to pick up on the fly

Evaluate the course overall.: **1 (unsatisfactory)**

Math

Evaluate the course overall.: **5 (excellent)**

CS124

Evaluate the course overall.: **3 (good)**

CS20 was extremely helpful

Evaluate the course overall.: **4 (very good)**

Math 101 and Stat 110 were both very useful for me.

Evaluate the course overall.: **4 (very good)**

Having some experience reading mathematical texts was pretty helpful.

Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **3 (good)**

CS124



Evaluate the course overall.: **1 (unsatisfactory)**

124

Evaluate the course overall.: **4 (very good)**

Wish I had taken a proof based class before this one - thankfully my friend helped a ton at the beginning but I couldn't imagine trying to do it without her.

Evaluate the course overall.: **4 (very good)**

CS 124 helped a lot.

Evaluate the course overall.: **4 (very good)**

Math 25 helped a lot in being rigorous and thinking mathematically. I would have liked to have taken a graph theory class to understand topics related to the graph problems better, but the background material on graphs was definitely enough for what was needed in this course.

Evaluate the course overall.: **3 (good)**

CS20 really helped a lot with understanding how to write proofs and what the basic types of proofs were. I wish I had taken linear algebra, just so I could understand the quantum stuff more.

Evaluate the course overall.: **5 (excellent)**

CS124

Evaluate the course overall.: **5 (excellent)**

Having a strong mathematical background was very helpful.

Evaluate the course overall.: **4 (very good)**

I have taken Math 55, which helped tremendously when it comes to proof based mathematics (as in, most of the course), but of course that is likely overkill.

Evaluate the course overall.: **5 (excellent)**

CS 124, CS 141

Evaluate the course overall.: **5 (excellent)**

textbook all the way. CS124 was very helpful too



Evaluate the course overall.: **4 (very good)**

CS20, CS124

Evaluate the course overall.: **3 (good)**

CS 124 helped a lot. Being familiar with some proof techniques was also super helpful.

Evaluate the course overall.: **4 (very good)**

Math 25 was most helpful. CS124 also helped a bit.

Evaluate the course overall.: **3 (good)**

Math 23c helped prepare me for the proof part of the class.

Evaluate the course overall.: **2 (fair)**

CS20 definitely helps but probably not absolutely necessary.

Evaluate the course overall.: **4 (very good)**

In order of relevance: 1. CS 124 2. Math 23a 3. Math 23c

Evaluate the course overall.: **4 (very good)**

CS 124 made my experience drastically easier, as almost all of the material was already covered in CS124. Aside from that I am grateful I was also enrolled in STAT 110 and feel like the M23 series also helped with mathematical formality / maturity.

Evaluate the course overall.: **4 (very good)**

Go to section more because I barely went and stay on top of the psets and the solutions to them after because that is the material you truly need to know for the course. I didn't take CS20 or know computational theory but this class still worked out fine.

Evaluate the course overall.: **5 (excellent)**

CS50/51. Sure, I wish I started coding at 12 as well so this all felt more accessible.

Evaluate the course overall.: **3 (good)**

CS20, reading Ch.0-2 over the summer I wish I had been more prepared, but not sure how I could've done that other than by reading the whole textbook over the summer / taking the course twice or something



Evaluate the course overall.: **1 (unsatisfactory)**

Expository Writing 20. Also I wish I took Cumrun Vafa's seminar "Physics, Math, and Puzzles." I think it would be excellent preparation.

Evaluate the course overall.: **4 (very good)**

Math 23abc, taking Math 152 and Stat 110 concurrently

Evaluate the course overall.: **2 (fair)**

learn reductions.

Evaluate the course overall.: **1 (unsatisfactory)**

CS 124

Evaluate the course overall.: **3 (good)**

CS20

Evaluate the course overall.: **4 (very good)**

Math 55

Evaluate the course overall.: **4 (very good)**

CS20, CS152

Evaluate the course overall.: **1 (unsatisfactory)**

CS20, proof-based math

Evaluate the course overall.: **4 (very good)**

I did the self-study over the summer and felt comfortable with everything we did! I think besides proof by induction and learning about functions/relations and set, you don't really need much else! Just getting used to reading/writing proofs was important I think.

Evaluate the course overall.: **1 (unsatisfactory)**

CS20 helped a bit, but you need some other theory intro in my opinion.

Evaluate the course overall.: **4 (very good)**

CS20, CS124



Evaluate the course overall.: **3 (good)**

CS20

Evaluate the course overall.: **3 (good)**

TAKE CS20, TAKE A THEORETICAL MATH COURSE, PROOF BASED. DO IT OR SUFFER.

Evaluate the course overall.: **3 (good)**

I would definitely recommend taking CS 20 before this course. CS 50 and 51 are also very good prep.

Evaluate the course overall.: **2 (fair)**

cs20

Evaluate the course overall.: **2 (fair)**

I wish i had taken CS20

Evaluate the course overall.: **4 (very good)**

CS20

Evaluate the course overall.: **3 (good)**

I can't point to one specific course, but I'd taken several proofs based courses prior to this (combinatorics and graph theory, rings, intro set theory), and felt significantly more equipped than most of the students, particularly those who had only taken CS 20 or who hadn't taken it at all. A lot of kids just didn't seem to know how to write proofs. (My partner for most of the pssets struggled with this throughout. In my opinion, he finished the course still not really knowing how to right a proper reduction proof)

Evaluate the course overall.: **2 (fair)**

CS20

Evaluate the course overall.: **4 (very good)**

I did not take CS 20 and did not feel like I needed to take it - the material in this course is very specific to this course, so taking general discrete math or whatever I don't think would be that applicable What I did do is spend about a week going through the preparatory material on the course website, and that was very helpful. In particular, you should know what graphs are and some very basic graph theory, set notation, logical expressions, proof by induction (well, sort of - not sure if we used this much), and a few more basic things on the prep part of the website. But preparing for this class without CS20 or other background is SO doable if you give yourself a solid week or so before the course begins.



Evaluate the course overall.: **5 (excellent)**

STAT 110 for the one probability problem set. Felt intimidated by lack of proof background initially but the subset of possible CS proof techniques we learned were pretty accessible.

Evaluate the course overall.: **4 (very good)**

Math 25a/b

Evaluate the course overall.: **1 (unsatisfactory)**

CS-124 was helpful. Just having a lot of programming and math prior experience was also helpful. What happened to CS 125? I would have loved to take that.

Evaluate the course overall.: **4 (very good)**

Math 25a/25b

Evaluate the course overall.: **1 (unsatisfactory)**

I'm a math major, so I already had experience in the kind of problem solving we did in the class.

Evaluate the course overall.: **1 (unsatisfactory)**

I wish I had taken CS20, but a lot of people said it didn't really help them a whole lot. I wish I had taken more theoretical math courses.

Evaluate the course overall.: **3 (good)**

All of the math I've ever done helped above all else. I wish I had more time in the semester to devote to this class.

Evaluate the course overall.: **5 (excellent)**

Math 152, was super helpful

Evaluate the course overall.: **2 (fair)**

CS124

Evaluate the course overall.: **2 (fair)**

CS 20, CS 51 , Stat 110 I wish I had more proof experience than what CS 20 offered. I think CS 20's material needs to change in accordance with the new material in this course.



Evaluate the course overall.: **4 (very good)**

Study in proof structure and math.