2023 Fall Individual Report FAS-COMPSCI 121-Introduction to Theoretical Computer Science 001 Boaz Barak

Project Title: **2023 Fall Harvard FAS Course Evaluation**

Course Audience: **99**
Responses Received: **91**
Response Ratio: **92%**

Report Comments

Note:
The order that the questions appear on this report is not the same as the way the questions were displayed to students.
The order has been changed to make the report more readable.

Creation Date: **Tuesday, January 2, 2024**

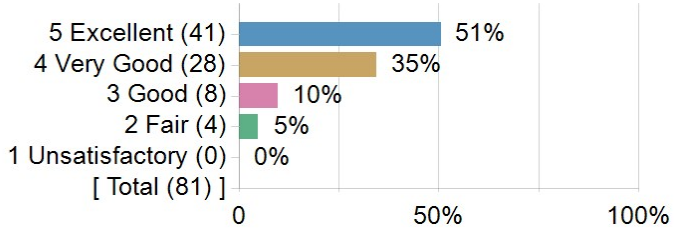
General Course Questions

Course General Questions

	Count	Excellent	Very Good	Good	Fair	Unsatisfactory	Course Mean	Dept Mean	Division Mean
Evaluate the course overall.	81	51%	35%	10%	5%	0%	4.31	4.08	3.98
Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.)	78	47%	37%	12%	1%	3%	4.26	4.17	4.03
Assignments (exams, essays, problem sets, language homework, etc.)	77	51%	35%	13%	1%	0%	4.35	3.95	3.85
Feedback you received on work you produced in this course	78	46%	36%	13%	4%	1%	4.22	3.78	3.72
Section component of the course	64	36%	33%	23%	8%	0%	3.97	4.05	3.96

Evaluate the course overall.

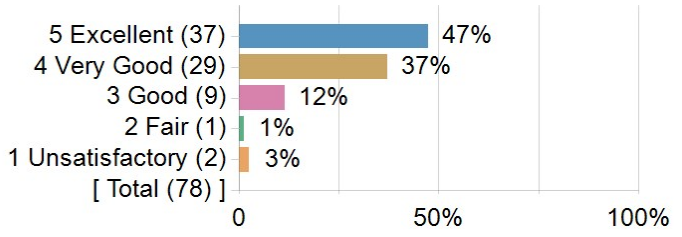
Evaluate the course overall.



Options	Score	Count	Percentage
Excellent	5	41	51%
Very Good	4	28	35%
Good	3	8	10%
Fair	2	4	5%
Unsatisfactory	1	0	0%
Statistics			**Value**
Response Ratio			82%
Mean			4.31
Median			5.00
Standard Deviation			0.85

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.)

Course materials (readings, audio-visual materials, textbooks, lab manuals, website, etc.)



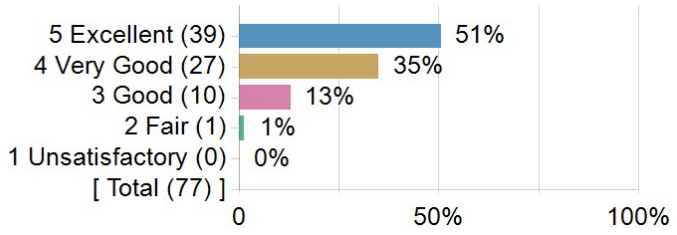
Options	Score	Count	Percentage
Excellent	5	37	47%
Very Good	4	29	37%
Good	3	9	12%
Fair	2	1	1%
Unsatisfactory	1	2	3%
Statistics			**Value**
Response Ratio			79%
Mean			4.26
Median			4.00
Standard Deviation			0.90

Add comments about course materials?

Comments
Lectures covered chapter highlights of textbook, so it's more helpful to read the textbook honestly.
Very solid textbook
I think the textbook can be difficult to read at points but I know many people who love it
The book is a little bit hard to digest but I think overall it introduced the concepts fairly well
Textbook was very clear and well–written
The textbook often got messy and lost in piles of examples or such. As a work in progress I think a bit of restructuring is in order, and I myself often turned to Sipser's textbook's formalizing of certain ideas instead. The website was incredibly helpful and illustrated all course logistics.
See my previous comment about the textbook. TL;DR it needs revising but mainly the textbook shouldn't play as huge a role as it does. The other thing is I don't think we should have as many of the advanced topics. Randomized algorithms are awesome and so helpful to know about, and cryptography was cool, but the others — especially quantum computing — weren't explained well enough (because of time) to really be meaningful. Maybe I'm biased on the ML theory because I already knew it from CS 181, but that part didn't seem that interesting, and we could've had the problem about it on the problem set without the lecture.
The book is dense and hard–to–read, goes on tangents, and contains a lot of bloat.
Readings were a bit long but everything else was good.
The textbook was written by Professor Barak and was awesome. Sometimes, the proofs got a bit too detailed, so skipping some of the theorems became necessary. But it was a great place to return to when studying or catching up on content.
Not much of a point in having both Canvas and Boaz's course website—I vote move it all to one or to the other (preferably Boaz's website; it looked like it was well set up). Ed was great, though—definitely keep that! Also a minor thing, but it would be better to put the psets etc. on Canvas or Boaz's website, it's kind of out of the norm to put them on Ed.
The course textbook was generally well written.
Textbooks and slides and section materials... there are too much materials and sometimes the less is better for helping ppl actually do the readings and retain them.
Great textbook! Appreciated the practice tests and the lectures and sections being recorded to reference back to later.
I did not love the textbook because it was slightly unconventional and sometimes difficult to find external resources
The readings are by far the best part of this class. This is because the professor has put in a lot of time to really create a reading material that covers everything needed in class.
Boaz's textbook is absolutely fantastic, and it obviously aligns well with the course content.
Textbook was hard to follow. I wish there were supplemental options
Textbook is good and should continue to be expanded.
Very thorough and made valuable re–organization when it became necessary for Ed.
The textbook (Perusall) and the readings system of the class are outright bad. To the point where you might lose points even if you do the reading through PDF.
I found that the textbook as times could be a little daunting to read and found myself relying on using the problem sets to build intuition and learn about each topic.
Textbook is a bit typo–ridden in the latter half of the course and the images aren't great, but it's still a better resource than anything else on the internet for theory (and much much better than Sipser), so fair enough, I suppose.
Great textbook.
Textbook was amazing.
I personally really dislike the Perusall system. I would much prefer something like a short answer or multiple choice reading quiz graded for completion for each lecture/reading instead. This would better both help students check their own understanding (e.g. if they see a question that they have no idea how to approach), and allows more flexibility in learning styles (e.g. when I read textbooks I almost always either read on an iPad or on physical paper copies so that I can mark it up most easily).
Textbook is verbose but informative.
it was actually really nice to have the readings to go along w lectures!
Very helpful section and preparation for exam material. A book that is too wordy and becomes difficult to use.
The textbook is quite dense and rather difficult to read through. I think it's just the material however, as a lot of intuitive explanations were given.
The textbook is very proof heavy, which is good, but the notation can feel a little hard to follow.

Assignments (exams, essays, problem sets, language homework, etc.)

Assignments (exams, essays, problem sets, language homework, etc.)



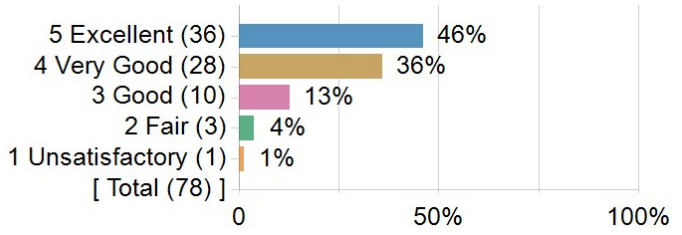
Options	Score	Count	Percentage
Excellent	5	39	51%
Very Good	4	27	35%
Good	3	10	13%
Fair	2	1	1%
Unsatisfactory	1	0	0%
Statistics			**Value**
Response Ratio			78%
Mean			4.35
Median			5.00
Standard Deviation			0.76

Add comments about course assignments?

Comments
Problem sets were challenging but doable with office hours help! Midterms were difficult, final exam was more manageable.
There was just one specific problem on pset 4 on DNA computers that is nearly impossible to sort out all the details of, and I went through a lot of pain to get to a point where I felt my answer was satisfactory. Maybe change that problem please lol.
Loved the pset grading policy using extra credit so there isn't undue stress about getting things wrong.
Mostly interesting and thought–provoking. But the point distributions on some of the problem sets I thought did not reflect the amount of time spent on each problem
I disliked the PSETs on boolean SIZE hierarchy as the inclusion and exclusion of certain notation made it hard for me to fully capture what the problem was asking each time. This was reinforced by the mediocre grading. However, most problems were incredibly interesting and a joy to solve.
the problem sets were fun (except the DNA computer), the exams were fair. No complaints.
Pset were fun and optimally challenging.
Exams weren't too difficult, but definitely tested the content well. Bonus points made them less stressful and brought up the distribution a ton.
I thought they were great!
The problem sets contained some pretty interesting problems.
Too much work. There is a sweet spot for enough work where ppl enjoy learning but this class is just too much
Psets were generous with extra credit which was very helpful and kept me from stressing about minutiae.
the assignment structure with bonus points and reflections helped make grading less of a focus and really emphasized the learning part of the course
I sometimes felt like I spent more time typing the PSETs up than solving and actually learning something from them. The exams were super fair, though.
Really liked the bonus points structure
Assignments were very constructive and helpful, with a very valuable and healthy extra points system.
The PSETS were excellent.
The homework was fine (except that DNA computer question on PSET 3...), some things are actually quite fun (prove BB is uncomputable, for instance). The final was great!
Great assignments.
pset questions were hard to parse
I found it a bit annoying that the midterms were at 7pm in the SEC. It would be better if they were during the usual class time.
Really appreciate all of the bonus! It makes this class way less stressful.
Please get rid of DNA computer that problem set was very annoying.
I loved the problem sets, super interesting questions & I liked having the bonus questions to see what harder questions might look like using the same fundamental concepts!
Interesting psets with a lot of opportunity for extra credit though with sometimes very confusing and/or hard questions. The exams were nicely done.
Half of the work of doing a problem set question is just understanding what the question is asking! I'm not sure if this was on purpose, was a result of my own lack of comprehension, or was simply a result of poorly worded and framed problems.
I really loved the problem sets. They forced me to think a lot more creatively, and I learned so much by powering through the sets myself.

Feedback you received on work you produced in this course

Feedback you received on work you produced in this course



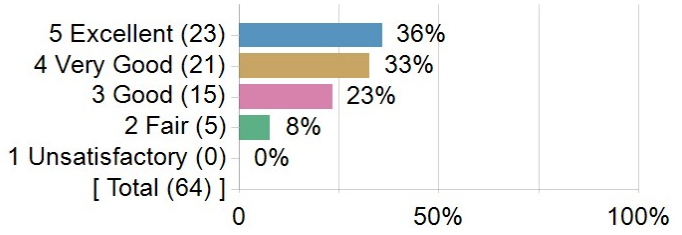
Options	Score	Count	Percentage
Excellent	5	36	46%
Very Good	4	28	36%
Good	3	10	13%
Fair	2	3	4%
Unsatisfactory	1	1	1%
Statistics			**Value**
Response Ratio			79%
Mean			4.22
Median			4.00
Standard Deviation			0.91

Add comments about course feedback?

Comments
Really appreciated the TFs!
TFs wrote really helpful comments on Gradescope for problem sets! Also solutions for problem sets and exams were super helpful to see what was incorrect about the work we submitted / what the teaching staff was looking for in a good, solid solution.
The problem sets were always graded on time and there was a good amount of course structure
Points were taken off valid solutions with no explanation, while logical fallacies in my own solutions went completely unnoticed. Most critically, there was a lack of explanation as to why grading decisions were made, meaning I could not learn from my past mistakes.
CAs were always very timely with grading, and offered helpful feedback! However, there were issues with consistency in grading between CAs.
Again, the comment about arbitrary requirements for rigor holds. Also, one time a TF corrected a regrade request on Gradescope and then pretended like my regrade request was incorrect, which was quite unprofessional.
Feedback was provided in a timely manner and was very comprehensive and helpful.
Gradescope worked great.
For the most part, the TFs were fantastic about giving detailed, helpful feedback when grading the PSETs.
Feedback was timely and detailed. My complements to the TFs.
The pairing of the thorough gradescope point system and comments alongside problem set solutions allowed for growth after our psets were graded.
Comments appeared randomly, and sometimes were too confusing in figuring out what part of my solution was wrong.
Feedback on PSETS and exams was actually great and very useful for improving.
Great.
pset graders were very harsh
Great.
Feedback was always quick and concrete.
Some of the feedback on the problem sets felt arbitrary, a few ticky tacky points taken off here and there without much reasoning behind it.

Section component of the course

Section component of the course



Options	Score	Count	Percentage
Excellent	5	23	36%
Very Good	4	21	33%
Good	3	15	23%
Fair	2	5	8%
Unsatisfactory	1	0	0%
Statistics			**Value**
Response Ratio			65%
Mean			3.97
Median			4.00
Standard Deviation			0.96

Add comments about the course section?

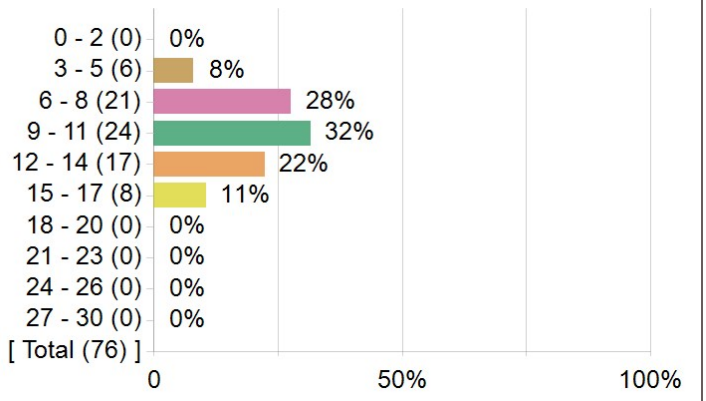
Comments
I could not make most sections but I really enjoyed 121.5!
I don’t think I really learned more in the section to be honest – they kind of just went over the readings on a more elementary level but it wasn’t super helpful for my understanding of the problem sets
121.5 was always an incredible experience, but for section itself I found TFs often just read through the material and didn't explain it all too well.
I didn't go often but it seemed helpful.
Most of the sections were either in the SEC, clashed with my schedule, or both. The teaching in section was fine but I could get most of the benefit from just reading the section materials.
Section was not good. It was just regurgitation of everything we had learned before.
Section was a very effective way to review the material covered in lectures and in the textbook readings.
Section seemed a tiny bit superfluous.
It felt like the TFs were themselves not prepared to teach section and sometimes improvised critical subjects.
Kitty was great, as usual!
Thorough calendar, communication, and attendance system which ensures students were able to get the most out of section.
Sections felt more like mini–lectures than a time to build practical experience.
Kenny was great and very clear.
Section was a great way to put into practice what we had learnt in lecture and prepare for the problem sets.
Kenny is great
Never went.
Sections were engaging and the section material was extremely helpful for understanding and reviewing the current week's content.
I attended Yueqi and Neeyanth's section almost weekly, and they were extremely helpful in elaborating on unclear concepts, such as in proving NP–completeness and Chernoff Bound and amplification in regards to randomized algorithms. I just want to give them a shout–out, and hopefully this is an area in which I can do so.

Requirements - What did this course require of you?

On average, how many hours per week did you spend on coursework outside of class? Enter a whole number between 0 and 168.

Frequency chart and mean excludes students who answered 31 or more hours.

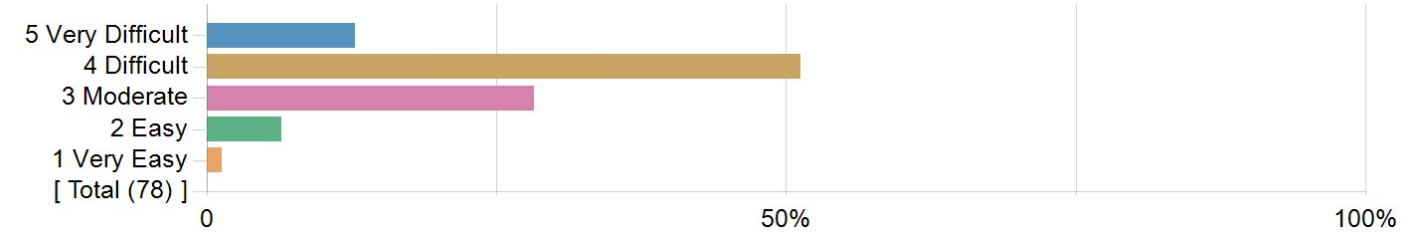
On average, how many hours per week did you spend on coursework outside of class? Enter a whole number between 0 and 168.



Statistics	Value
Response Count	76
Response Ratio	77%
Mean	9.84
Median	10.00
Mode	10
Standard Deviation	2.98

How difficult did you find this course?

How difficult did you find this course?



Options	Score	Count	Percentage
Very Difficult	5	10	13%
Difficult	4	40	51%
Moderate	3	22	28%
Easy	2	5	6%
Very Easy	1	1	1%

Statistics	Value
Response Ratio	79%
Mean	3.68
Median	4.00
Standard Deviation	0.83

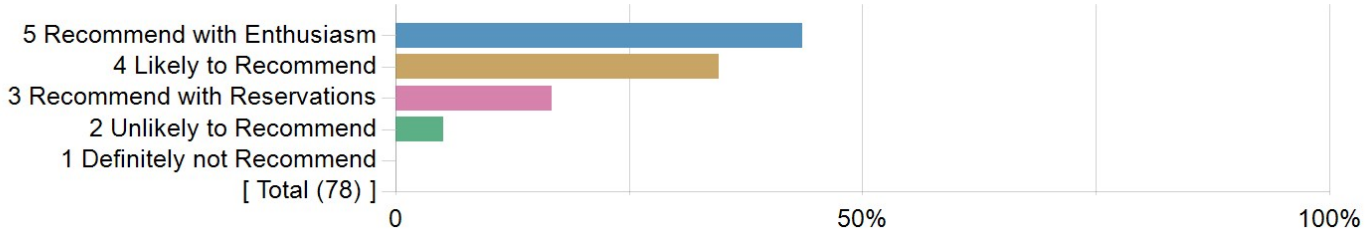
What was/were your reason(s) for enrolling in this course? (Please check all that apply)

Options	Count
Elective	11
Concentration or Department Requirement	71
Secondary Field or Language Citation Requirement	1
Undergraduate General Education Requirement	0
Expository Writing Requirement	0
Foreign Language Requirement	0
Pre-Med Requirement	0
Divisional Distribution Requirement	0
Quantitative Reasoning with Data Requirement	0

Recommendations - Would you recommend this course?

How strongly would you recommend this course to your peers?

How strongly would you recommend this course to your peers?



Options	Score	Count	Percentage
Recommend with Enthusiasm	5	34	44%
Likely to Recommend	4	27	35%
Recommend with Reservations	3	13	17%
Unlikely to Recommend	2	4	5%
Definitely not Recommend	1	0	0%

Statistics	Value
Response Ratio	79%
Mean	4.17
Median	4.00
Standard Deviation	0.89

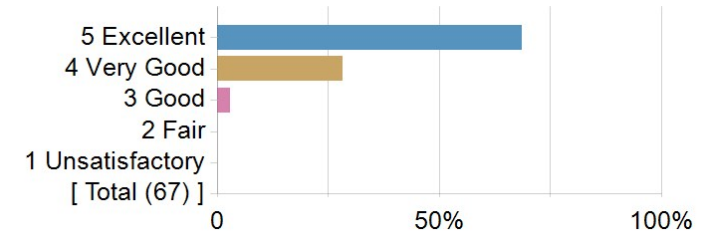
Evaluation of Instructors

General Instructor Questions

	Count	Excellent	Very Good	Good	Fair	Unsatisfactory	Instructor Mean	Dept Mean	Division Mean
Evaluate your Instructor overall.	67	69%	28%	3%	0%	0%	4.66	4.39	4.34
Gives effective lectures or presentations, if applicable	67	61%	25%	10%	3%	0%	4.45	4.30	4.23
Is accessible outside of class (including after class, office hours, e-mail, etc.)	63	54%	21%	19%	5%	2%	4.21	4.22	4.22
Generates enthusiasm for the subject matter	67	85%	10%	3%	1%	0%	4.79	4.50	4.45
Facilitates discussion and encourages participation	52	79%	15%	4%	2%	0%	4.71	4.46	4.37
Gives useful feedback on assignments	22	91%	9%	0%	0%	0%	4.91	4.33	4.16
Returns assignments in a timely fashion	33	88%	12%	0%	0%	0%	4.88	4.31	4.16

Instructor

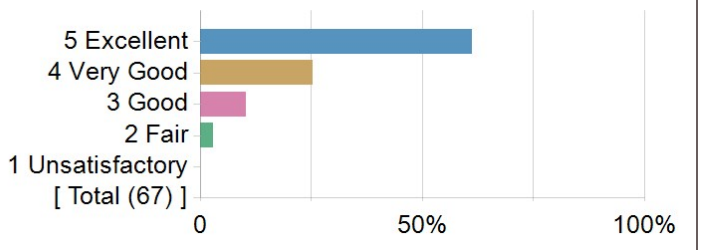
1. Evaluate your Instructor overall.



Options	Score	Count	Percentage
Excellent	5	46	69%
Very Good	4	19	28%
Good	3	2	3%
Fair	2	0	0%
Unsatisfactory	1	0	0%

Statistics	Value
Response Ratio	68%
Mean	4.66
Median	5.00
Standard Deviation	0.54

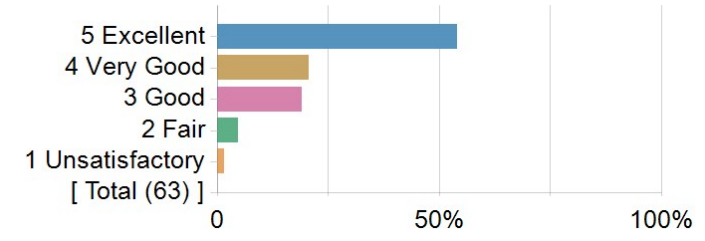
2. Gives effective lectures or presentations, if applicable



Options	Score	Count	Percentage
Excellent	5	41	61%
Very Good	4	17	25%
Good	3	7	10%
Fair	2	2	3%
Unsatisfactory	1	0	0%

Statistics	Value
Response Ratio	68%
Mean	4.45
Median	5.00
Standard Deviation	0.80

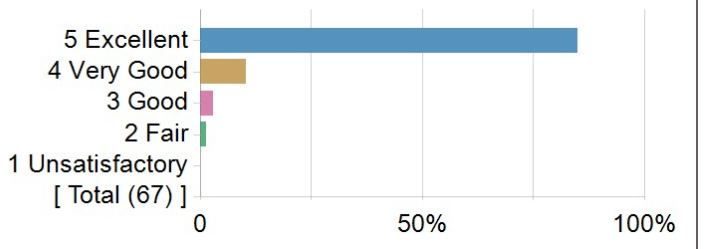
3. Is accessible outside of class (including after class, office hours, e-mail, etc.)



Options	Score	Count	Percentage
Excellent	5	34	54%
Very Good	4	13	21%
Good	3	12	19%
Fair	2	3	5%
Unsatisfactory	1	1	2%

Statistics	Value
Response Ratio	64%
Mean	4.21
Median	5.00
Standard Deviation	1.02

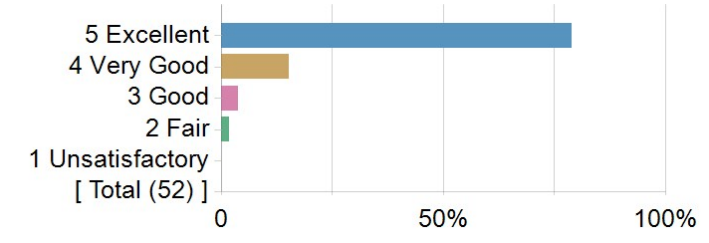
4. Generates enthusiasm for the subject matter



Options	Score	Count	Percentage
Excellent	5	57	85%
Very Good	4	7	10%
Good	3	2	3%
Fair	2	1	1%
Unsatisfactory	1	0	0%

Statistics	Value
Response Ratio	68%
Mean	4.79
Median	5.00
Standard Deviation	0.57

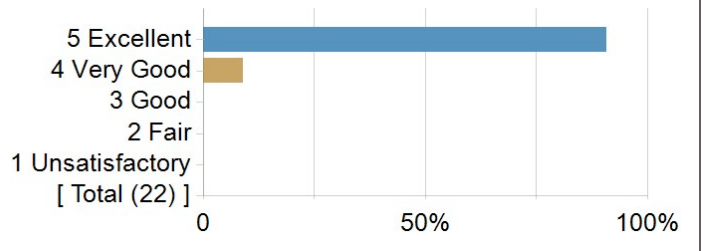
5. Facilitates discussion and encourages participation



Options	Score	Count	Percentage
Excellent	5	41	79%
Very Good	4	8	15%
Good	3	2	4%
Fair	2	1	2%
Unsatisfactory	1	0	0%

Statistics	Value
Response Ratio	53%
Mean	4.71
Median	5.00
Standard Deviation	0.64

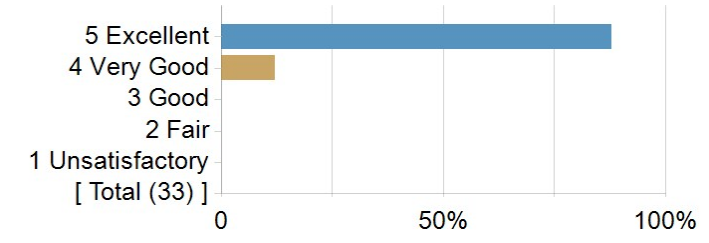
6. Gives useful feedback on assignments



Options	Score	Count	Percentage
Excellent	5	20	91%
Very Good	4	2	9%
Good	3	0	0%
Fair	2	0	0%
Unsatisfactory	1	0	0%

Statistics	Value
Response Ratio	22%
Mean	4.91
Median	5.00
Standard Deviation	0.29

7. Returns assignments in a timely fashion



Options	Score	Count	Percentage
Excellent	5	29	88%
Very Good	4	4	12%
Good	3	0	0%
Fair	2	0	0%
Unsatisfactory	1	0	0%

Statistics	Value
Response Ratio	33%
Mean	4.88
Median	5.00
Standard Deviation	0.33

General Course Questions - Comments

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

Comments
Lectures were great and the grading scheme/bonus points really encouraged learning and decreased stress! The teaching staff was also awesome.
The textbook was quite detailed, comprehensive, and helpful for problem sets and exams.
Overall, this course covered some pretty difficult topics in a digestible manner, and Boaz explained the high–level key takeaways that really stick with you as you continue to work in CS and think about what is computable in reasonable time. It's a solid introduction to theoretical CS, should you choose to pursue that route!
sequence/progression of topics; lectures
The strongest parts of the course were the wide breadth of the curriculum, especially by the end, where we were able to cover various parts of computer science.
The bonus problems on the problem set had some interesting material to think about.
Professor Barak’s lectures, the problem sets, the TFs
I thought the pset grading protocols fostered a very comfortable and welcoming learning environment :) great course staff, very responsive on class forums, lots of resources for different learning styles
Boaz Barak is an excellent instructor, he communicates ideas well, and lectures are incredibly well done. PSETs and exams are graded quickly, and the course structure is incredibly well organized and documented on his website. Course logistics are incredibly well handled overall.
This course has alot of breadth. You learn about everything from how circuits work together to compute to quantum computers!
I really liked the material —- so fascinating. The lectures were engaging and generally taught well, the problem sets were difficult but fun, and the grading scheme made the class much less stressful than it could've been.
The reflection assignments and the high number of bonus points on the problem sets and exams were very good for relieving stress and focusing on learning.
The bonus points really remove a lot of pressure. Boaz is also very understanding.
I think this course was very well–planned and structured. I also liked how the textbook was on Perusall which allowed you to get specific questions answered by the staff fairly quickly.
Lots of opportunities to reflect on your work and prove your knowledge of the subject matter.
I left the course more articulated, I learned reductions, and learn many cool things to know for a computer scientists.
This course had good lectures, a great textbook, not too difficult exams, and tons of content. The strengths are the content itself and the lectures.
Great grading setup (although maybe slightly too many bonus points on final), fun psets, great lectures/we love Boaz, and good material
The lectures were very engaging and the course material was extremely interesting.
The topics are must learns for anyone in CS. There are A LOT of components and moving pieces that would help you learn hard materials IF you put in the time. Boaz and Kitty and all the TFs are very nice and supportive.
Very organized course, great textbook, lectures, and office hours.
The readings are by far the best part of this class. This is because the professor has put in a lot of time to really create a reading material that covers everything needed in class.
This course was solid, and Boaz is clearly passionate about the subject material.
Covered interesting topics
Very well organized, grades came back in time, plenty of opportunities to get extra credit, emphasis on learning, clear structure, insomnia cookies were given at the end of every test!
Interesting selection of topics. Fair assignments and exams with generous bonus points. Friendly teaching staff.
The course provided a thorough understanding of what computers and algorithms can and cannot do, as well as what they can do in a "reasonable" time frame.
The section notes as well as the homework solution and reflection system allowed for extremely effective review and support systems.
I very much enjoyed the textbook, which not only taught the material well, but sparked interest in the subject.
The lectures are incredibly interesting and engaging. Professor Barak makes the material digestible and is always happy to answer any questions/spend more time on the confusing bits.

Comments
I found that this course heavily encourages self–study, but also offers leniency grades–wise, especially with bonus points.
The material was very interesting and it was taught well
Good content of the course. Boaz is invested in the class. Psets are not too tedious. Grading is generous.
The material is fascinating, and Boaz is enthusiastic about the material to the point where he can make most things in TCS interesting.
Good textbook.
The best part of the course is that it is taught incredibly well. Boaz is an exceptional instructor and makes extremely high quality material.
Great textbook. Fantastic TFs. Particularly Nick and Kenny
The course was very organized and Boaz is a great professor
The book is really good. The concepts are explained clearly and examples are provided to make comprehension easier. The problem sets are also very well–designed to help with solidifying understanding.
Boaz is a good teacher but the material is a bit tough and the textbook is super long and convoluted
I think the course covered a lot of material and had a good connection between them.
Awesome textbook, great lectures, fun psets, good exams, cool professor
It was well run and the lecture notes, problem sets, and exams all reflected the important content of the course
Interesting PSETS (except DNA computer–worst question I've answered ever) which really helped me understand the concepts (especially in the second half of the course).
great lectures, great teaching staff, fun & reasonable problem sets, etc. there was quite a bit of controversy regarding the problem sets on exam weeks, but i'd rather have one really bad week and then a good week, so i actually liked that a lot.
Extremely accessible and clear introduction to theoretical science. I came into this class pretty nervous because I certainly don't have a strong background in TCS, but found that I was more than capable of completing problem sets or approaching difficult questions after attending lectures/reading the text (though the textbook took a long time to get through sometimes). Love this class so much.
Interesting content. Nice lectures. Very helpful sections with section materials. Interesting homework. Lots of opportunities for extra credit. Useful and quick feedback. Lots of preparation material for the midterms and the final.
This course is one of the best designed courses at Harvard! The material is very interesting, and relevant to anyone interested in theoretical computer science. The problem sets are challenging, but not too stressful because of the bonus points. The lectures were very helpful, interesting, and interactive.
The course was challenging, especially for someone who did not have much experience in the material beforehand, but the lectures were extremely informative, and I had a great time solving the problem sets. They were very thought–provoking, and I fell in love with the material.
I think this course provided a great overview of theoretical, the the grading system gave less pressure on learning for grades and more for enjoyment.

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

Comments
Perusal was not a great platform and I eventually stopped reading the textbook — it felt to useful to reference/encourage students to read but maybe not as a very strict requirement
There was just one specific problem on pset 4 on DNA computers that is nearly impossible to sort out all the details of, and I went through a lot of pain to get to a point where I felt my answer was satisfactory. Maybe change that problem please lol.
Relating to the above, I thought it was interesting how lots of what we talked about over the course tied to many different topics and fields, but I would have liked it if we had more time to discuss these other topics, since it felt like the the topics near the end of the class were quite rushed and crammed into just one lecture (e.g., cryptography, machine learning, quantum computing).
The textbook could be vastly improved; a lot of the proofs (especially in the circuit chapters) were rather boring, and some explanations felt long for no reason. I think the textbook could have been much more engaging and the concepts better explained. Some of the problem set solutions were pretty easy to solve and intuitively get, but writing the proofs took too much time. Although the bonus problems are intended to allow students to not focus too much on the little details, the staff should understand that many students do try to get everything right and hence will be caught up on trying to make everything perfect.
I found the TF/grading quality this year wasn't the greatest, and oftentimes problems on PSETs were left entirely without feedback. I also personally disagree with some of the directional choices Professor Barak took in teaching the material of CS theory (Automata

Comments
Theory to me feels much more important and interesting than properties of boolean circuits), but overall this I do not mind that much.
Perhaps slightly more inclusion of coding into the problem sets. I found that I could do proof of implementation using code but it is rarely blatantly said to use coding to do problems.
Mainly this course should really reevaluate how it uses the textbook. If the lectures are going to be a very introductory version of part of the textbook material, then the reading should be assigned for after the lecture, not before. But even more than that, I would recommend adding a third lecture every week and then going in depth on the important proofs in all lectures (rather than just some). In this iteration, because textbook reading is hard and this one is still a work–in–progress, I would spend 4–8 hours per week doing the textbook readings (especially in the beginning where you're still getting used to the paradigm). I would've liked the readings to be supplemental to the lectures, but instead it was the other way around: if you actually read the textbook beforehand and worked through the proofs, the lectures would be completely useless. The material is hard enough that it takes some digesting to understand, and I get that it would be difficult to cover most of it in lecture, but with a third lecture I think it would be doable. That way the students would have to do less of the heavy pedagogical lifting themselves and could spend more time on problem sets (or going to section). Without that, it was super frustrating to be behind constantly while putting in so much more time than expected on the reading (it was certainly not advertised this way).
I think the system of readings with Perusall could be improved. Many times it wasn't clear what parts of the reading were important and which were not as important. The textbook chapters were long and sometimes a bit repetitive, making it hard to finish the reading in time for each lecture. Moreover, in many cases I found that I preferred to do the readings after having been exposed to the material in lecture, since the textbook was sometimes a bit dense. However, the course structure made this difficult. Another big issue was the discrepencies in grading between the different CAs. I found that the same quality of work could get drastically different grades based on which CA was assigned to grade my problem set.
Not much actually to improve. Book is hard to read though.
When lecturing, I would have appreciated more legible and structured writing on the board as well as explaining the complex topics of the class slowly and with examples. I also think that the OH could be scheduled better, which I guess would be difficult to control given everyone's schedule.
Section seemed to be a bit behind content covered in lectures. I wish section was canceled the week before an exam so that more office hours and review sessions are available. I wish there was also no pset due the same week as an exam.
I don't have a piece of feedback here.
Sometimes, the lectures got a bit off topic or distracted. Sometimes, Professor Barak would mumble a bit when explaining the end of a proof, and everyone would get lost. Overall, he is a great lecturer though.
The only thing I think is to be consistent in the requirements for rigor; I was kind of frustrated to go to office hours and learn that certain components of psets weren't required to be rigorous for no discernible reason, while other components were harshly penalized.
The course was very well taught!
Better teaching from Boaz and sections. I feel like it was hard to learn the material even though I listen to all the lectures and sections. It ends up being me doing the practice exams and reading alternative materials that helped me learn theory of computation. Everyone says the textbook is great but honestly I feel like there is just TOO much going on where even though big ideas are emphasized, it was hard to learn things methodologically. I wish the course could be better for investment to return ratio. It is so demanding and in the end I feel like with the hours I put in I should have gotten a lot more out of the class than I did.
Psets due next to midterms along with a bunch of new content from the week before made studying for the midterms pretty difficult. I felt much more prepared for the final because there was more time to study.
One big problem with the office hours is they are incredible incovenient. Literally 4 of the office hours offered clashed with times where I had completely unavoidable class commitments. They did not increase the office hour count or the timings.
The PSETs were kind of long, and I sometimes felt like I spent more time typing them up than solving and actually learning something from them.
Would be great if there were some opportunities to use coding within the class
I struggled a lot in this course, maybe a section dedicated solely for students who need extra help
Lack of nuance or breadth in later topics like encryption, ML, and quantum computing. Perhaps too much time spent on NP reductions.
The last three topics (cryptography, machine learning, quantum computing) got one or two lectures each and I don't think I understood them well. I do think they were good inclusions though. Perhaps prioritizing these or adding more section content for them would be good.
This course could be improved with a more thorough dive or anatomy of the SIZE, TIME, and success amplification problems, where I particularly struggled to find where certain values came from while reviewing solutions.

Comments
I thought the lectures and Perusall were the weakest points of the class. Boaz is an excellent author, but needs to be more focused in his lecturing style.
The last bit of the course is a bit rushed. It might be better to focus on just a few of the topics instead of glancing over a large number of topics.
I found that sections were mainly supplementary lectures, rather than a potential place for discussion. I feel that if sections acted more so like an office hour for the supplementary material would provide a more constructive time to work and learn.
Nothing
I feel the mandatory perusall / attendance component is a bit strict.
The grading, maybe. It could be a lot clearer, and the philosophy around homework not counting for much is actually quite stressful, since a lot of the extra credit is out of reach for a significant portion of the class since it's harder than the rest of the exams and students are under significant time pressure already.
More office hours.
Lectures could be more straightforward. They were a little rambly
Unreasonable participation expectations
The part of the course after the second midterm feels a bit too rushed. There is a single lecture on machine learning and no book chapter on it, for example.
Better textbook materials and more compact notes
I think the pace of the assignments was very quick. A bit more leniency during weeks with midterms would be appreciated.
N/A
Boaz could work on some basic public speaking (not saying emmm or ehhhh so much during lecture), and also the textbook is quite verbose.
I think the later components of this course were a mix of under/over–developed. Since topics like cryptography and quantum computing were broad overviews, I felt that it was actually significantly more difficult to understand what the textbook was saying (& am still pretty uncertain about these topics), so I feel like I didn't even understand the fundamentals of these topics. It might be better to focus in on the key components rather than relying on details to understand the broader picture.
The book is too wordy and I found it difficult to use. Lectures sometimes felt too quick (mainly when the professor explained the content) and sometimes too slow (when the professor answered questions or gave time for us to think about questions).
I think this course could be improved by structuring in more time for exam preparation. Making a problem set due the same week of the midterms is brutal, and really prevented me from having time to study and perform my best.
This course could be improved by explaining the more unintuitive ideas and examples of how to solve problems with them, such as cellular automata or lambda calculus.

Requirements Comments - What did this course require of you?

In your opinion, what preparation or background is necessary to take this course?

Comments
MATH 22ab for proofs was great/sufficient, STAT 110 was somewhat helpful but not necessary for the randomized algorithms section, math/proofs in general
Taking a proof–based match course such as MATH22 or CS20.
A decent bent for mathematical proofs.
discrete math, some experience with proof writing
Formal logic, probability theory
Not too much, comfort with mathematical proofs is ideal but probably not *necessary*
I think that an understanding of proofs is nice but not completely necessary because there are a lot of example proofs provided. I didn’t come in with any computer science background so I don’t necessarily think you need a super strong CS background to begin with to di well
It helps to have more experience in mathematics / proof courses, but I would hardly call that necessary.
Math 22a, CS 50/51, another class with some proofs or real math in it, Stat 110 (ish).
In my opinion there are no strict prerequisites for the course, but background in proof based math (Math 101, or anything above Math 25), familiarity with programming (CS 50, or even better CS 61), and STAT 110 are all very very helpful.

Comments
CS20, STAT110, CS50
CS20 is a must, and CS120 or a similar level of theoretical computer science content is encouraged.
Some kind of theoretical CS/discrete math is needed.
Just CS20.
CS50, MATH101 or CS20, and STAT110.
Familiarity with proofs and graphs, so CS20/Math 22a/etc.
It is important to have some background in proofs and basic probability theory.
CS 20 is helpful for proof experience. I also took CS 124 before I took this one, and while it wasn't necessary, it was certainly helpful.
proofs, 124 makes this class a lot easier
The textbook. The definition of expectation.
I would recommend having some proof background, at least at the level of AM 22, Math 22, or CS 20. I think having some experience programming in Python is also helpful.
familiarity with proof writing
cs20 and a lot of self–studying
Comfort with proof–based mathematics, some familiarity with programming
CS50 or CS 107, CS20
Experience with proofs highly recommended, as well as minor (not extensive) statistics knowledge.
CS50, basic mathematical maturity. I don't think CS20 or AM107 are really necessary.
CS 20 CS 120 Math 21B Math 1B Stat 110
There is some math background, but mostly just logical reasoning and some proof experience.
I found that having some discrete math background is mandatory for this course. Other courses may help with parts of the course, but for each part of this course, understanding how to prove and basic discrete math topics was necessary.
Basic understanding of mathematical proofs
Familiarity with proof writing and a bit of Stat 110
Some proof knowledge beforehand, knowledge of graph theory, but nothing else, really.
Need to take cs120
CS20
Nothing.
A background with writing proofs
I would definitely recommend having some background in proofs. Having programmed before could be helpful to understand the examples, but definitely prioritize math knowledge.
Strong CS background and a willingness to pour lots of time into the content
CS 20, Math 101, or a different proof based math course
Some basic programming experience in any language (you will write minimal code on paper, pseudocode works too)
Definitely a proofs class. Other than that, probably not much.
Some mathematical / proof background
Formal proofs (at the level of 22a when Dusty taught the course) but otherwise I don't think that any CS knowledge is necessary.
def proof writing (for me that was math 22a and then cs 124 – both were super helpful!) is the most important. other stuff is helpful but not necessary
Stat 110, formal math background
Good experience with proofs
Having a proof–based background while being mathematically oriented is very helpful to take this course.
Proof course

Recommendations Comments - Would you recommend this course?

What did you take away from your experience in this course? What did you learn? How did this course change you?

Comments
I learned a lot of theoretical CS which I did not expect to enjoy as much as I did. I had a great time and feel like the class taught me a very broad range of valuable topics and really improved my proof skills.
I learned a lot about Turing Machines, Circuits, and what it means for a function to be computable vs. uncomputable. The course made me appreciate theoretical computer science more deeply.
I learned that theoretical CS is probably not for me, but there's still some interesting problems in there that can make decent advances in the field of CS.
I learned a lot about general TCS and it's made me more inclined to pursue something like CS research over one of my terms.
I learned so much about how computers work, what it means to compute something, what computer science really is, what it means for a problem to be hard, and the barriers to our understanding the universe (namely the incompleteness theorem).
I'm a lot more familiar with a lot of the big concepts in theoretical computer science, such as computability, polynomial vs exponential problems, other complexity classes, and reductions.
This course taught me how to think more critically about questions surrounding computer science. I was challenged many times and not always succeeded, so another aspect of this course that I learned was to pick myself up when things got tough and push through to the finish line. I sure hope the staff saw that in me.
I had to articulate myself better and become a better problem solver.
I learned a ton from this course, since it was packed with content. It changed the way I looked at computation and computer science in general.
I learnt a lot about the foundations of theoretical computer science.
This class was philosophical and metaphysical at times, and Boaz's worldview / approach to seeing the world was very refreshing.
This was a great overview of TCS.
...
This course taught me a fair amount about computational complexity and current open questions in the field of TCS.
learned interesting theory behind computation and computational limits
I really didn't like the content of this course, though Im sure if you like theory, you would like it.
I learned that there are many equally powerful models of computation, that there are some problems we can't solve, and that there are inherently difficult problems. I will keep these in mind when programming.
When considering my takeaways from this course, I mainly experienced further practice with proofs and thoroughly considered the magnitude of solving questions as well as equating them others.
I really enjoyed learning about the frameworks of theoretical computer science. It made me realize the ideas behind the history and foundations of CS and gave me skills in reading mathematics.
I found that keeping up to date reading is a must. I fell behind and found it incredibly hard to catch up as the class progressed. My main takeaway is that for the beginning of a course, my main goals become finding other students I have a similar pset style to and keep myself honest with keeping up and understanding new material.
I gained an appreciation for theoretical computer science
It's actually insane that I've come away understanding everything in theory at the undergraduate level: the amount of stuff that Boaz manages to pack in one semester is wild and I really appreciate it about him.
I learned how to think more critically and logically.
This is one of the coolest classes I've taken
I learnt a lot about the nature and history of computation. It gave me a better understanding of the world around us.
Learned a bit about theoretical computer science
I understand P vs NP which is so cool.
I feel I learned a fair amount about a lot of topics in theoretical computer science, and I feel comfortable when engaging with this content on my own or with others outside of class.
I took away a greater understanding and appreciation for theoretical computer science.
I learned an insane amount about theoretical CS, something I have never dabbled in before. This class taught me to think in ways I have never thought before (thinking abstractly with computation concepts). It acted as a gateway to an elusive field I hope to be more involved with in the future, and this course definitely deepened my love for computer science.
I gained more logic in the way I solve CS problems.— starting with the definition, and some problem solving aspects.

What would you like to tell future students about this class? (Your response to this question may be published anonymously.)

Comments
I spent the beginning of this semester extremely conflicted between 120 and 121 and I'm really happy I ended up taking 121. Especially if you are planning to take 124, 120 overlaps quite a bit with 124 and the union of the knowledge from 121 and 124 should be significantly more. The class was not as difficult as I thought it would be –– as least under Boaz the ample bonus points (even an entire bonus pset and final average/median > 100) greatly reduced grade pressure. I felt I could focus on learning the material because of the grading structure. This course also covers a range of fascinating topics that are fundamental –– are there uncomputable functions? What if P = NP? With more circuit gates or time can we always compute more functions? Would recommend this course!
This course is pretty difficult, so be sure to allocate significant time to it each week. Problem sets are doable if you go to office hours routinely. More than watching lectures, I highly recommend reading the textbook chapters FULLY, since this will be more comprehensive to your learning and what you need to know to complete problem sets and do well on exams.
Take this class if you're interested in theoretical CS — it's rather well–taught and has good materials. Otherwise, in my opinion, there's more advances happening elsewhere in CS.
I think that if you do the problem sets yourself, you'll be able to get the most out of this course, both in terms of knowledge and final grade.
If you are currently deciding between 120 and 121, I think it's probably worth it to try both and then drop one later. One amazing thing about 121 (not completely sure about how 120 grading works) is that each pset has lots of extra credit as well as 10 pt reflection assignments that are essentially 10 pts of extra credit per pset. You can't get over 100% in the pset category (you are given a 100 regardless of whether you have 101 or 120 in any category, be it psets or midterms or the final), but this is a very good system to encourage tackling bonus questions and the entire pset while making you worry about small mistakes that would get you docked a few points here and there. The exams are also pretty chill: the medians for the midterms were ~75 for the first one and ~90 for the second one, and the median on the final was 101 (possible with lots of ~25 pts of extra credit in bonus question on each exam). Just keep up with the class and I think the overall experience grade–wise is pretty chill, it just takes some effort to keep up with all of the material because it does get quite confusing and difficult to understand at times. Boaz (who I believe will not be teaching next year, but I imagine this statement will hold true for whoever the next instructor is) is a great teacher and has a pretty good grasp on what students often find difficult, so he does a lot of practice and review during class, which helps a lot with comprehension.
In my opinion, this class was a badly executed introduction to theoretical computer science. The textbook is boring to read and seems to drag on, the material (especially the circuit units at the beginning as well as implementing Turing machines) is dull, and some of the proofs on the problem set take too long to get every single detail just right. Sometimes, there are tiny details that are technically necessary, but take quite a long time to think about and get perfectly, and I think the class would be much better if it could do away with those. However, as the computer science concentration does require three formal reasoning courses, I would think of this class as a necessary evil although it was easily my least favorite CS course thus far.
Comment more relevant to people who fall into one of the following buckets: don't consider themselves to have much CS experience / have a bit of a math background / are doing this class with simultaneous enrollment / are joining late This was the first CS class I took here. I think I learned a lot about the interesting problems that are being addressed in theoretical computer science right now. We spent the bulk of our time on complexity theory, though my favorite parts were the cryptography and quantum computing units at the end. I think if there's a lot of interest it might be worth extending these units a bit instead of cramming them into the last two weeks of class. As someone with more of a math background, this definitely made the class more manageable since a large portion of it is proofs. However, I don't think you necessarily need to have a big math background to do well at all; there are a lot of example proofs in lecture, the textbook, and in section that will introduce you to the kind of logic and thinking you need for the problems. I don't consider myself to be a strong "cs kid" at all as someone who started very late – so if you're in the same boat I don't think you should be worried. If you're doing this simultaneous enrollment, it is possible to do well without watching all the lectures and basically just reading the book and doing problem sets. However if you want to get a really deep understanding of the material I think it's good to block out time each week to dedicate to watching lectures, posting on Ed/Perusall because it's very easy for this to slide under the radar with all your other classes going on. There are high expectations for how you will engage in the course so make sure you show up to sections, office hours, 121.5 bonus lectures. If you're joining late, good luck lol…it was rough but it's doable and the TFs & staff are really awesome and supportive. Just try to join early enough to have time to study for the first midterm without being overwhelmed
Much of what I have to say about this course pertains to Professor Boaz Barak, so if you're taking this in 2024 fall you will have a different professor and it may not be applicable.

Comments
The material is really interesting and I recommend giving this course a shot. There's also at least 30 bonus points every PSET and exam, along with an entirely bonus PSET at the end + 10 more bonus points per PSET if you do a reflection, so grades aren't too significant of a worry and with a little time it should be a pretty easy A. Lectures are pretty optional but Boaz is hilarious and a little bit unhinged so I highly recommend attending. The material is important to know if you're gonna major in CS, and I think the class is definitely worth your time if so.
This is a great but challenging class. I was heavily considering taking 120 because I was scared only math gods could do well in 121, but this is totally not the case. I had relatively little background in math/proofs coming in (just math 22a, didn't take cs20), but I think if you're interested in the material and you like theory / proving stuff (again, even if you don't have much experience in it), you can totally do well regardless of background. Plus there's tons of extra credit so the grading is actually way more chill than it might seem. And the material is FASCINATING. This is one of those courses where you will leave seeing the world —– and especially computers —– differently. You will understand what it means to compute something, what time complexity is, and what it means for a problem to be hard in an entirely new way. BUT the course is still difficult and very time–consuming. Boaz's lectures are solid, but they cover relatively little, meaning you unfortunately have to spend a lot of time (2–4hrs per week) slogging through the textbook. The upside is that I found the problem sets pretty fun even if they were hard (and the TFs are really supportive).
I'm a pure math concentrator who took this class essentially as an elective. Overall it's a good class —– the material is very interesting and Boaz Barak is an engaged professor and an effective lecturer. However, there were also many issues with class structure that made the class less enjoyable. I don't regret taking the class, but I'm not sure I would take it again given that there are so many other nice classes. CS 121 will give you a solid understanding of the major concepts in theoretical computer science, and you get to learn about lots of cool ideas along the way such as Godel's Incompleteness Theorem and P vs NP. Boaz is a good lecturer, and he really cares about the course and his students. There are many bonus points both in the problem sets and the exams, meaning that grade–wise it's not too stressful of the class. The textbook is a good resource to have, but it also isn't amazingly written, and the online platform Perusall used for reading is annoying. The problem sets are mostly manageable, except for a few random hard problems that were thrown in every now and then that you have to get from office hours or pset groups. Very annoyingly, grading was pretty inconsistent between CAs, and the grade you got on a pset depended heavily on which CA was grading it —– however, many people complained about this so hopefully by next year they will have fixed this issue.
There are a lot of opportunities to earn bonus points and you don't have to worry about the PSet component. Just focus on the exams the most. If you are hesitant about taking the course, I was in your shoes once. I read a lot of negative Q Guide reviews and tried my best to avoid the course. However, I have no regrets now. The bonus points mean I don't have to worry about the weekly psets much. Boaz is also a fantastic professor, very kind, caring, friendly, and approachable. That alone is reason enough to take the course.
This class is challenging, though important. I took this course as a requirement for my CS concentration, after having taken CS20 and CS120 in the same sphere. Despite this prep, I found this class challenging. However, I found a good support network within my pset buddies and regular attendance at office hours. The homework assignments were therefore quite manageable, so long as you start them early and work through confusion with staff and pset buddies. The exams are very difficult, but with proper studying can be done well. Overall, this class is a tough nut to crack and will take a fair bit of time on your part (I dedicated 15–16 hrs/week outside of lecture on it). However, don't let this discourage you from taking it, especially if you need to take it to fulfill some requirements! There are opportunities to make up missed points on the psets via reflections, the last pset is fully bonus points, and all psets have additional bonus questions for you to attempt for extra credit. Despite not having adequate tutoring support for this class (ARC or otherwise), I think I learned something in this class and was able to make my way through. You can too!
Take this instead of CS120. I shopped CS120 and then switched immediately to CS121. CS121 is more interesting and better taught imo. Boaz is so nice, funny, and cool. I had never heard about most of the content covered in the course before CS121. It is true that students may be more cracked than the average in CS120 but that makes for a better course experience. The psets are not difficult as long as you look at them early and use OH. The amount of time isn't any different than that for CS120 so why not take the course with most interesting content? Highly recommend!
This course is great: 2 midterms and a final make it a lot of studying, but they're not too difficult and the grading is generous. Lots of necessary content for any true computer scientist, so taking this course is a no–brainer.
This class covers a lot of interesting material that forms the foundation for theoretical computer science!
This class is so much work, I don't know why CS department expects students to put in so much work going to lecture and section and doing all the psets and even expectations of showing engagement like commenting on readings and on active on Ed... I think in an ideal world where CS121 is the only course I am taking this is reasonable, but with everything else going on when one is busy, these just become counterproductive and induce a lot of unncessary stress and in the end not helpful to one's learning.

Comments
Boaz is nice but I wish he was a better lecturer. When Madhu came to lecture once I felt like the way he lectured made so much more sense, mainly bc it was organized and he knew how to first give motivating examples.
This is a pretty great class and you get a broad overview of a bunch of different fields in theoretical computer science! Follow lecture and readings are helpful and the problem sets shouldn't be too bad.
Boaz is a fantastic teacher and cares so much for the students. I didn't really like this class until after the first midterm, where it becomes a lot easier and more intuitive. Don't worry too much about your grades because the bonus points and reflections will take care of that mostly. 124 makes this class a lot easier, but I also think that 121 would make 124 a lot easier.
Buckle up as you are about to read the best review on this page. Oh yeah, I'm back baby! But firstly, I am pleased to inform you all that GTA 6 is coming in 2025! At least its not coming to PC before I graduate and I can finish college before I acquire such a crippling addiction. Lets get going. For those of you know don't know, I keep strict track of time spent on the class. Then I share that with you all! The first time entry begins from September 4 and each entry is one week's worth of time starting from the monday of the week. Here are my results: (Format is hour:minutes:secs. or hour:min) By Categories: CS 121 Ed Board Question 0:10:00 CS 121 Final Exam 3:00:00 CS 121 Lecture 22:00:42 CS 121 Math Review 0:18:20 CS 121 Midterm Exam 3:30:00 CS 121 Midterm Review 6:12:34 CS 121 Office Hour 14:30:04 CS 121 Participation Grade Form 0:31:00 CS 121 Problem Set 60:53:30 CS 121 Reading 13:58:25 CS 121 Section 8:42:15 By weeks: 4–Sep 5:06:00 11–Sep 9:03:31 18–Sep 15:38:11 25–Sep 14:23:56 2–Oct 12:52:30 9–Oct 16:03:46 16–Oct 1:25:35 23–Oct 10:14:48 30–Oct 14:40:41 6–Nov 9:44:20 13–Nov 0:00:00 20–Nov 6:44:38 27–Nov 12:47:18 4–Dec 0:37:45 11–Dec 4:23:51 Total Hours: 133:46:50 I stopped going to lecture after the first week and followed all the classes online. I have a firefox extension that lets me speed up Panopto to arbitrary speed. Years of listening podcasts are paying off as I pretty much listened to all lectures at 3.5x speed. Sorry Boaz but too much ehmm, umms, and uhhhs in the lectures. I may or may not have missed some sections too. No comment. I need my participation grade. Context and Background: I took 4 classes. CS 121, CS 61, Econ 1010A, and Stat 110. Sophomore. I feel like CS61, CS121, and Stat 110 are the actual intro to CS classes and more of a taste of actual CS concentration than CS50. Relevant classes taken: CS124, Math 22A/B (proofs). Almost

Comments
no extracurricular clubs and activities. Having taken 124 before, this class was a breeze in comparison. Don't get me wrong, my time data suggests that I was putting in the hours. But, I was not mentally as drained as 124 lol. Review: The readings are by far the best part of this class. This is because the professor has put in a lot of time to really create a reading material that covers everything needed in class. If you have any interest in ToC, then this is of course for you. You should really be reading the textbook over the semester. I started out very diligent and did all the readings before class and took away a lot from the class because of it. However, as your own experience also probably shows, keeping up with readings is a task for giants and no mere mortal Harvard student can do such schenanigans. Trust me, read it over the summer. Thank me on sidechat later. Don't half ass it and read the first two chapters like me! Some really interesting concepts we covered were Uncomputability, Efficient Computation, Reductions, and Complexity Classes of different problems. The course is speaks to a fundamental level of our world. As in the concepts that we cover are not about CS specifically, it is about what is possible in the world in the first place. The Church–Turing Thesis is a great example of how we can think about the capabilities of systems not just computers. I would of course suggest taking some proof based class before this. Know how to prove things. In terms of assignments, there are 11 total with one of them being pset0 which counts for half credit. There are so many extra credit points in this class. If you put in the time, there is no way you don't get a perfect pset grade. Beware that some weeks the pset will be uncharacteristically hard. I'd say showing equivalence to a Turing machine or a Deterministic Finite Automata are some of the harder problems. Go to OH of course. One big problem with the office hours is they are incredible incovenient. Literally 4 of the office hours offered clashed with times where I had completely unavoidable class commitments. They did not increase the office hour count or the timings. So, get started early. Review the questions for an hour, go to OH and get hints and disccuss approaches with students and you'll be fine. 2 midterms. Both have a lot of extra credit (around 25 extra credit). Because of that, it is also relatively easy to get a 100 average in that category. Know your reductions well! Uncomputability and Polynomial time Reductions feature HEAVILY in class. First midterm is hardest. Second one is a joke and will get you the points you lost in the first one if you do well. (Extra credit points transfer if you have an exam above 100 and one that is below.) Final is also a mix of midterm 1 and 2. Just prep your cheatsheet and you're fine. Idk if the reason the course was easy is because of me having taken 124. So your milage may vary. Take my advice but apply to your context. Midterm 1 results: 79.4 mean, 80.5 Median, 17.03 STD, Max was 116.5 (bruh). Midterm 2 Results: Mean 89.27, Median: 89.5, STD 20.06, Max: 125(achievable). Final Exam Results: 101 median, 98.65 mean, 18.03 STD, 127 MAX (128 was total available points). Ultimately, this is a class you'll take if you are doing CS anyways. I think the ABUNDANT extra credit opportunities and the Cookies make this class far better than 120 and I would suggest really reconsidering taking CS120 and instead recommend taking 121. Quick note on those who have already taken 124: Forget about efficient algorithm design. Note that until you arrive at the time–complexity topics, the only thing relevant about an algorithm is whether it exists in the first place. Do NOT try to find efficient algorithms as in 124. A python program that literally goes through all the computation space in exponential time is ok if showing the existance of an algorithm. You should brush up on your knowledge of MaxCut, ISET, Vertex Cover as they are very common NP–Comp problems. Learn how to reduce a problem from any of these into any other. Given a 3SAT solver, can you make a ISET solver? etc. You won't need Linear Programming concepts or anything about efficient algorithms for NP–complete problems for this class. Finally, for the love of all that you love, get a local Latex Compiler so you don't have to use Overleaf and worry about harvard wifi. I recommend the Latex–Workshop extension from VSCode. May you stand unshaken in your endevours and remember that Love is a Long, Long Road.
This course is probably exactly how the Q guides make it out to be. I'd recommend going to office hours frequently, since the TFs can be super helpful and help you avoid spending hours going down the wrong rabbit hole. The textbook is also super helpful; it was written by Boaz himself, so obviously it aligns really well with the course materials.
This class was very difficult and made me reconsider my concentration in computer science. Don't take it as a senior.
I enjoyed Boaz's teaching of this course; while not perfect, he was good at pacing himself through the content, and is a true whiz. The material was surprisingly interesting and not too challenging, and the exams were very fair, with leniency given with bonus points up for grabs on every assignment. However, I do think the balance of the course could've been shifted, with less time given to NP reductions (something covered in 124) and more given to the miscellaneous topics at the end of the class, like ML theory and quantum computing; as it was this year, the discussion of these latter topics was cursory and not very informative.

Comments
As with feedback for previous offerings of this course, beware of self–selection (i.e. lots of people taking this course could've taken CS120, but chose not to, probably because they like/want to explore/are better at theoretical stuff). The class generally moves quickly and problem sets can be hefty. Also, I'm a senior who took 124 long ago, so my viewpoint might not be representative.
This is one of the class is fairly foundational to understanding what computing is and why it works. I highly recommend it.
Challenging yet excellent course with wide opportunities for constructive growth and further rewarded study.
Take this class if you really want to develop stronger math and theoretical CS skills! You really DO NOT need to worry about your grade in this course (average on final was 100%), so I would treat it like a learning opportunity rather than a concentration requirement. That being said, I would really just read the textbook and go to section if I could do this class again. You'll learn more in the textbook than in lectures, and the section leaders were all amazing.
This course is a lot of fun, even as a non–cs concentrator.
Barak was a great professor in very many different ways and I thoroughly enjoyed the course this fall. I am sad to see that he will not be teaching in fall 2024 but I am sure that the course will remain top–notch, especially since the teaching staff is traditionally wonderful. Many CS majors will probably be disinclined to take this course because they are drawn to the option of CS 120, but I wholeheartedly recommend CS 121 because it is extremely interesting, instructive, and accessible despite what some might say.
This course has a lot of interesting concepts and I would recommend! However, do not underestimate this class and potentially even try to find more supplementary material, which I recommend Sipser's Theory to Computation.
This is a great class. The material is very interesting and it is taught well.
I took this class for the CS concentration requirements. The content covered in the class is pretty interesting, despite being super theoretical. The topics connect well to each other, and the material builds on itself nicely. Boaz grades generously and if you've taken CS 124 that definitely makes the class easier. Sometimes the psets are a bit tedious, but generally not super difficult.
This class is like a visit to the doctor's office. The doctor is lovely, they're clearly skilled at what they do and they can get their point across to you very well. It is also healthy that you go, because you cannot go further in life without the insight the doctor's visit will give you. However, your visit may sting a bit. (Also Boaz won't be teaching this next year so take the following with a grain of salt) TL;DR: This is a great class, but the grading is a bit annoying, the textbook isn't the best, and Boaz doesn't show up to his own final, which feels somehow symbolic. Let me explain. Boaz is great: he speaks a bit slowly, but his lecture style is very strong, clear, and gets at the important bits of the material while cutting out as much of the faff as possible. The material is also wonderful: it really is bedrock–level stuff for understanding CS and the class really gives you an overview of the entire field of CS theory, so while you may not be able to read current research right after it, you at least know where to start and can recognize the ideas and link them back to what you know. There is also a great deal of pop–CS that you learn here: P vs NP, Turing Machines, Uncomputability, Godel's Theorem, as well as trendy stuff like ML theory and quantum computing. The homework is fine: it's a significant amount of work but it's not killer (I think 10–12h a week is fair), plus there are reflections available on the psets to regain some credit, and usually some bonus on top of that (PSET 3 is the hardest, hang in there!). The flip side: everyone gets 100% on the pset component of the class, meaning your grade is entirely determined by the exams. So, how are these exams? They're not *too* bad: all the material this year I feel was quite fair, but you should definitely read the textbook, since some questions were lifted straight from solved textbook examples. The final was also very very fair and there was plenty of practice material (plus PSET 10's solutions were quite useful). Ok, so how is the textbook? Not great. It's very verbose at times, there are mistakes/typos here and there (which increase exponentially in the second half of the course) which make it hard to read and a bit boring, so I recommend watching the relevant lecture first to discriminate what the important bits of information from a particular chapter are, to then deepen those with the textbook, not the other way around. Boaz also has his own definitions of things, separate from the standard, which are debatable; they are definitely more intuitive and modern, but I don't know if proofs over them are as simple as they would be with Sipser's definitions, for instance. The section component is great, and a lot of my understanding came from the CAs, who were all excellent. Attendance being required and Perusall comments / Ed questions being tracked are a bit annoying, but they're not a great hassle.
I would tell them that this is a great class but you need to have taken 120 or some other theory course beforehand or else it is easy to get lost.

Comments
Background: took CS 124 previously. Was personally undecided between taking this and CS120, and ended up taking this class because of the argument that I wouldn't have learned much in CS 120 and would have learned more instead. Probably true by design? But anyways this course did in fact end up being quite a bit of work. Really the quintessential undergrad class – weekly psets that take 6–10 hours – but I did learn quite a bit from them. In hindsight I probably would have chosen CS121 again just for that extra intellectual pursuit.
If you're down to work a little bit, take this class. I'm a pretty bad student and struggled through most of my CS career here, but I absolutely loved this course. It's probably the best taught hard CS class I've taken.
This class is very heavy in the first few weeks. I time–tracked my hours spent (including lecture) and there's a very clear downward trend in work difficulty. Weeks 1–3: 16 hours on average Weeks 4–6: 14.7 hours Weeks 7–9: 8.75 hours Weeks 10–12: 7.75 hours Weeks 13–15: 4.3 hours So two pieces of advice: 1) get a head start on the textbook so you're not dying in the first week (I spent 22 hours the first week doing pset 0 and textbook readings I should have done over the summer), especially if you're shopping other classes so you're not spread to thin. 2) If you find yourself deciding between this class and CS 120, and find the material interesting but too time–consuming, I'd recommend sticking it out to week 4 at least. The class gets so much chiller, I don't think the first few weeks are a good benchmark for how much time you'll spend on average throughout the semester.
CS 121 is a great class and one of the best classes I have taken at Harvard. The course is very organized and Boaz is a great professor who cares about his students. I honestly don't think it is a hard class, work load is maneagable and the first half of the class is very different from the second half, with the second half being very easy.
I forgot the backward direction on an IFF proof on the final and now may get an A– or B+ for it... tough grading. Be careful. Lots of 100's with EC.
Doing the readings thoroughly (and before class) will help you immensely with problem sets and concept understanding. It is quite the time commitment, but you learn about many cool topics. I took 124 before 121 and if I could go back I would reverse the order. Formalizing reductions and knowing how to prove soundness and completeness would have helped a lot with the former.
I think this class covers a very broad selection of topics, but connects them nicely.
This class is so fascinating, and a great introduction to the theories of computer science that will be helpful for future research, future math courses, etc. I hadn’t taken CS 124 prior to this course. Exams are fairly difficult and require deep understanding of the concepts to be successful. If you are debating between CS 120 and CS 121, I would not rule out CS 121 right away; if you are interested in more theoretical concepts or studying math, I think CS 121 may be more helpful. I think algorithms and the content from CS 120 can be more easily self–studied. Also, Prof Boaz is an amazing person and definitely cares a lot about his students. We loved the Insomnia cookies after each exam :)
Extremely good class. Textbook is good and interesting. Boaz wrote it himself with passion and arranged it in such a way that the entire course flows smoothly and led to one another. Boaz teaches in a flipped classroom format which works extremely well, and the lectures serve a good review of the material. It is therefore essential to read the textbook before the lecture (or you can simply watch it later after reading). Problem sets are very doable and pedagogical. It is fun to work through and very helpful in helping you to be familiar with the material. Grading is perfect! A lot of opportunities of extra credit in both the problem sets and exams. If you are interested in the material and you put in the work (which is not that much anyway, definitely less than 15 hours a week), you will get a good grade. Boaz and his fantastic teaching team also managed to make it possible for everyone to engage with them too! The textbook is hosted on Perusall that allows comments and discussions (fantastic tool that I hope every class uses!), the Q&A site Ed is extremely helpful as usual (make sure you utilize it! There are a bunch of smart CS students who are willing to help out) and OH is also good. Perfect 10/10 class, both in material, execution and student satisfaction. He is leaving though, so check who's teaching next.
Interesting class. The lectures are fun, though you can get by just doing the readings + psets if needed. I don't regret taking this over CS 120 – I thought 121 would be very time–consuming, but it was maybe 8 hours/week of work. (I say this as someone who's taken lots of theoretical CS classes, including 124.)

Comments
I think this course is a good introduction into theoretical computer science, and the course staff do a great job of structuring the class and preparing relevant materials to support the content. The class does have a relatively hands–on engagement / attendance policy, including weekly psets, lecture and section attendance, and commenting on textbook readings and Ed. You don't have to do all of those so it isn't a large amount of additional work , but it may be something to keep in mind as there are definitely more regular tasks for this course than other ones.
The course material is largely fun. The exams are manageable if you can spend a couple of days preparing for them. Boaz is a good lecturer but sometimes I found him slightly hard to follow. That could totally have been because I don't have a super strong math background. Boaz's textbook for the course is very helpful but also it is evident that it is still a work in progress. That sometimes makes doing the assigned readings a pretty time–taking process
Force yourself to do the psets on your own before getting help.
i LOVED this course. i've struggled a lot with cs courses here and this was one of the first courses where i could sit with a problem set and pick at it and actually understand what was happening. maybe that's just me choosing my cs courses poorly, but i think it really goes more towards the fabulous teaching staff (boaz is wonderful and the office hours are super great – they actually are helpful!) and the great lectures and readings. could not recommend more.
The class itself was amazing. It was informative and interesting, and in general I thought the problem sets were engaging/helpful. Though I came in not thinking that I'd have any interest in theoretical cs & unsure if I was even "smart enough" to take this class, I thoroughly enjoyed it and also now understand why people find theoretical cs cool. This class is a bit self–selecting, and I will admit that at times I struggled with the types of people that may take this class (office hours/section could feel unwelcoming due to some of the students' attitudes/backgrounds, though the course staff was incredibly kind & understanding). But this was my only complaint, since I didn't have many friends in the class. (In general, note that it didn't seem like many women took this class my year, maybe <=30%. Hopefully this changes in future years.) Take this class (with friends, under Boaz) even if you're unsure that you're qualified, because I genuinely think this is one of the best CS classes I've taken at Harvard. A little bit of stat 110 (ok if concurrently taking) & proofs background on the level of math 22a/cs20 should be more than enough. It's also excellent preparation for CS 124 (imo much better than CS 120 in preparing you for algorithmic problem solving, if this is why you're considering this class). Be prepared to put in a bit of work to read the textbook and think about the problem sets. Exams are more than fair (with lots of bonus points available), and Boaz ensures the homeworks are doable/non–stressful (again with a lot of bonuses). I've heard a couple of complaints about Boaz's teaching style and I don't really get why... I think his slides are clear, he gives really good examples (he even shows how theory concepts can be converted to Python code) and it's refreshing to see how excited he gets to teach these concepts. Take this class!!!!
I recommend this class. Yes, there were times when it felt confusing and frustrating, but eventually it all came along. Then, it is quite time–consuming (weekly difficult psets, two midterms, and a final) but this time is needed to really engage with the course content. Pros: – Interesting content (though confusing at times) – Nice lectures (though sometimes they felt too fast, and sometimes too slow) – Wonderful sections with section materials (extremely helpful for understanding and reviewing the current week's content) – Interesting homework (though often time–consuming and difficult) – Lots of opportunities for extra credit – Useful and quick feedback – Lots of preparation material for the midterms and the final. Cons: – Very time–consuming – Often difficult and confusing (though there are a lot of resources available) – The book is too wordy and difficult to use (every chapter is about 30 pages and you cover two chapters per week) Overall, I feel I learned a fair amount about a lot of topics in theoretical computer science, and I feel comfortable when engaging with this content on my own or with others outside of class.
This is a really good course, and I would highly recommend taking it if you are interested in theoretical computer science in the slightest. Boaz is a great lecturer, and the course is structured well. It offers lots of interesting, useful background on things like computability, quantum computing, cryptography, machine learning, probabilistic computation, P vs NP, etc.
This class is challenging yet amazing. I highly recommended taking this course, as it would help kind of "retrain" your mind to think about algorithms and computation more abstractly. I come from a math–heavy and proof–based background, and this definitely helped me power through some of the more abstract stuff.
In the first half of the course, the book is very essential to read, which can take up a lot of time since there are many symbols you must parse through. However, if you are able to understand the book during the first half of the course, with good intuition, you will be set up for success mostly for psets. Then, during the second half, the book is not too important, and most of the material covered

Comments
in psets can be learned through lecture. This course is a good high leveloverview of circuits and computability.
Boaz is absolutely amazing, and the concepts are really useful and interesting. Coming in as a very systems person focused on very practical things, with no proof background (not even cs 20), I will say that this class was absolutely doable and enjoyable. The class concepts are hard, but what was tested was very easy, and the PSETs don't require you to go to office hours or spend a lot of time on thinking about things. I will say that sometimes it felt *too easy* as in the grade inflation might have been favored in disadvantage of learning. You get what you want from this class, but you can truly learn a lot. Don't get discouraged by the first third of the class, as it gets much easier.

Instructor Comments

Please comment on this person's teaching. (Your response to this question may be published anonymously.)

Comments
Boaz is a wonderful professor and incredible lecturer. I had to watch lectures online due to simultaneous enrollment, but even from Panopto lecture recordings I was thoroughly engaged in the lectures and felt his enthusiasm for the subject matter.
Boaz is a very smart guy. Sometimes he seems to not really answer people's questions well, but overall he knows what he's talking about when it's on–topic, and I'm pretty sure he wrote this textbook which is quite good.
Very engaging and personable!
Boaz is an excellent teacher. He's passionate about the material, gives interesting, funny, and helpful lectures, and provides opportunities for students to learn more about what they choose to study.
Boaz is an ok lecturer but great teacher –– he cares deeply about the success and understanding of his students, and that goes a long way.
Boaz Barak is very good at giving lectures, and he cares a lot about his students!
The accessible score was only fair. By your own admission, you have a very busy schedule.
Boaz knows a lot about what he teaches, no question about it! He is also very receptive to questions and feedback. However, his style of lectures can be improved. He has great lecture slides with amazing transitions, but perhaps he assumes we know more than we do, so he glazes through a lot of the material and jumps straight into the difficult components of the class. When he stops to ask questions from the class, when the class is quiet, it's not because we don't have any questions but precisely because we have SO many that we stall in silence. I would have appreciated a more gentle introduction to some of the more challenging or foundational concepts of the class. For example, PRGs, bounds, and BPP/BQP!
Boaz is a great professor. I enjoyed his lectures. His passion for the topic comes out in every lecture. He cares about his students and that was very important to me. He is also funny (not like Eddie tho :p)
Professor Barak is an exciting lecturer. He loves the content and heeds questions well. Sometimes, he gets wrapped up in a proof and mumbles for a bit and everyone gets lost, but he makes up for it with fun slides and fun content. Overall, thoroughly enjoyed most lectures.
I can't complain overall—Boaz is a very likeable lecturer with a sense of humor, but also you can tell he's very enthusiastic about the subject matter, and I think the lectures went at about the right pace.
His lectures were very engaging and he answered all of the students' questions in a comprehensive manner.
Boaz is not a great instructor, and I think there is a lot of improvement for the way he delivers his teaching.
Boaz was absolutely fantastic! He's clearly passionate about the subject material, and it's super fun speaking with him about various subjects within the field.
Boaz explains the content clearly and lectures are a great help in understanding complicated subject matter.
Enjoyable lectures with great efforts to make it interactive and engaging.
Boaz writes beautifully in his textbook, but I found his lecturing style to be a bit too rambling and unfocused.
Barak was a great professor in very many different ways and I thoroughly enjoyed the course this fall. I am sad to see that he will not be teaching in fall 2024 but I am sure that the course will remain top–notch, especially since the teaching staff is traditionally wonderful. Many CS majors will probably be disinclined to take this course because they are drawn to the option of CS 120, but I wholeheartedly recommend CS 121 because it is extremely interesting, instructive, and accessible despite what some might say.
Barak has a very good understanding of the material and provides reasonable to good motivation on why these topics matter.
Boaz is a great teacher
Great! Boaz is a wonderful lecturer and very enthusiastic and approachable.
Great teacher and lecturer!

Comments
He is not the best lecturer but he is really knowledgeable on the topic. I really benefited from asking questions in lecture.
Emmmmm Ehhhhhh Boaz could be a better lecturer but I love him!
boaz is the best!
Boaz really took his time to explain concepts both in lecture and on Ed whenever students were confused on any topic. I think his enthusiasm for the subject was a primary reason that I engaged so deeply with the content.
I liked this style of teaching. Presentations were very informative, though sometimes lectures felt too quick (mainly when the professor explained the content) and sometimes too slow (when the professor answered questions or gave time for us to think about questions).
Boaz was an excellent teacher. I appreciated how carefully thought out and planned all of his lectures were. You could tell he really cared about the course because the slideshows were very good. They had great visuals for understanding concepts, links to supplemental material, and interactive questions to test the knowledge of the audience. He is very engaging overall.
Professor Barak injects a lot of humor into his lectures, which I really appreciate lol. He explains the concepts so thoroughly, and he'll often stick around after class if you go up to the front and ask him questions about the material.