

ERIK DEMAINE: I think in the very first lecture of class we gave out a little survey. Partly see what people's background was, because this is sort of a new class. We didn't know exactly how to position it or where people would be coming from. So how much complexity theory they knew and how much algorithms they knew.

And it more or less aligned with our intent, which was-- people knew some algorithms and some of them knew complexity theory. Some of them didn't, and that's what the class is aimed for. You don't need to know a lot of algorithms to take this class, but definitely familiarity with that kind of concept is super helpful.

And then the other part we were curious about because we were still designing the class-- and we designed it throughout the semester that I was teaching it-- was what topics were more interesting than others. And so we had some topics that I was sure I wanted to cover, because I knew them really well. And I thought I had a lot say about them, and they're important.

And then there were a bunch of other topics where I was going to have to learn the materials so I wanted to some idea of which ones I should learn more of. I think the general response was everything sounds interesting. But there were a few that stood out.

One of them for example, was algorithmic game theory. And a class called PPAD where luckily my friend Costas Daskalakis-- another professor in computer science here-- came in and gave two guest lectures. Really great, and they showed everyone how this field worked. And I learned a lot, and we've done research based on that since.

So there were a few topics that stood out as particularly interesting to the students. And then one thing I was particularly curious about was the use of fun examples. I was worried that students would not take the material seriously if I only used fun examples.

But the feedback that I got was that a lot of people said, I want to see games and puzzles. That's like among the list of topics everyone said, yeah, more games and puzzles. That's really fun. They saw Mario in the first lecture. They're like, yeah that was cool. It was exciting.

So I took that as permission to use a lot more fun examples. And still there are some more serious algorithmic problems. But I think you can get all the same-- you can understand all the same material and learn all the same things through fun examples, and just enjoy it more.

So the students-- I was very happy-- were aligned with that view, and I didn't have to worry about students also seeing tons of serious examples in addition to the fun ones.

I used the survey to really get to know the students. And see where they're coming from, and help aim the class in a direction that they would get the most out of it.