

The CS-Ed Podcast

Kristin Stephens-Martinez

The CS-Ed podcast is a new podcast featuring 6 episodes hosted by Kristin Stephens-Martinez from Duke University. Each episode is a conversation with a computer science educator talking about teaching and managing their classrooms.

Our goal is to release one episode a month. The first episode came out in November and featured David Malan on CS50's tools. December's episode featured Dan Garcia on designing exams. Episodes in 2020 will include Amy Ko on teaching students how to debug, Mark Guzdial on live coding in class, Armando Fox on MOOCs in the classroom, and Colleen Lewis on how she teaches her class and uses peer instruction.

The Numbers

As of 12/18/19

Episodes Released	2
Episodes in Production	4
Twitter Followers	152
Facebook Page Followers	167

Episode	Released	Episode listens
Teaser	11/15	101
Episode 1: David Malan	11/23	74
Episode 2: Dan Garcia	12/13	12

Podcast Format

The podcast format focuses on a main topic between the guest and the host. After the main conversation, we ask each guest about something they find awesome in computer science. We then close with Too Long Didn't Listen (TL; DL), where our guest summarizes the most important things they want our listeners to learn from our conversation. So, if all else fails, our listeners can skip to the end to get the gist.

Summary of Episodes

David Malan

In this episode, we talk with David Malan from Harvard University, Professor of the Practice of Computer Science in the School of Engineering and Applied Sciences. He teaches Computer Science 50, Harvard University's largest course.

Our conversation focused on CS50 tools. An overview of the tools is in a YouTube video David provided. We spent most of our time talking about help50 and style50. Help50 is a tool that, when fed error output, returns a suggestion or question a student should focus on to help

interpret the error output. Style50 is a tool to help students fix the style of their code by highlighting what to change. However, David emphasized that he wanted the tool to require the student to change the code themselves.

When asked about something awesome in CS he'd like to share, David talked about containerization, especially tools like Docker. In CS50, they use containers on both the server and client-side. He finds they are a great way to package up everything for students.

His Too Long, Didn't Listen (TL; DL) focused on encouraging fellow teachers to see if someone else has already created an educational tool that would fit their needs rather than reinventing the wheel.

Dan Garcia

In this episode, we talk with Dan Garcia, a teaching professor at UC Berkeley in the EECS Department. He was selected as an ACM Distinguished Educator in 2012 and ACM Distinguished Speaker in 2019. He has won all four of his department's computer science teaching awards.

Our conversation focused on designing exams, which he boiled down to his five-finger rule: (1) material coverage, (2) reasonable time, (3) range of difficulty, (4) variety of question types, and (5) ease of grading.

His "something awesome in computer science" highlighted his mentors Mike Clancy and Brian Harvey, who are both emeritus teaching professors at UC Berkeley. Mike taught him about having a variety of question types on his exams. Brian taught Dan his philosophy about grades and grading in general.

Dan's Too Long; Didn't Listen (TL; DL) summarized this five-finger rule into an excellent short sound bite.

Amy Ko

In this episode, we talk with Amy Ko, an Associate Professor at the University of Washington Information School. She directs the Code & Cognition Lab and studies human aspects of programming.

Our conversation focused on how to teach students to debug, a skill many of us undoubtedly struggle to get our students to do effectively. Amy suggests step 1 is to have students articulate what is happening versus what should happen (current output versus correct output). Step 2 is brainstorm different ways (hypotheses) that might be causing the discrepancy and exploring each idea to see if it is the cause. If a student runs out of ideas before they find the bug, go back to step 1 and confirm they understand what should and should not be happening.

When asked to share something awesome in computer science, Amy talked about her interest in computer science history and Donald Knuth. Knuth is one of the originators of many core algorithms in computer science. He also spent 10 years cataloging every mistake he made while working on the typesetting programming language LaTeX. So, his interests were broad, and he also wrote bugs!

In Amy's Too Long; Didn't Listen (TL; DL), she emphasized that debugging is a primary skill and is something we should teach. And we are starting to find ways to teach this skill.

Mark Guzdial

In this episode, we talk with Mark Guzdial, Professor of Electrical Engineering and Computer Science, with a courtesy appointment in the School of Information at the University of Michigan.

Our conversation focused on live coding, which is programming in front of the class as the students would program. Mark emphasized that the most essential part of live coding is modeling process. The second part is modeling how to manage mistakes. And the third is to create opportunities for students to make predictions.

For this "something awesome in computer science," Mark talked about how he loved that computer science can be anything else. That computer science can look like and behave like any other discipline.

Mark's Too Long; Didn't Listen (TL; DL) broadened our original conversation by pointing out that, while live coding is useful, it's one method among many and not necessarily the most important one. Other teaching methods he thought were also important included peer instruction, contextualized computing education, and generally, to start with a problem.

Armando Fox

In this episode, we talk with Armando Fox, Professor of Computer Science and Faculty Advisor to the MOOCLab at UC Berkeley. With David Patterson, he co-designed and co-taught Berkeley's first Massive Open Online Course (MOOC) on "Engineering Software as a Service," offered through edx.org.

Our conversation touched many topics involving MOOCs. We discussed the history of MOOCs, how he got into it, creating Small Private Online Courses (SPOCs), how MOOCs call into question established teaching habits, some experiments he's planning that break those habits, and how to get student buy-in when using a MOOC in the classroom.

When asked about something awesome in computer science, Armando talked about his love for the history of computing. One thing he observed is how much ideas get recycled in computer science. He even has a web page he calls "Master geek theater" of his recommended documentaries ranging from five minutes to three hours.

Armando's Too Long; Didn't Listen (TL; DL) focused on MOOCs' long-term legacy. He does not think they will replace instructors. Instead, they will enable instructors to use their time more creatively because they have well-curated, interactive, battle-tested exercises available to them. Moreover, they will help us think about how to get the non-deep content experts involved in helping the students or their peers in learning the material.

Colleen Lewis

In this episode, we talk with Colleen Lewis, Associate Professor of Computer Science at Harvey Mudd College. She specializes in computer science education and diversity issues, as well as is the creator of <http://csteachingtips.org/>, which we at the CS-Ed Podcast post about often.

This conversation was a question and answer with Colleen. Our topics included: peer instruction, how she structures her lecture and class, how becoming a better and better teacher is a marathon, cheating on assignments, the pros and cons of splitting students based on prior experience, and where to hold office hours.

Colleen's "something awesome in computer science" was another podcast, Modern Figures Podcast. It highlights the work of black women in computing. The audience is geared towards teenage girls interested in computer science.

Colleen's Too Long; Didn't Listen (TL; DL) was two tips. First, was survey your students and respond to that feedback. The second focused on how your teaching practices should allow for opportunities to see into student thinking.

Appendix A: Links

Website	https://sites.duke.edu/csedpodcast/
Facebook	https://www.facebook.com/csedpodcast
Twitter	https://twitter.com/csedpodcast
Warpwire (podcast hosting)	https://warpwire.duke.edu/w/HRgDAA/
Apple Podcast	https://podcasts.apple.com/us/podcast/the-cs-ed-podcast/id1486344352
Spotify	https://open.spotify.com/show/5NtCNeqOxgtyh0FTb5OLf2
Stitcher	https://www.stitcher.com/podcast/the-csed-podcast
Overcast	https://overcast.fm/itunes1486344352/the-cs-ed-podcast
Google Play	https://play.google.com/music/listen?u=0#/ps/larzcyl2vs6m7so6onolggxa3xy
TuneIn/Alexa	https://tunein.com/podcasts/Education-Podcasts/The-CS-Ed-Podcast-p1271699/