

## C Spire Let's Talk Tech Podcast #7 transcript – Cracking the code on computer science in the classroom

**Host:** Dave Miller

**Guests:** Dr. Jeff Gray, University of Alabama computer science professor  
Kagan Coughlin, co-founder of non-profit Base Camp Coding Academy

**Preview: Dr. Jeff Gray:** “There’s a difference between computer literacy vs. computer science. In most high schools across the U.S. today, the computer lab is really just a bunch of glorified typewriters where students are learning how to type and use office products. It’s slowly changing now where many educators are beginning to understand the value of computation, computational thinking, and what role computer science has and what computer science even means.”

**Introduction: Stacey Kirkland:** Hello, and welcome to *Let's Talk Tech*, a monthly podcast that explores the latest emerging technologies, the people behind them, and how these trends will affect the way we work, live, and play. I’m Stacey Kirkland of C Spire, and in today’s episode, show host Dave Miller interviews Dr. Jeff Gray, a University of Alabama computer science professor and one of the nation’s leading experts on the front lines of efforts to integrate computer science in the classroom and Mississippi entrepreneur Kagan Coughlin, co-founder of the non-profit Base Camp Coding Academy, which provides minority youth with a fast-paced, year-long vocational training program in computer programming. Join them as they explore how these specialized classroom and vocational training programs promise to help alleviate the shortage of skilled programmers in the U.S. workforce and enable students to develop more critical thinking and problem-solving skills.

**Dave Miller:** Welcome to C Spire's *Let's Talk Tech* podcast. I’m Dave Miller, and today we are discussing the growing demand for computer programmers in business and industry and the need to boost computer science instruction in America’s classrooms to address the shortage. Joining us via phone to talk about this subject is Dr. Jeff Gray, a University of Alabama computer science professor and one of the nation’s leading experts on computer science instruction in the classroom. Also on the program today is Kagan Coughlin, a businessman who heads a non-profit organization in a small, rural Mississippi town that is launching an innovative, vocational technology training program for minority high school students to help meet the computer programming needs of local and regional employers. Welcome to the program, gentlemen.

**Jeff Gray:** Hi Dave, thanks for having us.

**Kagan Coughlin:** Thank you, Dave. Good to be here.

**Dave Miller:** We’re pleased both of you could join us on today’s program, as we learn more about the need to integrate computer science instruction in America’s classrooms and community-based efforts to address the historic, underrepresentation of minorities in the computer programming field. We’ll start with Dr. Gray first. You serve on Code.org’s Education Advisory Council and you’ve been offering K-12 outreach programs, including summer camps and contests, science-fair mentoring and teacher-professional development for over a decade now. Since 2011, you’ve been a National Pilot Instructor with the College Board for the upcoming Computer Science Principles course. You’ve received the National Science Foundation Career Award and were the 2008 Carnegie Foundation Professor of the Year. Can you give our listeners a brief assessment of where you think American public education is today in embracing the value and importance of computer science and how that is translating to changes in how and what is taught in the classroom?

-more-

**Jeff Gray:** Certainly. We've seen more growth in the past three years in awareness about the importance of computer science than in the last 30 years. There's widespread interest in computing. It's actually been difficult to keep up with pace of all the exciting things that are happening in this space. The big change has been in the general understanding that there's a difference between computer literacy vs. computer science. As you know, computer literacy is about learning in general how applications work like Microsoft Word and other Microsoft Office products. In most high schools across the U.S. today, the computer lab is really just a bunch of glorified typewriters where students are learning how to type and use Office products. It's slowly changing now where many educators are understanding the value of computation, computational thinking, and what role computer science has and what computer science even means. For instance, in Arkansas, the governor actually has a mandate to teach computer science in every high school. There's been a lot of interest in New York City, Chicago and in Broward County, Florida to bring computer science to virtually all of those schools. The National Science Foundation is working with a dozen or so groups through NSF grants to bring more computer science into the classroom. In Alabama, we are teaching 47 high school teachers across the state on computer science principles. The exciting thing about our work in those schools is that 49 percent of the students are either young women or underrepresented minorities whereas in a traditional AP class, those percentages are in the single-digits or down in the teens.

Google and Gallup collaborated on a poll last fall about what's going on in other parts of the country. You can explore those results at [csedu.gallup.com](http://csedu.gallup.com). Those results were really an eye-opener to many of us. One finding is that 90 percent of parents and students want true computer science taught in their classrooms, but only eight percent of administrators believe that there is sufficient parent interest. There's clearly a gap between what school administrators think and what our parents and students want. In Mississippi, only five students in the entire state took the Computer Science AP exam on the College Board and none of them were women or African-American. The good news is that those five who took the test did really well, averaging over a four on the score. There's also a new program that Mississippi State University is working on at [csforms.org](http://csforms.org) and Ole Miss is considering some extensive outreach efforts. So some sectors of higher education in your state are really trying to help improve the situation.

**Dave Miller:** Thanks, it sounds like we are beginning to make some significant progress, particularly in the last three years as you've indicated. I want to switch gears for a minute and talk with Kagan Coughlin who's the co-founder and trustee for an innovative program that's getting underway right here in Mississippi. The non-profit Base Camp Coding Academy plans to provide minority youth with a fast-paced, year-long vocational training program in computer programming. Once they graduate, the students will be ready for entry-level IT jobs with local and regional companies. Kagan, can you explain how this program developed and where you hope to take it in the future?

**Kagan Coughlin:** I'd be happy to. This really began as a conversation in the early spring of 2015 with Glen Evans, the president of FNC in Oxford, Mississippi. At the time, he was lamenting all the challenges that they have as a software company headquartered in Mississippi when they are looking for new talent. My interest at this point—with two young children in Mississippi public schools—was with the opportunities for Mississippi youth going through the public school system. A portion of that population has the opportunity to go to some of our very best schools—and we have a very diverse and broad community college system, but there is a large segment of the population that does not have any opportunity like that, to pursue computer science or any other higher-learning endeavor.

### Page 3 - C Spire Let's Talk Tech podcast episode #7 transcript

Glen is a former work colleague of mine, and we've kept in close contact. The conversation quickly developed into: we could complain about this, or, we could do something about it. Base Camp Coding Academy is what we've developed over the last 10 months or so. Our first challenge, as with any educational endeavor, was figuring out how to fund the program. We've also spent a good amount of time meeting with all of the business and philanthropic leaders across Mississippi considering ways to finance what is really an economic development initiative that invests in the future resources all of our businesses will need to be successful. We've had amazing success. I'd like to especially thank the C Spire Foundation, which was one of our first founding sponsors. The Foundation fully understands our goal of investing in the youth across Mississippi who have the aptitude and the heart to become future programmers and who will be able to contribute to the growth of the individual companies across the state and whatever they choose to do with the rest of their lives.

The execution of a fast-paced, rigorous, vocational-training program has three legs to it. One is the financing, which we have been very successful. That's entirely because of the generosity of companies across Mississippi who are helping finance this initiative and also the interest from our teaching community. One of the challenges in educating any young person is finding that person and finding them at the critical time in their life when they are both driven to succeed and have the time and the energy and the support mechanism behind them to make sure that they succeed. So we have been meeting with teaching professionals at the administration level across the high schools within about an hour of our location here in Water Valley, Mississippi, and also with those folks who are in the classroom with high school students every day. They are on the front lines in helping us identify those young students who have that drive to succeed and no alternate channel to go and pursue that.

What we are striving to accomplish with Base Camp is a home for the young, smart, driven, Mississippi youth who today does not really have a path to walk. When they leave high school, they have some opportunities, if they have either those folks in their lives that can show them the scholarship route, if they can find some opportunity through financial aid, but often times the channels that accept students like that—it's hard to get into a small Mississippi community and identify the students and give them that encouragement and show them where they could go.

Our goal is both to identify those youth with the assistance of our teachers across Mississippi, and then offer them a 100 percent scholarship to our one-year program, which will culminate in many of the institutions, the companies, across Mississippi that are funding Base Camp Coding Academy as a charity operation will be their future homes for the beginning of their career. This is work-force training taken right down to the street-level, where we will find, with the assistance of anyone who is out there listening, and all of our teachers across Mississippi—those students who, if given this opportunity, will excel. And once we have them, in 12 months, we will take them from essentially zero experience in software programming to the ability to walk through the doors of some excellent Mississippi companies like C Spire, FNC, Nissan and Toyota with the skills and abilities to immediately contribute to their success. Our mission includes both the educational component as well as mentoring these young professionals when they leave us through the first several years of their career. We expect to meet with them, buy them lunch bi-monthly, check on how their employment is going and hopefully bring them back to function as mentors for a subsequent class at Base Camp.

-more-

## Page 4 - C Spire Let's Talk Tech podcast episode #7 transcript

**Dave Miller:** That's great information. I appreciate the background on your program. You obviously have a lot of passion for what you're doing. This next question—I really want both of you to answer—is about finding that balance between what Dr. Gray talked about earlier in promoting computer literacy and its benefits in the classroom with real-world opportunities like the Base Camp Coding Academy.

**Jeff Gray:** My personal goal - and this is also the objective of many of the national efforts - is not to convert every student to computer science. What we are really trying to do is raise awareness. That's the first level of need in many places. Most high schools, when I walk in and talk to students, they cannot even give me a definition of software. So we're trying to just let those students realize that software is all around them in their daily life, whether it's the phone they wear, to some folks who wear medical devices or pacemakers, there's software there. The cars they drive. Just raising their awareness. The second area is the idea of computational thinking itself - helping in the logical, analytical, problem-solving space. So what we've seen with some of these efforts like the Hour of Code, where millions of students have had a chance to learn programming in a very fun way, it's about problem-solving. So it's little puzzles that students get engaged in. It's fun and engaging. There's Star Wars there. There's Elsa and Ana and Angry Birds. So being able to draw and engage the students, getting them hooked on something that's actually all about problem-solving. Another thing that we try to do, aside from teaching that software is useful in many domains, is to promote this idea of the growth-mindset. This is a very popular term in education circles. It's all about helping students develop the grit or persistence they need to stick to a task until it is finished. The puzzles we see in many of these K-12 efforts are engaging, and the students keep wanting to work with those. So those kind of things have benefits beyond even coding. While it's very useful to make sure that the students get a deep understanding, in many places we're just trying to get the awareness out about what computing really is.

**Dave Miller:** What about the Base Camp Coding Academy? Obviously there's some practical elements of what you're trying to do through a very fast-paced, year-long curriculum that you're trying to get these motivated kids out the door with an opportunity to land a job with a company that is in sore need of their skills. How do you balance what Dr. Gray is talking about, developing this ancillary benefit of personal resolve and gumption along with the analytical skills to solve problems with the need to complete an accelerated program?

**Kagan Coughlin:** Actually, I love the question, because it brings to light a conversation that I have over and over again with folks that still perceive coding or software programming as someone sitting in a dark room, looking at a black and green screen and writing some secret language that will impact some major machine somewhere across the world. The world has changed drastically in the last 20 years, so that everything we touch—and Dr. Gray gave a great rundown—we are hard-pressed to find a component of our world that has not accelerated in the technology field, to a point where there is so much information available, there are so many mechanisms we have, that we have these amazing opportunities to streamline with a piece of software, taking lots of information. And really all programming is, is having the ability to think logically to a future scenario where a human being is not present and laying out a path for decisions where, if it's a website, when someone arrives, there's software there determining what elements are presented at what opportune time, so that that person experiences whatever they are looking to experience when they went to that address. All the way down to traction control in a vehicle that a piece of software needs to determine which wheel turns when, depending on slippage rate. To talk about programming as a finite, narrow bandwidth for learning is an interesting but, I think, no longer a relevant way of speaking about programming. The other side about learning, more globally, is that this is very much a problem-solving, logical training of any young person

or adult. Often times I like to observe a thriving chess club in our local public school here in Water Valley. You can just tell how their young brains are working. It's exercising not just one component of their brain, but it's getting them trained to think and problem-solve, and they can take that and apply that anywhere. So in our short bandwidth, we're training how to write software code, how to problem-solve in a very specific field, depending on where our graduates will end up working, but it is also equipping them to learn for the rest of their lives, in whatever field they go and specialize in.

**Dave Miller:** That's a very thoughtful and profound answer. I'd like both of you to respond to this next question. There's been some good news for the computer science field in recent months. Last fall, Congress added provisions in the new education bill that includes computer science in the definition of well-rounded education subjects in the Every Student Succeeds Act, which effectively puts it on equal footing with subjects like math and science when local and state policymakers figure out how to dole out federal education funds. But here's the thing that I want both of you to try to address: sometimes we tend to have this idea that, "Well, it's law now and we've solved the problem," but making it the law and developing policies and curriculum to make computer science an every-day reality in the classroom, it's really two different things. How optimistic are you that real progress can be made to help students become more well-rounded learners and erase the growing shortage of qualified candidates for the now estimated 1.4 million computer-programming job openings that we're predicting U.S. companies will need in just four years? We'll start with Dr. Gray.

**Jeff Gray:** Well, I'm very optimistic on this for several reasons. The new federal law also helps those of us who are trying to provide, for example, professional development and work with curriculum access to more grants. The problem in the past is that many of the science and math areas dominated the funding levels, so we had less access to be able to go out and help those who have a desire to learn more, particularly training teachers and so on. So there's going to be more opportunities for those who want to help at the federal level. Other things I would suggest is don't rely just on the federal level. When you look at Arkansas and Alabama, they worked closely with the state Department of Education to gain their support and participation. In Alabama, our superintendent of Public Education visits schools and does these Hour of Code puzzles with students. We were able to get this new course to be able to count as a math elective in Alabama. So Alabama one of the 18 states now that have math counting for this new CS Principles course, so those are the kind of things that send a message out across the state, not just the federal level, but the state. Hey, there's opportunity here. Then, of course, the issue of curriculum, a lot of folks are working to address that area. In just the past three months, a group from Code.org, the Computer Science Teachers Association and the Association for Computer Machinery began what's called the New K-12 CS framework, so this is an alignment of computer science principles across all different grade levels, what should be taught there, how it might even be integrated with other existing federal standards, whether it's common core or some other standard that may be met, so those kind of things of pulling computer science out of its own discipline, as well as showing how it can integrate with the other disciplines, is a goal that a lot of people are looking at now. There's a lot of recent movements and efforts, trying to pursue that.

**Dave Miller:** That's great. Kagan, what about from your perspective?

**Kagan Coughlin:** I have to say listening to Dr. Gray, I'm more optimistic now. From a different direction, I've spent a great deal of time in our local high schools and as I speak with teachers and explain what we're starting here at Base Camp and asking for their assistance in identifying those prospective students, and what I've seen is that with all the challenges that we have, especially with the changing curriculum and the testing requirements, and the testing calendar in most of our public schools, that to

have a public policy come in and adopt something that will fix our computer science short-comings is a long road. And I'm optimistic that we will see improvement over time. In fact, what I'm seeing in the classroom is that, even outside of having computer science as something that is on the required class list for our current public schools, the way that they are teaching all of their subjects is becoming more and more technology-focused. I have two young children that are in first and second grade, and they use more software programs in their daily classroom learning for reading and math and science than when I was in high school 15 years ago. So there are programs like i-Ready and even the testing mechanisms that we use in Mississippi for state and federal funding, all of that requires the students to spend time in front of a computer, learning some basic uses, which will make our job at Base Camp and the adoption of any computer science curriculum much easier for the student body that's already enrolled today.

**Dave Miller:** Thank you. Dr. Gray, you're in the forefront of efforts to more closely integrate computer science instruction in the classroom. In your view, how much progress is being made in developing age-appropriate computer science curriculum so that children learn critical thinking and refine transferrable skills like sequencing along with other logical problem-solving and algorithmic thinking concepts that they can apply to other subjects like math and English and music theory? Is it possible to get kids across a broad spectrum to literally think about thinking? – a process that is referred to as metacognition, that has proven benefits in a lot of different areas, including self-monitoring and independent learning? What's your perspective on this?

**Jeff Gray:** So thinking about the idea of what kind of age-appropriate tools or curriculums available, there's a large amount of new tools that are being developed. For example, young children using ScratchJr, there's no words, it's all about symbols, so students can learn to program using arrow buttons and things like that. In the code.org K-5 curriculum, I'm facilitating some instruction on that. The CS Fundamentals course is particularly for kindergarten through 5<sup>th</sup> grade. We've had a chance to train over 700 teachers, and by the way, we've had some Mississippi teachers also come over to that training. So that's also age-appropriate. There's three courses in that sequence, some for earlier non-readers. We also use the idea of Computer Science Unplugged, where we're not always needing to be in a computer lab, in front of a computer. We're teaching ideas of computation and thinking and problem-solving without a computer, in a fun way that's almost like kinesthetic learning, things are up and moving around the classroom. It's a lot of fun.

Regarding the topic of other subjects—math, English, and music theory. One of the things we are looking at in particular with the new College Board AP exam is to really highlight the idea of creativity. In fact, creativity is one of the so-called “seven big ideas” of this new course. We have one school in Birmingham that is a school of fine arts, and they have a math and science division, but all of their art students take this Computer Science Principles course. It's really exciting because those students are creating artifacts in computing but using their really deep creative skills that they have in art and music. So those kind of connections are really strong and exciting to explore. We look at creativity and also the global impact of computing in addition to talking about algorithms and programming and other types of things. So the whole idea of independent learning, some of these things can be done online, and I think, in general, online

education is a challenge for many students. But the way that some of the efforts of code.org and some of these others that have online tools, even like Khan Academy, they have these online tools and quizzes and different environments that you can use, those are really helping students progress and do these kinds of activities in an independent way. And being able to think about computing in a way that it might apply to their daily life, and they can begin the self-monitoring and understanding how computation involves them in their daily life.

**Dave Miller:** I appreciate those insights, Dr. Gray. Thanks for that. I'd like both of you to respond to my last question, which gets at the heart of whether these efforts that we've been discussing can ultimately be successful. Some parents are concerned that focusing primarily on students in our efforts to gain more exposure to computer science in the classroom could overlook the other half of the equation: teachers. As both of you know, many teachers lack a basic understanding of technology and have even less background in programming, which could potentially create barriers and hurdles for more timely and widespread adoption and implementation of updated curriculum. Are there any strategies designed to overcome these potential impediments? We'll start with Mr. Coughlin, and I'd like you to tell our audience how you're addressing this knowledge gap and the selection of instructors for the Base Camp Coding Academy program.

**Kagan Coughlin:** That also is a great question. Dr. Gray mentioned the Gallup Poll results earlier, which really points out the gap between our education professionals today and their understanding of the need for computer science on the curriculum, vs. what the parents and students feel is necessary for real progress. And something about computer science, this field is changing very quickly. While many of the foundational languages were developed in the late 80s and early 90s, their application and the things that they can do on mobile phones, on apps, on gaming systems these days, they are accelerating so quickly into what is possible that, at Base Camp, we could hire the best in their field as our teacher in the classroom, and if they went into a box for six months, they would be out of date. And so the continuing education component for computer science is so critical, and that will apply to our own teachers as well as to all the efforts around adding to the teacher certification across the nation a computer science component. This is a field where the core foundation is learning how to learn, continuing your own education. Our graduates will enter the work force ready to continue learning. That is our goal. There is no endgame, check this box, and now you're good to go for a foreseeable amount of time.

**Dave Miller:** Thanks, I really appreciate your answer. Dr. Gray, what about from your perspective? I know you've been involved in a lot of curriculum development and teacher training, so I'd be interested in your view as well.

**Jeff Gray:** Teacher professional development is the million-dollar question, quite literally, or the multi-million dollar question. The challenge is that most of the teachers that were asked to teach computer science have very little background or no background at all. So the idea is that we generally would bring teachers up to speed, like high school teachers, would be in the College Board one week APSI, just simply won't work. We have to have training that's much deeper. We do our training on a National Science Foundation grant over two years. So there's one year of training and we come back and revisit topics. We have math teachers and also career-tech teachers, and they go through curriculum and training processes over a 120 hours each year. Jan Cuny, program director for computing education at the National Science Foundation, coined the term of the CS 10K movement, so her goal was to train

## Page 8 - C Spire Let's Talk Tech podcast episode #7 transcript

10,000 across the U.S. and to explore ways in which that might be done. There's no quick solution. You cannot train a teacher. It's the equivalent to bringing an English teacher in high school into a way to teach AP Calculus in one week. That's just impossible, and it takes some time and some patience with the teachers, their understanding to working and giving them all the resources.

There are a lot of efforts out there, the National Science Foundation efforts for training, as we mentioned, we're just ending our project in Alabama after three years. There's project Lead the Way, which is engineering education in K-12. They have a computer science pathway. The code.org activities and efforts are available. There are many other groups that are training—the Beauty and Joy of Computing out of UC Berkeley had some free training as well for teachers that go on for a long time. This is probably actually the heart of the problem—bringing out the way to equip our teachers to be effective educators in an area they've never really seen before and the long-term commitment we have to training them. It can't be done in one week. It will be doomed to failure, and if we do it that way, actually it'll be probably worse than not doing anything, because what will happen is the students in those classes will become disengaged and not like the class. The teachers will not be prepared. We really have to take our time and be patient, but it's probably the most critical issue to me, personally, in this whole space.

**Dave Miller:** Those are excellent insights and I really appreciate your comments, Dr. Gray. I wish we had more time to explore this topic, but alas we need to move on. As we've learned today, integrating computer science instruction and computer literacy in the 21st Century classroom is crucial to our success. I appreciate the time both of you have spent with us on the program. If you'd like to learn more about K-12 computer science education in your state, go to [www.code.org](http://www.code.org). To learn more about Dr. Gray's computer science training workshops, go to <http://gray.cs.ua.edu>. And you can follow the progress of the Base Camp Coding Academy at [www.basecampcodingacademy.org](http://www.basecampcodingacademy.org). Thanks again, gentlemen, for coming on today's program, and we'll look forward to catching up with you soon.

**Jeff Gray:** Thanks. I had a great time.

**Kagan Coughlin:** Thank you very much.

**Episode #8 Preview - Stacey Kirkland:** Thanks for listening to today's podcast. You can follow Jeff Gray on Twitter @jgrayatua. To learn more about Gray's computer science workshops, go to <http://gray.cs.ua.edu>. You can follow the progress of the Base Camp Coding Academy at [www.basecampcodingacademy.org](http://www.basecampcodingacademy.org). To find out how you can support K-12 computer science education in your state, go to [www.code.org](http://www.code.org). If you like the show, subscribe through Soundcloud, iTunes, Stitcher, or TuneIn. Join us next time as we talk with the founder and past president of the American Telemedicine Association on how technology is revolutionizing the delivery of healthcare across the U.S. Learn about a Mississippi success story—the University of Mississippi Medical Center's Telehealth Program— and how it has quickly become a national leader in the use of connected care, resulting in lower costs, better patient access to specialists, and saved lives. Discover why the state, not generally known as a leader in healthcare, is one of only seven nationwide to receive an A rating from the ATA for its progressive telehealth policies and programs.

###