SPHERO WHITEPAPER SPRING 2020

7 MISCONCEPTIONS ABOUT INTRODUCING COMPUTER SCIENCE TO YOUR CLASSROOM

sphero®

Introduction

Many common misconceptions are floating around out there about introducing computer science into classrooms. Teachers, administrators and parents alike can have concerns about being already stretched thin with limited resources in the classroom, finding the time to devote to a new curriculum, age appropriateness of the subject matter and more. Even students have misconceptions about it.

In this whitepaper, we'll dispel some of the most common myths. The goal is to give you a better understanding of why, perhaps now more than ever before, it's not only a good idea, but critical to teach your students the basics of computer science in and beyond in the classroom. (Hint: Their futures may depend on it.)

Misconception: I don't have training in computer science or the confidence to teach it in the classroom.

Reality: Teachers don't have to be experts. In fact, they will never be as prepared or ready as their digital native students when it comes to technology. Teachers can learn along with their students.

We get it. It can be intimidating to teach a subject you aren't completely familiar with. But the reality is, K-12 teachers do it all the time. High school or college professors who teach one subject and one subject only? Yes, they need to be the experts in that subject. But in many K-12 schools, especially in the lower grades when students are in one classroom all day, teachers are dealing with every subject. Nobody is an expert in it all.

What good teachers do when they're not completely familiar with the subject matter is study the curriculum. With computer science, and STEM, it's no different. **Many computer science classroom programs, like the ones we offer at Sphero, include complete curricula for teachers to follow.**

Many teachers even go day by day, studying up for the next day's projects. After all, this is about discovery. If you're excited about the process of discovery, rolling up your sleeves and getting into the projects with your students, they'll be excited, too.

The students don't need to be perfect, and neither do you.

> ***It starts with educators! We must become excited about learning and shifting our mindset that computer science is not just for the stereotypical 'nerds' but for everyone, and especially women. Anyone can learn if we shift the mindset and come together to crash the digital divide.***

Kimberly Lane Clark

Sphero Hero & Director of Blended Learning, Computer Science/Ed Tech Consultant, Lancaster ISD.

Misconception: It takes up too much time during my already-full classroom day.

Reality: Computer science curriculum can be easily integrated into other subjects you're already teaching, especially math and science, and other core classes.

And it's not all numbers and code. The very foundation of computer science is problem solving. That's one of the most important things to focus on, and it can be the most fun for your students. Learning about problem solving can enhance their performance in other subjects, too.

It's about giving students room to explore theories, experiment with testing those theories, and yes, even failing and going back to the drawing board. That's a powerful way to learn and it can be applied to many subjects, some of which you may not expect, like English and Social Studies.

Misconception: My students are too young or too old for computer science.

Reality: Computer science is appropriate for any age group.

Elementary-age kids are snapping littleBits together (a hands-on learning system that consists of electronic building blocks). In the lower grades, children are using them both at home and in the classroom, giving them the basics and building blocks of circuits and showing them the types of fun things they can build when they have those basics down.

It's highly beneficial to start kids in the lower grades with simple computer science projects because it turns a subject that can be intimidating into something fun and creative. That gives students a solid foundation and positive mindset for when they get into the upper grades, where they can start writing more complex code to program more sophisticated robots.

There's another benefit to starting students earlier rather than later in computer science.

The earlier they are exposed to it, the more likely they'll be to develop an interest in it. In lower grades, kids are figuring out what subjects and concepts they're good at, what they like, and what they don't particularly enjoy. **By putting computer science in those first two categories, you're increasing the likelihood that they'll carry that interest and curiosity with them into the higher grades.**

So, there is something for every age group, and starting early is a great boon to your younger students.

Misconception: Computer science means sitting behind a computer, and my students are already getting too much screen time.

Reality: Computer science means computational thinking and problem solving. It takes kids away from the computer screen and into collaborative, hands-on projects.

It's about working as a team to apply problem-solving techniques, and teaches so much more than just the facts and figures on the page. Those are needed parts of the curriculum, to be sure, but getting students out from behind their computer screens and into the classroom can bring the concepts and information on the page to life in exciting, new ways.

Misconception: I need a classroom of “A” students to introduce computer science.

Reality: Nothing could be further from the truth! You don’t need a roomful of budding Bill Gates prodigies to teach computer science. In fact, it’s great for the students who may not be at the top of your grading curve because it opens their minds to a new way of thinking and problem solving, using creativity and teamwork to get the job done.

And remember, failure is part of the trial and error process here. It’s crucial to learning to test theories that don’t always work the way they should. In computer science, failure is not a catastrophe like it might be on a history or French test.

Instead, failure is a tool that can and does lead to success in the end. It allows students to see what they missed, question their methods, and come up with new theories and strategies.

That’s a powerful lesson to teach a student who has had less-than-stellar academic performance. Failure on tests, essays and other assignments can seem like the end for students, who may give up trying as a result. But with the mindset that failure is an opportunity to focus on what you missed, simply part of the process of getting to success, it can change a student’s perception the next time he or she misses the mark in another subject.

Misconception: It’s too expensive. Budgets are tight everywhere and I don’t teach in an affluent school district. We should just focus on the basics.

Reality: The basics like reading and math are an important foundation. But truly educating students today means preparing them for the future of work. We don’t know what that future will look like, but we do know that a solid foundation in technology will be necessary for students of today to compete in the workplace of tomorrow.

It’s just as important for kids in lower-income neighborhoods, perhaps even more so. With plenty of funding resources and grants available for computer science programs, it’s about achieving equitable access. Sometimes teachers believe that if they are not in affluent schools, they can’t advocate for computer science in their schools. But, often the opportunities are not presented to teachers because some students in low socioeconomic areas spend more time on remediation.

The foundations of computer science — solving problems by breaking them down into simple steps, thinking critically about possible solutions and outcomes, creating theories and testing those theories — will be necessary skills and core competencies as today's students become tomorrow's job seekers.

And we're not just talking about white-collar, Silicon Valley jobs. As the rapid advancement of technology continues to disrupt industries, it's clear that many positions, from computer science to manufacturing and everything in between, will require at least basic knowledge of computing principles, critical thinking and problem solving.

All of that said, the buck stops somewhere. Download our free guide: Guide to STEM Funding [link] for ideas and resources to fund computer science in your classroom.

Misconception: I can't get buy-in from administrators, so why even try?

Reality: Administrators like to see teachers do innovative, new things, but might need a little convincing to open the school's pocketbook. Here are a few ideas to try:

1. **Find an online community** of teachers who are already doing it. If nobody at your school is teaching robotics, say, head to the internet. You will certainly find a group of like-minded teachers. Ask for advice, lesson plans and descriptions of what's working.

2. **Do your research** into the benefits of computer science education.

3. **Show your administrators what you're doing.** Invite administrators and principals into your classroom to give them a firsthand look at your students' excitement, how teamwork is enhancing the project, their use of critical thinking and so on.

Conclusion

Learning the truth behind common misconceptions about bringing computer science into the classroom is the first step to making it a reality.

Bringing computer science into the classroom will give your students benefits that extend well beyond the science curriculum. It teaches them the power of teamwork, turns the concept of failure on its head, fosters creativity and outside-the-box thinking, teaches creative problem solving and encourages hands-on projects that get them away from their computer screens.

The earlier you introduce these concepts to students, the better. They can use these skills in all of their other subjects, enhancing their entire academic experience. Further, as our world goes increasingly digital and our process becomes increasingly automated, computer science will be a must-have skill when your students enter the job market. We don't know exactly what the future of work will look like, but we're certain it will include computer science skills in all levels of employment, not just white-collar tech jobs.

By teaching your students computer science, you'll be preparing them for further education, and for life beyond the classroom. Isn't that what it's all about?

At Sphero, we're dedicated to bringing educators the tools they need to not only teach computer science in the classroom but also to secure buy-in from administrators and even parents. Contact us today to hear more about it.