

Advice from School Leaders: Preparing for Computer Science in Your School

QUESTION What advice about computer science do other school leaders have for me?

TOOL DESCRIPTION When beginning a computer science initiative in a school, learning from others' experiences can save time and resources. Nobody knows better how to support a school leader starting a program than school leaders who have already taken those initial steps. This tool captures that valuable advice.

WHY DO YOU NEED TO KNOW THIS? Enacting change in schools and classrooms is always a challenge. Learning from school leaders who have "been there" can help make introduction of a computer science education program easier.

HOW DOES THIS TOOL HELP? This tool includes advice directly from high school principals who know, first hand, about the practical and logistical steps needed to start and maintain a computer science program.

TOOL CONTENT

Some of the advice below might seem like common sense; but many leaders know that explicit attention to some of these basic steps may prevent them from falling through the cracks of the busy and sometimes haphazard school leader's day. The advice below comes from interviews with school leaders who have decided to bring computer science to their schools, and who want to share their experiences with others. All are of equal importance, but might be prioritized based on your contexts and conditions. The first three, while shared in the context of computer science, could be applicable to any new program or practice. The remaining four focus specifically on computer science education issues.

1. Commit or Don't Do it

"...you have to decide that you're going to make it mean something."

The decision to bring computer science education to your students needs to be one driven by a commitment to see it through. An "if we can afford it" attitude will not work if it halts progress mid-stream. As one principal said, "...it's an investment, it's an investment in the infrastructure of the building. It's an investment in the technology itself. It's an investment in the teacher and overall it's an investment in the students."

2. Prepare Your Budget Early

"...how much and when are the big questions so that we can start preparing those budgets as soon as possible..."

Early budget planning and preparation is essential. This is true for all efforts, but computer science education in particular often has ongoing or additional expenses other subjects don't have related to equipment and software updates. When planning, consider both the near- and long- term costs.

3. Develop and Nurture Community Support

"The whole idea is the neighborhood—to get the entire buy-in of the neighborhood where the school is in. At some point you are going to need help in funding, whether that's through grants or local businesses here or there. Getting that buy in ahead of time so that if you do have to go ask for funding, they'll be at the table and know ahead of time that the request doesn't just come out from nowhere."

Creating partnerships and taking initiative to expose your program to business leaders and local officials can add value to your program educationally and financially, enhancing the experience for your students. Additionally, getting higher education institutions involved locally or regionally can help provide options for making a computer science education class happen. The **Partnering with Industry People** and **Partnering with University Faculty** tools can help. One principal explains "If you've got a city college that's close by that's offering it and you can't do it, maybe you can partner with the university to offer an introductory class... [or] ... you may have someone in the community that has a background in that says I [would] love to be a guest teacher and maybe teach ten weeks."

4. Dedicate Time and Space

“...before you start the program make sure you have space for it. It's not a classroom; it's a lab. The computer science classroom should be just as important in design as the science classroom.”

A computer lab should not be prioritized as an ordinary classroom space or the place where tests are administered. The priority of the computer lab, if a school is fortunate enough to have one, should be computer science teaching and learning. While multi-purposing may be possible, one principal makes her priorities very clear: “Either it’s going to be used as a computer lab or it’s going to be used as a classroom... it’s very difficult to do dual-purpose in a computer lab.” As another principal explains, “What’s going to happen [when kids are pulled] out of computer science classes so somebody else can go into the lab to take a test? What are you going to do with the kids in that time?”

5. Get the Correct Technology in Place

“...on a logistical level and monetarily, you have to think about the equipment you need, you need to think about the software you need. You need to think about the layouts of classrooms that are more effective than another.”

Concerns with the technological aspect of operating a computer science education program are many and varied. One principal summed it up: “...the biggest thing is the space and making sure that you’re hardwired and that you have a lot of wireless in the building...” See the **An Introduction to Bandwidth in Schools** for more information.

6. Identify a Teacher and Make Sure They Have the Correct Credential for Your State

“The first thing that a principal would have to do is [ask yourself] how do I staff this? That’s the biggest thing. Is it physically possible for me to have a person in the building able to do this?”

Knowing in advance who wants to teach, who needs to be credentialed, and what they need to do to be credentialed is essential. Ideally, you already have a likely candidate. Then, as one principal suggests, “if you have a teacher that you think could be successful in the program, but they don’t have the right certificates or the right credentials, [ask yourself] do you have an avenue that you can get that person certified and caught up before the students are there on day one, before you even really offer it to the students.”

Determining your state's teaching requirements for computer science can be a challenge, but you need a teacher, and you need one authorized to teach computer science content. The **Teacher Credentials** tool can help. As an alternative, one leader suggested, "if you don't have a [full time] teacher, ask yourself, can I get someone [else] that's part time maybe? Maybe I can offer one class. Maybe I have to do something after school as a club. Sometimes people can't offer things and they do it as a club, so it could be an after-school activity." Options for schools without dedicated computer science teachers are discussed in the **Computer Science Online Learning Opportunities** tool.

7. Elevate the Academic Value of Computer Science

"...you have to have that clear understanding that every class will have a purpose, and that purpose has got to enhance the academic experience for the student and [make sure] that there are no dumping grounds or babysitting or non-academic classes."

Historically, computer science has been confused with technology literacy, computer applications classes, and other courses that are not associated with high academic challenge (see the **Principal's Guide to Addressing Misconceptions about CS** and **Computer Science Terminology** tools). It is important to communicate to teachers, counselors and students that computer science is a challenging academic discipline and, as the principal quoted above said, "not a dumping ground."

Copyright © 2015 by University of Chicago

Suggested citation: LeadCS.org (2015). Advice from School Leaders: Preparing for Computer Science in Your School CEMSE, Outlier Research & Evaluation, University of Chicago. <http://www.leadCS.org>