The National Science Foundation has assembled a group of thought leaders to bring the concepts of computational thinking to the K–12 classroom.

A group of high school students cluster around a computer looking at a series of graphs and charts on the screen and talking quietly but intently. They are collaborating with a group of students in South America using Skype. Together they have gathered data and created a model depicting the rate of deforestation of the rain forests around the world. Today they are discussing the changes they need to make to their data representation and algorithm before running their simulation. These students are engaged in what is called computational thinking.

What Is Computational Thinking?

In a seminal article published in 2006, Jeannette Wing described computational thinking (CT) as a way of “solving problems, designing systems, and understanding human behavior by drawing on the concepts fundamental to computer science.” She noted that computational thinking involves some familiar concepts, such as problem decomposition, data representation, and modeling, as well as less familiar ideas, such as binary search, recursion, and parallelization. She also argued that “computational thinking is a fundamental skill for everyone, not just for computer scientists. To reading, writing, and arithmetic, we should add computational thinking to every child’s analytical ability.”

Wing’s article gave rise to an often controversial discussion and debate among computer scientists, cognitive researchers, and educators regarding the nature, definition, and application of CT. While many people have proposed revisions and refinements to Wing’s original description, so far no single, widely accepted definition of computational thinking has emerged. As a result, PK–12 educators who recognize the importance of CT and want to help students acquire these skills have lacked a clear and practical definition to guide their work.

How Can We Make CT Accessible?

In 2009, the National Science Foundation (NSF) funded a project titled Leveraging Thought Leadership for Computational Thinking in PK–12. Led jointly by ISTE and the Computer Science Teachers Association (CSTA), the project is intended to make the concepts of computational thinking accessible to educators by providing an operational definition, a shared vocabulary, and relevant, age-appropriate examples of computational thinking tied to current educational objectives and classroom practices.

A year ago, the project convened a diverse group of educators with an interest in CT from higher education, PK–12, and industry to help define a common language surrounding computational thinking, articulate the challenges and opportunities of integrating it throughout PK–12 education, and identify the most promising practices and strategies for moving computational thinking from concept to deep integration.

From that meeting a consensus emerged regarding the essential elements of CT—that is, a description of its components that educators can use to build CT skills across the curriculum through all grade levels and content areas.

• The ability to communicate and work with others to achieve a common goal or solution
• Tolerance for ambiguity
• The ability to deal with open-ended problems
• The ability to communicate and work with others to achieve a common goal or solution
• The ability to communicate and work with others to achieve a common goal or solution
• The ability to communicate and work with others to achieve a common goal or solution
• The ability to communicate and work with others to achieve a common goal or solution
• The ability to communicate and work with others to achieve a common goal or solution

More than 82% of the 697 respondents agreed or strongly agreed that this definition captured the essential elements of CT. An additional 9% confirmed that the definition would do as a means to build consensus in the PK–12 community. On the basis of this survey and feedback from educators gathered through conference presentations and other informal data collection, project leaders have begun implementing the next phase of the project, which involves computational thinking in PK–12 classrooms.

Learn More

To learn more about how to teach the concepts and vocabulary of computational thinking in PK-12 classrooms, please visit iste.org/computational-thinking or the CSTA website at http://csta.acm.org. Check back in a few months to find curriculum resources, vocabulary tools, and a toolkit for leaders.

Learn More

Computational thinking is a problem-solving process that includes:

• Formulating problems in a way that enables us to use a computer and other tools to help solve them
• Logically organizing and analyzing data
• Representing data through abstractions, such as models and simulations
• Automating solutions through algorithmic thinking (a series of ordered steps)
• Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources
• Generalizing and transferring this problem-solving process to a wide variety of problems

These skills are supported and enhanced by a number of dispositions or attitudes that are essential dimensions of CT, including:

• Confidence in dealing with complexity
• Persistence in working with difficult problems
• Tolerance for ambiguity

Computer Science Teachers Association

Leveraging Thought Leadership for Computational Thinking in PK–12

CSTA

ISTE

Computational Thinking in PK–12
developing examples of how CT skills look in the classroom as well as assembling resources to support and guide the implementation of computational thinking concepts in PK–12 education.

How is CT Different?
Many of the concepts, skills, and dispositions listed in this operational definition are not new. So how is computational thinking different from critical thinking or mathematical thinking?

This question has given rise to much debate but, as yet, no widely accepted consensus. The participants in the workshops sponsored by the ISTE/CSTA project proposed that CT differs from critical thinking and mathematical thinking because:

• It is a unique combination of thinking skills that, when used together, provide the basis of a new and powerful form of problem solving.
• It is more tool-oriented.
• It makes use of familiar problem-solving skills such as trial and error, iteration, and even guessing in contexts where they were previously impractical but which are now possible because they can be automated and implemented at much higher speeds.

Why is CT Important?
The application of computer technology to virtually every field of study has changed the way work is done today. While the human mind is by far the most powerful problem-solving tool we have, the ability to extend the power of human thought with computers and other digital tools has become an essential part of our everyday lives and work. We all need to understand how, when, and where computers and other digital tools can help us solve problems, and we all need to know how to communicate with others who can assist us with computer-supported solutions.

Students already learn many elements of the set of computational thinking skills in a variety of disciplines, but we need to ensure that all students have the opportunity to learn the complete set of skills so their combined power is available to them. The NSF/ISTE/CSTA project has explored how students learn computational thinking at all grade levels and in all disciplines. The long-term goal is to recommend ways that all students have the opportunity to learn these skills and to ensure that they can be transferred to different problems and used in different contexts.

This material is based on work supported by the National Science Foundation grant CNS-1030054.