K-12 BLUEPRINT

Computational Thinking: A Vital Component of Modern K-12

Though the needs and standards of states and communities can dictate what's taught in the classroom, it is vital for learning to adapt and evolve.

The Need

The job skills market is changing rapidly with new categories of jobs emerging, partly or wholly displacing others. And the continuing impact of technology will not stop redefining jobs and skills needed for tomorrow. In fact, many of tomorrow's jobs have yet to be even thought of yet. What skills will a nano-medical engineer need? What combination of science, math, and food skills will a 3D food printer biochemist need?

According to Code.org, 90% of U.S. schools don't teach computer science as part of their curriculum. This is especially exasperating in that Code.org also reports that there will be more than 1.4 million computer jobs by 2020, yet only 400,000 students will go on to study computer science in college.

So while computer science is a primary driver of the U.S. economy, it has yet to play a significant role in K-12 education: with only 27 states allowing computer science courses to count toward high school graduation.

In a world of Internet searches, is the traditional method of lectures, rote memorization and testing the correct path? Or is it better to teach how to learn rather than what to learn?

21st Century Problem Solving

Computational thinking pushes students to be more actively involved in their education. It's a less passive and more innovative approach that has teachers and students learning together. Much like how a coach provides the structure and practice for players to become—ultimately—better than they, a computational thinking teacher isn't intimidated if their students exceed them in certain skills. In fact, that is the overriding goal.

Computational thinking requires the use of processes to pose and solve problems or prepare programs for computation. It teaches how to recognize and make use of repeating patterns: abstracting data into representations that can be easily understood. It is the foundation for most any endeavor requiring logic and algorithmic thinking, and can be immensely useful for every single student, in any career.

Computational thinking is often mistaken for coding or programming. Coding is rather like learning a new language whereas computational thinking describes a way of thinking at multiple levels of abstraction, not only the ability to program. It's an important skill set for most anyone looking to succeed in their field, not only those working with computers. Computational thinking allows someone to break down problems into smaller pieces so that they can be tackled step-by-step.

Skills for Innovation

Computational thinking is one of six key skills that Intel® believes will best support today's students to help them evolve their skills in preparation for tomorrow's workforce.

- **Simulation and Modeling:** In industry, nearly everything is simulated and designed and sometimes 3D printed before it is built. In science—when it's too expensive, impractical, or impossible to perform experiments—we use simulation and modeling. Most of our deepest insights into space, physics, and chemistry come from simulation and modeling.
- **AI/Machine Learning:** AI/machine learning are emerging technologies being applied to many industries: autonomous vehicles, manufacturing systems, and quality check systems, just to name a few. AI will continue to move toward the mainstream and become a significant component of human life.
- **Data Science:** Data science looks for information hidden in data. Analytics, AI, and machine learning all begin with insights derived from data science. Understanding and gaining insight out of data is essential to Industry 4.0.
- **Design Thinking:** Successful products and concepts are designed, not born. The design thinking process includes problem-finding, decision-making, creativity, sketching, prototyping and evaluating. New solution ideas can lead to a deeper understanding of the problem, and trigger more solution ideas.
- **Digital Content Creation:** Consider how much formal and informal education goes on through YouTube videos. Interactive learning, too, is mainstream across industries to more effectively teach a new skill or concept and assess the learning of the learner. Digital content creation is taking a new meaning with emerging virtual and augmented realities and the closing gap of human-machine interface.
- **Computational Thinking:** Computers only do what they're told or learn to do. Computational thinking teaches students to think like a computer program so it can be designed efficiently.

All of these key skills can overlap and work synergistically with one another. For instance, programming and coding technology can be used to support computational thinking through the creation of applications that represent problems and their solutions; the use of AI and Machine Learning tools; and solving complex problems by applying the Design Thinking process.

Simulation and modeling technology could also be used to support computational thinking by abstracting problems in the form of models; creating virtual representations of real-world processes or systems; and creating immersive 3D visual models.

The Right Technology

Engaging students by actively working with various types and levels of technology—such as computers and programming languages, virtual labs, design and simulation labs, artificial intelligence projects, 3D modeling, data science and others—will put their creative, math, and science skills in a living context. For technology to truly support computational thinking, it must be used to develop key skills such as the four C's (collaboration, communication, critical thinking, and creativity) and the technology-based competencies associated with modeling and simulation, digital content creation, artificial intelligence, computational thinking, and many others.

Not Just for Higher Grades

Computational thinking can be easily differentiated across grade levels since the problem complexity, for which computational thinking is applied, can be proportional to the grade. Activities to develop computational thinking could begin as early as elementary school, with simple projects such as growing plants from seeds and creating treasure maps. In middle school, this could evolve into designing mazes and creating surveys while— in high school— projects building computational thinking skills could include designing a robot to drive along a track and to predict the weather based on sensor data.

Computational Thinking and its Connection to Industry 4.0

Industry 4.0 is a term often associated with the manufacturing industry and the concept of a "smart factory" where cyber-physical distributed systems monitor both autonomous and manual processes, create virtual representations of the physical world, and then make real-time, decentralized decisions orchestrated by data and machine learning. Industry 4.0 competencies also apply across most, if not all, modern industries.

Computational thinking is a key competency associated with Industry 4.0 as it is used to decompose complex problems, identify patterns, and then design algorithms. It can help develop higher-order thinking skills and change the way students think about problem solving. And the ubiquity of digital technology is positioning computational thinking as the new literacy.

It's a must-have skill for students to succeed in the 21st century: sparking divergent thinking, creativity, problem-solving skills, and inquiry-based learning, collaboration, and persistence.

Definitions

The following are definitions of the major aspects of computational thinking:

Abstraction: The fundamental concept of object-oriented design and programming, and further includes the concept of encapsulation where each element of a system is represented by an interface that describes inputs and outputs, and the underlying details are not visible at the interface. Abstraction can also be applied to the solution.

Algorithm: A list of steps that can be followed to carry out a task. Algorithms includes the concept of data flow in which data flows through a system, and the concept of data states, in which data changes state according to set conditions.

Automation: Controlling a process by automatic means, reducing human intervention to a minimum.

Computer science: The study and use of computers and computational thinking to solve problems.

Decomposition: Breaking a problem down into smaller, more manageable parts. The idea is that a problem may not be seem solvable in its aggregate, but once reduced can be more easily solved. Once each piece is solved, then the solutions can be integrated or aggregated into a solution that solves the original, larger problem.

Pattern matching: Finding similarities between items as a way of gaining extra information.

"Every kind of industry that a student will go into as a contributing participant in our economy and our society—every sector of our economy—is underpinned by technology now. And if you understand the underpinning of computing, you will be more likely to succeed, as opposed to someone who is simply 'receiving' computer science"

—Jane Krauss, *co-author of Computational Thinking and Coding for Every Student: The Teacher's Getting-Started Guide.*