WISCONSIN STANDARDS FOR

Computer Science

Wisconsin Department of Public Instruction

WISCONSIN STANDARDS FOR
Computer Science

Wisconsin Department of Public Instruction
Tony Evers, PhD, State Superintendent
Madison, Wisconsin

This publication is available from:

Wisconsin Department of Public Instruction
125 South Webster Street
Madison, WI 53703
(608) 266-8960
http://dpi.wi.gov/computer-science

June 2017 Wisconsin Department of Public Instruction

Table of Contents

Foreword

Computer science education introduces students to topics from programming to being creators of technology. Students engaged in computer science education find relevance, application, and understanding of the core subjects to prepare them for college and careers in the 21st century.

Computer science is recognized for the numerous ways it adds value to our students' education and success. Incorporated into school curriculum as a stand-alone class or as an enhancement of other disciplines, computer science education supports our vision to prepare every Wisconsin student to be college and career ready.

To clearly identify what students in the area of computer science should know and be able to demonstrate, I have directed Wisconsin experts and educators to develop Wisconsin Standards for Computer Science through a transparent and open process with public engagement. These standards provide a foundation for aligning computer science curriculum, instruction, and assessment.

The standards within this resource will strengthen computer science options for students from kindergarten through high school and beyond. The influence of computer science and the availability of technology will shape ideas and skills gained from engaging in these computer science standards. By adopting the standards, districts drive student learning to meet their students' interests and future ambitions.

Wisconsin is a national leader in this academic area, and one of the few states to develop and adopt standards for computer science. With these nation-leading standards, we aim to improve student achievement and success while preparing all our students to be college and career ready graduates and productive citizens who will add to our economic prosperity.

Tony Evers, PhD
State Superintendent

Acknowledgements

The Wisconsin Department of Public Instruction (DPI) wishes to acknowledge the ongoing work, commitment, and various contributions of individuals to develop our state's first academic standards for computer science. Thank you to the State Superintendent's Standards Review Council for their work and guidance through the standards process. A special thanks to the Computer Science Writing Committee for taking on this important project that will shape the classrooms of today and tomorrow. Thanks to the many staff members across the division and other teams at DPI who have contributed their time and talent to this project. Finally, a special thanks to Wisconsin educators, businesspeople, parents, and citizens who provided comment and feedback to drafts of these standards.

Wisconsin Standards for Computer Science Writing Team

Co-Chairs: **Dennis Brylow,** Professor of Mathematics, Statistics, and Computer Science, Marquette University
Joe Kmoch, Retired Math and Computer Science Teacher

DPI Liaisons: **John W. Johnson,** Director, Literacy and Mathematics, and Director for Academic Standards
Mary Mooney, Mathematics Consultant
Brent Kindred, Technology and Engineering Consultant
Dave Thomas, Business and Information Technology Education Consultant

Kristina Blomquist
Pewaukee High School

Darell Cronkright
Park High School

Thomas Gendreau
UW-LaCrosse

Lori Hunt
Middleton High School

Andrew Kuemel
Madison West High School

Kristin Matchey
Arcadia High School

Annette O'Hern
LaCrosse School District

John Quinn
Wausua West High School

Daniel Rhode
Baraboo High School

Karla Saeger
UW-Whitewater

Randy Schullo
Rice Lake High School

Jill Underly
Pecatonica Area School Dist.

Doug Waterman
Fox Valley Technical College

Amanda Werner
Madison Elementary Science

Department of Public Instruction, Academic Standards

- John W. Johnson, Director, Literacy and Mathematics, and Director for Academic Standards
- Meri Annin, Lead Visual Communications Designer
- Marci Glaus, Strategic Communications Consultant
- David McHugh, Strategic Planning and Professional Learning Consultant

Department of Public Instruction Leaders

- Sheila Briggs, Assistant State Superintendent, Division of Academic Excellence
- Emilie Amundson, Chief of Staff, Office of the State Superintendent
- Scott Jones, Special Assistant, Office of the State Superintendent

Section I

Wisconsin’s Approach to Academic Standards

Purpose of the Document

The purpose of this guide is to improve Computer Science education for students and for communities. The Wisconsin Department of Public Instruction (DPI) has developed standards to assist Wisconsin educators and stakeholders in understanding, developing and implementing computer science course offerings and curriculum in school districts across Wisconsin.

This publication provides a vision for student success and follows *The Guiding Principles for Teaching and Learning (2011)*. In brief, the principles are:

1. Every student has the right to learn.
2. Instruction must be rigorous and relevant.
3. Purposeful assessment drives instruction and affects learning.
4. Learning is a collaborative responsibility.
5. Students bring strengths and experiences to learning.
6. Responsive environments engage learners.

Program leaders will find the guide valuable for making decisions about:

- Program structure and integration
- Curriculum redesign
- Staffing and staff development
- Scheduling and student grouping
- Facility organization
- Learning spaces and materials development
- Resource allocation and accountability
- Collaborative work with other units of the school, district and community

What Are the Academic Standards?

Wisconsin Academic Standards specify what students should know and be able to do in the classroom. They serve as goals for teaching and learning. Setting high standards enables students, parents, educators, and citizens to know what students should have learned at a given point in time. In Wisconsin, all state standards serve as a model. Locally elected school boards adopt academic standards in each subject area to best serve their local communities. We must ensure that all children have equal access to high-quality education programs. Clear statements about what students must know and be able to do are essential in making sure our schools offer opportunities to get the knowledge and skills necessary for success beyond the classroom.

Adopting these standards is voluntary. Districts may use the academic standards as guides for developing local grade-by-grade level curriculum. Implementing standards may require some school districts to upgrade school and district curriculums. This may result in changes in instructional methods and materials, local assessments, and professional development opportunities for the teaching and administrative staff.

What is the Difference Between Academic Standards and Curriculum?

Standards are statements about what students should know and be able to do, what they might be asked to do to give evidence of learning, and how well they should be expected to know or do it. Curriculum is the program devised by local school districts used to prepare students to meet standards. It consists of activities and lessons at each grade level, instructional materials, and various instructional techniques. In short, standards define what is to be learned at certain points in time, and from a broad perspective, what performances will be accepted as evidence that the learning has occurred. Curriculum specifies the details of the day-to-day schooling at the local level.

Developing the Academic Standards

DPI has a transparent and comprehensive process for reviewing and revising academic standards. The process begins with a notice of intent to review an academic area with a public comment period. The State Superintendent's Standards Review Council examines those comments and may recommend revision or development of standards in that academic area. The state superintendent authorizes whether or not to pursue a revision or development process. Following this, a state writing committee is formed to work on those standards for all grade levels. That draft is then made available for open review to get feedback from the public, key stakeholders, educators, and the Legislature with further review by the State Superintendent's Standards Review Council. The state superintendent then determines adoption of the standards.

Aligning for Student Success

To build and sustain schools that support every student in achieving success, educators must work together with families, community members, and business partners to connect the most promising practices in the most meaningful contexts. The release of the *Wisconsin Standards for Computer Science* provides for the first time a set of important academic standards for school districts to implement. This is connected to a larger vision of every child graduating college and career ready. The graphic below illustrates the relationship between academic standards and other critical principles and efforts that function together to educate every child to graduate college and career ready. Here, the vision and set of Guiding Principles form the foundation for building a supportive process for teaching and learning rigorous and relevant content. The following sections articulate this integrated approach to increasing student success in Wisconsin schools and communities.

Relating the Academic Standards to All Students

Grade-level standards should allow ALL students to engage, access, and be assessed in ways that fit their strengths, needs, and interests. This applies to the achievement of students with IEPs (individualized education plans), English learners, and gifted and talented pupils, consistent with all other students. Academic standards serve as the foundation for individualized programming decisions for all students.

Academic standards serve as a valuable basis for establishing concrete, meaningful goals as part of each student's developmental progress and demonstration of proficiency. Students with IEPs must be provided specially designed instruction that meets their individual needs. It is expected that each individual student with an IEP will require unique services and supports matched to their strengths and needs in order to close achievement gaps in grade-level standards. Alternate standards are only available for students with the most significant cognitive disabilities.

Gifted and talented students may achieve well beyond the academic standards and move into advanced grade levels or into advanced coursework.

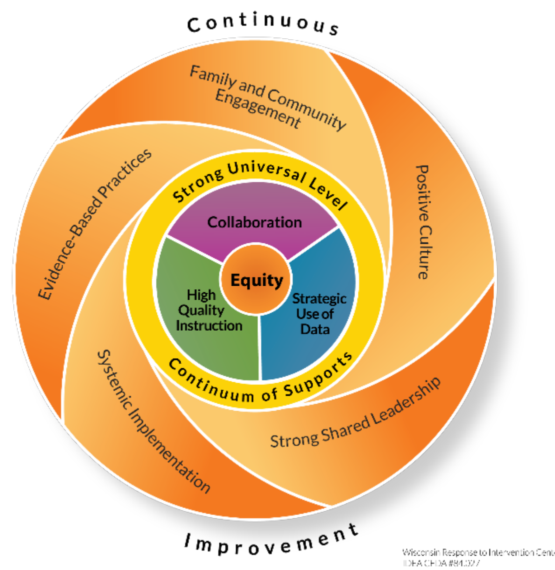
Our Vision: Every Child a Graduate, College and Career Ready

We are committed to ensuring every child graduates from high school academically prepared and socially and emotionally competent. A successful Wisconsin student is proficient in academic content and can apply their knowledge through skills such as critical thinking, communication, collaboration, and creativity. The successful student will also possess critical habits such as perseverance, responsibility, adaptability, and leadership. This vision for every child as a college and career ready graduate guides our beliefs and approaches to education in Wisconsin.

Guided by Principles

All educational initiatives are guided and impacted by important and often unstated attitudes or principles for teaching and learning. *The Guiding Principles for Teaching and Learning (2011)* emerge from research and provide the touchstone for practices that truly affect the vision of Every Child a Graduate Prepared for College and Career. When made transparent, these principles inform what happens in the classroom, direct the implementation and evaluation of programs, and most importantly, remind us of our own beliefs and expectations for students.

Ensuring a Process for Student Success



For Wisconsin schools and districts, implementing the *Framework for Equitable Multi-Level Systems of Supports (2017)* means providing equitable services, practices, and resources to every learner based upon responsiveness to effective instruction and intervention. In this system, high-quality instruction, strategic use of data, and collaboration interact within a continuum of supports to facilitate learner success. Schools provide varying types of supports with differing levels of intensity to proactively and responsibly adjust to the needs of the whole child. These include the knowledge, skills and habits learners need for success beyond high school, including developmental, academic, behavioral, social, and emotional skills.

Connecting to Content: Wisconsin Academic Standards

Within this vision for increased student success, rigorous, internationally benchmarked academic standards provide the content for high-quality curriculum and instruction and for a strategic assessment system aligned to those standards. With the adoption of the standards, Wisconsin has the tools to design curriculum, instruction, and assessments to maximize student learning. The standards articulate what we teach so that educators can focus on how instruction can best meet the needs of each student. When implemented within an equitable multi-level system of support, the standards can help to ensure that every child will graduate college and career ready.

References

The Guiding Principles for Teaching and Learning. 2011. Madison, WI: Wisconsin Department of Public Instruction. Retrieved from https://dpi.wi.gov/standards/guiding-principles.

Framework for Equitable Multi-Level Systems of Supports. 2017. Madison, WI: Wisconsin Department of Public Instruction. Retrieved from https://dpi.wi.gov/rti.

Section II

Wisconsin Standards for Computer Science

What is Computer Science Education?

Wisconsin defines computer science (CS) as an academic discipline that encompasses the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, networks, and their impact on society. The standards outlined in this document provide an important foundation to prepare students for post-secondary education and careers.

Computer Science Education in Wisconsin

Computer science drives job growth and innovation throughout the economy and society. In 2017, demand for computing jobs in Wisconsin was higher than any other occupation category, and this growth is projected to continue for much of the next decade. The need for CS education is increasing because all students will need some foundational knowledge in CS, regardless of their occupational path. To offer formal coursework and integrate CS into K-12 learning opportunities, developing CS academic standards across grades K-12 is an essential first step. In the 21st century, career and college readiness will increasingly require a CS component.

At the elementary level, CS content and concepts can be integrated throughout the curriculum. Teachers can effectively use CS concepts in instruction to develop foundational skills and also can create a connection to secondary CS options. At the middle and high school levels, all students should have access to CS, including those who wish to pursue advanced courses.

These standards articulate end-of-grade level expectations. Some students - including students with who receive special education services through an Individualized Education Program (IEP), students with gifts and talents, and English language learners - may benefit from additional supports or challenges. Some barriers to learning and engagement can be minimized through Universal Design for Learning (UDL). In addition, learning can be personalized through collaboration between educators, school staff, families, and students.

Wisconsin's Vision for Computer Science

The Wisconsin vision for computer science is shaped by Wisconsin practitioners, experts, and the business community, and is informed by work at the national level and in other states. The overarching goal is to introduce the principles and methodologies of CS to all students. Wisconsin's vision for K-12 CS is to:

1. introduce the fundamental concepts of CS to all students, beginning at the elementary school level;

2. present CS at the secondary-school level in a way that will be both accessible and worthy of a CS credit, or as a core graduation credit;
3. offer additional secondary-level CS standards that will allow interested students to study facets of CS in depth and prepare them for entry into a career or college; and
4. increase the knowledge of CS for all students, especially those from underrepresented groups in this field.

Wisconsin's Approach to Academic Standards for Computer Science

With the release of the *Wisconsin Standards for Computer Science*, Wisconsin CS teachers have access to the foundational knowledge and skills needed to educate students for successful entry into hundreds of high-wage, high-demand occupations and careers. Vetted by business, industry, and education professionals, these academic standards guide Wisconsin schools, teachers, and community partners toward development and continuous improvement of world-class CS courses.

The Computer Science Teachers Association (CSTA) is a professional organization that supports and promotes the teaching of CS. The *2011 CSTA K–12 CS Standards* represented the consensus view across the computing profession, educators, and academia. The writing of these standards was informed by an interim draft made available during 2016, as well as a separate but related K-12 Computer Science Framework under development with the involvement of many other states. The Wisconsin Standards for Computer Science share five overall conceptual strands with these previous standards documents. The learning priorities and performance indicators contained within each set of CS standards consists of knowledge and skills specific to each of the five content areas.

- Algorithms and Programming
- Computing Systems
- Data and Analysis
- Impacts of Computing
- Networks and the Internet

These are critical as students develop an understanding of CS as a discipline and how these skills intersect with other content areas. In addition, there are many knowledge areas, skills, and dispositions delineated in these CS academic standards that are common to the pursuit of careers and postsecondary education in many fields.

Numerous existing sets of standards and standards-related documents have been used in developing the Wisconsin Standards for Computer Science. These include:

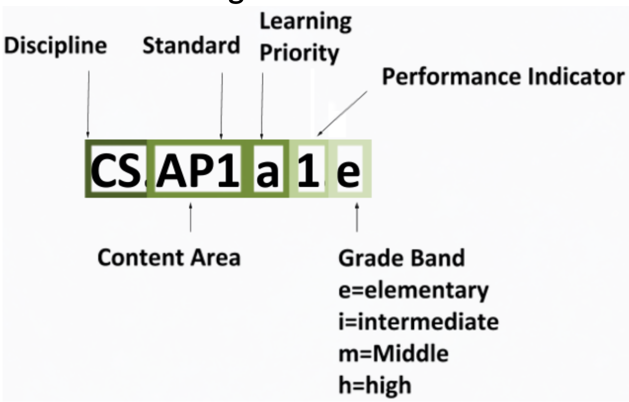
- The (interim) CSTA K-12 Computer Science Standards, revised 2016 https://www.csteachers.org/page/about-csta-s-k-12-nbsp-standards
- The K-12 Computer Science Framework https://k12cs.org/
- Approved or draft standards from the following states: Arkansas, Florida, Idaho (draft), Indiana, Massachusetts, New Jersey, South Carolina (draft), Texas, Washington

The *Wisconsin Standards for Computer Science* may be taught and integrated through a variety of classes and experiences. Each district, school, and program area should determine the means by which students meet these standards. Through the collaboration of multiple stakeholders, these foundational standards will set the stage for high-quality, successful, contemporary CS courses and programs throughout Wisconsin's PK-12 systems.

Standards Structure

The Wisconsin Standards for Computer Science follow a specific structure.

Standards Coding



Content Area: Algorithms and Programming (AP)

Standard AP1: Students will recognize and define computational problems using algorithms and programming

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP1.a: Develop algorithms.	AP1.a.1.e Construct and execute algorithms (sets of step-by-step instructions), which include sequencing and simple loops to accomplish a task, both independently and collaboratively, with or without a computing device.	AP1.a.4.i Construct and execute algorithms (sets of step-by-step instructions), which include sequencing, loops, and conditionals to accomplish a task, both independently and collaboratively, with or without a computing device.	AP1.a.6.m Decompose (break down) a computational problem into parts and create solutions for one or more parts.	AP1.a.8.h Analyze a problem and design and implement an algorithmic solution using sequence, selection, and iteration.
	AP1.a.2.e Decompose a larger computational problem into smaller sub-problems independently or with teacher guidance (e.g., to draw a snowman, we can draw several different, simpler shapes).	AP1.a.5.i Decompose a larger computational problem into smaller sub-problems independently or in a collaborative group.	AP1.a.7.m Identify how sub-problems could be recombined to create something new (e.g., break down the individual parts that would be needed to program a certain type of game and then show how the parts could be reused in other types of games).	AP1.a.9.h Explain and demonstrate how modeling and simulation can be used to explore natural phenomena (e.g., flocking behaviors, queueing, life cycles).
	AP1.a.3.e Categorize a group of items based on the attributes of actions of each item, with or without a computing device.			AP1.a.10.h (+) Provide examples of computationally solvable problems and difficult-to-solve problems.

Standards Formatting

- **Standard**: Broad statement that tells what students are expected to know or be able to do
- **Learning priority**: Breaks down the broad statement into manageable learning pieces
- **Performance indicator by grade band**: Measurable degree to which a standard has been developed or met

Grade Bands

Grade bands of K-2, 3-5, 6-8, and 9-12 align to typical elementary, middle, and high school levels

- Grade band K-2 and 3-5 performance indicators represent knowledge and skills that should be integrated throughout the elementary curriculum.
- Computer science education should be part of the core curriculum for all middle school students. Awareness, exploration, and building foundational skills should occur in middle school.
- Computer science education at the high school level continues to develop student foundational understanding of CS in the world through in-depth CS learning, including awareness and exploration activities.
- Performance indicators marked with a (+) for grades 9-12 represent advanced CS learning expectations for students with aspirations toward careers and postsecondary studies in computing disciplines.

References

Computer Science Teachers Association (2011). *CSTA K-12 Computer Science Standards*, Revised 2011. Retrieved from https://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/Docs/Standards/CSTA_K-12_CSS.pdf.

Computer Science Teachers Association (2017). *CSTA K-12 Computer Science Standards*, Revised 2017. Retrieved from http://www.csteachers.org/standards.

K–12 Computer Science Framework. *(2016).* Retrieved from http://www.k12cs.org.

Section III

Discipline: Computer Science (CS) Standards

Content Area: Algorithms and Programming (AP)

Standard AP1: Students will recognize and define computational problems using algorithms and programming

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP1.a: Develop algorithms.	AP1.a.1.e Construct and execute algorithms (sets of step-by-step instructions), which include sequencing and simple loops to accomplish a task, both independently and collaboratively, with or without a computing device.	AP1.a.4.i Construct and execute algorithms (sets of step-by-step instructions), which include sequencing, loops, and conditionals to accomplish a task, both independently and collaboratively, with or without a computing device.	AP1.a.6.m Decompose (break down) a computational problem into parts and create solutions for one or more parts.	AP1.a.8.h Analyze a problem and design and implement an algorithmic solution using sequence, selection, and iteration.
	AP1.a.2.e Decompose a larger computational problem into smaller sub-problems independently or with teacher guidance (e.g., to draw a snowman, we can draw several different, simpler shapes).	AP1.a.5.i Decompose a larger computational problem into smaller sub-problems independently or in a collaborative group.	AP1.a.7.m Identify how sub-problems could be recombined to create something new (e.g., break down the individual parts that would be needed to program a certain type of game and then show how the parts could be reused in other types of games).	AP1.a.9.h Explain and demonstrate how modeling and simulation can be used to explore natural phenomena (e.g., flocking behaviors, queueing, life cycles).
	AP1.a.3.e Categorize a group of items based on the attributes of actions of each item, with or without a computing device.			AP1.a.10.h (+) Provide examples of computationally solvable problems and difficult-to-solve problems.

NOTE: This standard continued on next page.

Content Area: Algorithms and Programming (AP)

Standard AP1: Students will recognize and define computational problems using algorithms and programming (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP1.a: (cont'd) Develop algorithms.				AP1.a.11.h (+) Decompose a large-scale computational problem by identifying generalizable patterns and applying them in a solution.
				AP1.a.12.h (+) Illustrate the flow of execution of a recursive algorithm.
				AP1.a.13.h (+) Describe how parallel processing can be used to solve large computational problems (e.g., SETI at Home, FoldIt).
				AP1.a.14.h (+) Develop and use a series of test cases to verify that a program performs according to its design specifications.
				AP1.a.15.h (+) Explain the value of heuristic algorithms (discovery methods) to approximate solutions for difficult-to-solve computational problems.

Content Area: Algorithms and Programming (AP)

Standard AP2: Students will create computational artifacts using algorithms and programming

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP2.a: Develop and implement an artifact.	AP2.a.1.e Construct programs to accomplish a task or as a means of creative expression, which include sequencing, events, and simple loops, using a block-based visual programming language, both independently and collaboratively (e.g., pair programming).	AP2.a.3.i Construct programs to solve a problem or for creative expression, which include sequencing, events, loops, conditionals, parallelism, and variables, using a block-based visual programming language or text-based language, both independently and collaboratively (e.g., pair programming).	AP2.a.6.m Develop programs, both independently and collaboratively, which include sequencing with nested loops and multiple branches [Clarification At this level, students may use block-based and/or text-based languages].	AP2.a.10.h Use user-centered research and design techniques (e.g., surveys, interviews) to create software solutions.
	AP2.a.2.e Plan and create a design document to illustrate thoughts, ideas, and stories in a sequential (step-by-step) manner (e.g., story map, storyboard, sequential graphic organizer).	AP2.a.4.i Create a plan as part of the iterative design process, both independently and with diverse collaborative teams (e.g., storyboard, flowchart, pseudo-code, story map).	AP2.a.7.m Produce computational artifacts with broad accessibility and usability through careful consideration of diverse needs and wants of the community.	AP2.a.11.h Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computational artifacts.
	AP2.a.5.i Use mathematical operations to change a value stored in a variable.	AP2.a.8.m Use an iterative design process (e.g., define the problem; generate ideas; build, test, and improve solutions) to solve computational problems, both independently and collaboratively.	AP2.a.12.h Design, develop, and implement a computing artifact that responds to an event (e.g., robot that responds to a sensor, mobile app that responds to a text message, sprite that responds to a broadcast).	AP2.a.5.i Use mathematical operations to change a value stored in a variable.

NOTE: This standard continued on next page.

Content Area: Algorithms and Programming (AP)

Standard AP2: Students will create computational artifacts using algorithms and programming (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP2.a: (cont'd) Develop and implement an artifact.			AP2.a.9.m Create variables that represent different types of data and manipulate their values.	AP2.a.13.h (+) Decompose a computational problem by creating new data types, functions, or classes.
				AP2.a.14.h (+) Develop programs for multiple computing platforms (e.g., computer desktop, web, mobile).
				AP2.a.15.h (+) Implement an Artificial Intelligence (AI) algorithm to play a game against a human opponent or solve a problem.
				AP2.a.16.h (+) Demonstrate code reuse by creating programming solutions using libraries and application program interfaces (APIs) (e.g., graphics libraries, maps, API).

Content Area: Algorithms and Programming (AP)

Standard AP3: Students will communicate about computing ideas

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP3.a: Recognize and cite sources.	AP3.a.1.e Give credit to the source when using code, music, or pictures that were created by others.	AP3.a.2.i Use proper citations and document when ideas are borrowed and changed for their own use (e.g., using pictures created by others, using music created by others, remixing programming projects).	AP3.a.3.m Provide proper attribution when code is borrowed or built upon.	AP3.a.4.h Compare and contrast various software licensing schemes (e.g., open source, freeware, commercial).
AP3.b: Communicate about technical and social issues.	AP3.b.1.e Follow simple instructions to complete a task, such as a simple visual tutorial.	AP3.b.2.i Understand that algorithms have impacted society in both beneficial and harmful ways.	AP3.b.5.m Discuss how algorithms have impacted society—both the beneficial and harmful effects.	AP3.b.8.h Evaluate and analyze how algorithms have impacted our society and discuss the benefits and harmful impacts of a variety of technological innovations.
		AP3.b.3.i Compare different problem-solving techniques.	AP3.b.6.m Compare different algorithms that may be used to solve the same problem in terms of their speed, clarity, and size (e.g., different algorithms solve the same problem, but one might be faster than the other). [Clarification Students are not expected to quantify these differences].	AP3.b.9.h (+) Compare a variety of programming languages and identify features that make them useful for solving different types of problems and developing different kinds of systems (e.g., declarative, logic, parallel, functional, compiled, interpreted, real-time).

NOTE: This standard continued on next page.

Content Area: Algorithms and Programming (AP)

Standard AP3: Students will communicate about computing ideas (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP3.b: (cont'd) Communicate about technical and social issues.		AP3.b.4.i Modify a set of instructions (e.g., in dancing, cooking, or other areas) and discuss how many paths can lead to the same result.	AP3.b.7.m Modify existing code to change its functionality and discuss the variety of ways in which to do this.	AP3.b.10.h (+) Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).
AP3.c: Document code.			AP3.c.1.m Interpret the flow of execution of algorithms and predict their outcomes. [Clarification Algorithms can be expressed using natural language, flow and control diagrams, comments within code, and pseudocode].	AP3.c.3.h (+) Describe how Artificial Intelligence (AI) drives many software and physical systems (e.g., autonomous robots, computer vision, pattern recognition, text analysis).
			AP3.c.2.m Use documentation regarding code to modify programs.	AP3.c.4.h Write appropriate documentation for programs.
				AP3.c.5.h (+) Use application programming interface (APIs) documentation resources.
				AP3.c.6.h Use online resources to answer technical questions.

Content Area: Algorithms and Programming (AP)

Standard AP4: Students will develop and use abstractions

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP4.a: Create and use abstractions (representations) to solve complex computational problems.	AP4.a.1.e Use numbers or other symbols to represent data (e.g., thumbs up or down for yes or no, color by number, arrows for direction, encoding or decoding a word using numbers or pictographs).	AP4.a.2.i Use several existing functions or procedures to solve a problem (e.g., using several square, circle, and triangle drawing functions to create a larger picture).	AP4.a.3.m Define and use functions/ procedures that hide the complexity of a task and can be reused to solve similar tasks. [Clarification Students use and modify, but do not necessarily create, functions or procedures with parameters].	AP4.a.4.h Demonstrate the value of abstraction for managing problem complexity (e.g., using a list instead of discrete variables).
				AP4.a.5.h Understand the notion of hierarchy and abstraction in high-level languages, translation, instruction sets, and logic circuits.
				AP4.a.6.h Deconstruct a complex problem into simpler parts using predefined constructs (e.g., functions and parameters and/or classes).
				AP4.a.7.h (+) Compare and contrast fundamental data structures and their uses (e.g., lists, maps, arrays, stacks, queues, trees, graphs).

NOTE: This standard continued on next page.

Content Area: Algorithms and Programming (AP)

Standard AP4: Students will develop and use abstractions (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP4.a: (cont'd) Create and use abstractions (representations) to solve complex computational problems.				AP4.a.8.h (+) Critically analyze and evaluate classic algorithms (e.g., sorting, searching) and use in different contexts, adapting as appropriate.
				AP4.a.9.h (+) Discuss issues that arise when breaking large-scale problems down into parts that must be processed simultaneously on separate systems (e.g., cloud computing, parallelization, concurrency).
				AP4.a.10.h (+) Define the functionality of an abstraction without implementing the abstraction.
				AP4.a.11.h (+) Evaluate algorithms (e.g., sorting, searching) in terms of their efficiency, correctness, and clarity.

NOTE: This standard continued on next page.

Content Area: Algorithms and Programming (AP)

Standard AP4: Students will develop and use abstractions (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP4.a: (cont'd) Create and use abstractions (representations) to solve complex computational problems.				AP4.a.12.h (+) Identify programming language features that can be used to define or specify an abstraction.
				AP4.a.13.h (+) Identify abstractions used in a solution (program or software artifact) and reuse those abstractions to solve a different problem.

Content Area: Algorithms and Programming (AP)

Standard AP5: Students will collaborate with diverse teams

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP5.a: Work together to solve computational problems using a variety of resources.	AP5.a.1.e Work together with a team to create a solution to a computational problem.	AP5.a.3.i Apply collaboration strategies to support problem solving-within the design cycle of a program.	AP5.a.5.m Solicit and integrate peer feedback as appropriate to develop or refine a program.	AP5.a.6.h Design and develop a software artifact working in a team.
	AP5.a.2.e Use teachers, parents, and other resources to solve a computational problem.	AP5.a.4.i Understand there are many resources that can be used or tapped to solve a problem.		AP5.a.7.h Demonstrate how diverse collaborating impacts the design and development of software products (e.g., discussing real-world examples of products that have been improved through having a diverse design team or reflecting on their own team's development experience).
				AP5.a.8.h (+) Demonstrate software life cycle processes (e.g., spiral, waterfall) by participating on software project teams (e.g., community service project with real-world clients).

NOTE: This standard continued on next page.

Content Area: Algorithms and Programming (AP)

Standard AP5: Students will collaborate with diverse teams (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP5.a: (cont'd) Work together to solve computational problems using a variety of resources.				AP5.a.9.h (+) Use version control systems, integrated development environments (IDEs), and collaboration tools and practices (code documentation) in a group software project.
AP5.b: Foster an inclusive computing culture.	AP5.b.1.e Understand the value for teams to include members with different perspectives, experiences, and backgrounds, including race, gender, ethnicity, language, ability, family background, and family income.		AP5.b.2.m Analyze team members' strengths and use them to foster an inclusive computing culture.	AP5.b.3.h Create design teams taking into account the strengths and perspectives of potential team members.

Content Area: Algorithms and Programming (AP)

Standard AP6: Students will test and refine computational solutions

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
AP6.a: Test and debug computational solutions.	AP6.a.1.e Analyze and debug (fix) an algorithm, which includes sequencing and simple loops, with or without a computing device.	AP6.a.2.i Analyze and debug an algorithm, which includes sequencing, events, loops, conditionals, parallelism, and variables.	AP6.a.3.m Use testing and debugging methods to ensure program correctness and completeness.	AP6.a.4.h Use a systematic approach and debugging tools to independently debug a program (e.g., setting breakpoints, inspecting variables with a debugger).
AP6.b: Develop and apply success criteria.		AP6.b.1.i Determine the correctness of a computational problem solution by listening to a classmate describe the solution.	AP6.b.2.m Apply a rubric to determine if and how well a program meets objectives.	AP6.b.3.h (+) Evaluate key qualities of a program (e.g., correctness, usability, readability, efficiency, portability, scalability) through a process such as a code review.

Content Area: Computing Systems (CS)

Standard CS1: Students will communicate about computing systems

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
CS1.a: Identify hardware and software components.	CS1.a.1.e Identify and use software that controls computational devices to accomplish a task (e.g., use an app to draw on the screen, use software to write a story or control robots).	CS1.a.3.i Select and operate appropriate software to perform a variety of tasks and recognize that users have different needs and preferences for the technology they use.	CS1.a.5.m Justify the suitability of hardware and software chosen to accomplish a task (e.g., comparison of the features of a tablet vs. desktop, selecting which sensors and platform to use in building a robot or developing a mobile app).	CS1.a.6.h Develop and apply criteria (e.g., power consumption, processing speed, storage space, battery life, cost, operating system) for evaluating a computer system for a given purpose (e.g., system specification needed to run a game, web browsing, graphic design, or video editing).
	CS1.a.2.e Use appropriate terminology in naming and describing the function of common computing devices and components (e.g., desktop computer, laptop computer, tablet device, monitor, keyboard, mouse, printer).	CS1.a.4.i Use appropriate terminology in naming internal and external components of computing devices and describing their relationships, capabilities, and limitations.		CS1.a.7.h (+) Identify the functionality of various categories of hardware components and communication between them (e.g., physical layers, logic gates, chips, input and output devices).

NOTE: This standard continued on next page.

Content Area: Computing Systems (CS)

Standard CS1: Students will communicate about computing systems

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
CS1.b: Understand how the components of a computer system work together.	CS1.b.1.e Identify the components of a computer system and what the basic functions are (e.g., hard drive, and memory) as well as external features and their uses (e.g., printers, scanners, external hard drives, and cloud storage).	CS1.b.2.i Model how a computer system works. [Clarification Only includes basic elements of a computer system, such as input, output, processor, sensors, and storage].		CS1.b.3.h (+) Explain the role of operating systems (e.g., how programs are stored in memory, how data is organized and retrieved, how processes are managed and multi-tasked).

Content Area: Computing Systems (CS)

Standard CS2: Students will test and refine computing systems

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
CS2.a: Problem solve and debug.	CS2.a.1.e Identify, using accurate terminology, simple hardware and software problems that may occur during use (e.g., app or program not working as expected, no sound, device won't turn on).	CS2.a.2.i Identify, using accurate terminology, simple hardware and software problems that may occur during use, and apply strategies for solving problems (e.g., reboot device, check for power, check network availability, close and reopen app).	CS2.a.3.m Use a systematic process to identify the source of a problem within individual and connected devices (e.g., follow a troubleshooting flow diagram, make changes to software to see if hardware will work, restart device, check connections, swap in working components).	CS2.a.4.h Devise a systematic process to identify the source of a problem within individual and connected devices (e.g., research, investigate, problem solve).

Content Area: Computing Systems (CS)

Standard CS3: Students will develop and use abstractions in computing systems

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
CS3.a: Generalize in computer systems.			CS3.a.1.m Analyze the relationship between a device's computational components and its capabilities. (e.g., computing systems include not only computers, but also cars, microwaves, smartphones, traffic lights, and flash drives).	CS3.a.2.h Demonstrate the role and interaction of a computer embedded within a physical system, such as a consumer electronic, biological system, or vehicle, by creating a diagram, model, simulation, or prototype.
				CS3.a.3.h (+) Describe the steps necessary for a computer to execute high-level source code (e.g., compilation to machine language, interpretation, fetch-decode-execute cycle).

Content Area: Computing Systems (CS)

Standard CS4: Students will create and modify computing systems

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
CS4.a: Modify and create computational artifacts.			CS4.a.1.m Extend or modify existing programs to add simple features and behaviors using different forms of inputs and outputs (e.g., inputs such as sensors, mouse clicks, data sets; outputs such as text, graphics, sounds).	CS4.a.2.h Create, extend, or modify existing programs to add new features and behaviors using different forms of inputs and outputs (e.g., inputs such as sensors, mouse clicks, data sets; outputs such as text, graphics, sounds).
				CS4.a.3.h (+) Create a new artifact that uses a variety of forms of inputs and outputs (e.g., inputs such as sensors, mouse clicks, data sets; outputs such as text, graphics, sounds).

Content Area: Data and Analysis (DA)

Standard DA1: Students will create computational artifacts using data and analysis

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
DA1.a: Represent and manipulate data.		DA1.a.1.i Use numeric values to represent non-numeric ideas in the computer (e.g., binary, American Standard Code for Information Interchange (ASCII), pixel attributes such as Red Green Blue (RGB)).	DA1.a.3.m Represent data using different encoding schemes (e.g., binary, Unicode, Morse code, shorthand, student-created codes).	DA1.a.4.h Convert between binary, decimal, and hexadecimal representations of data (e.g., convert hexadecimal color codes to decimal percentages, ASCII/ Unicode representation).
		DA1.a.2.i Answer a question by using a computer to manipulate (e.g., sort, total and/or average, chart, graph) and analyze data that has been collected by the class or student.		DA1.a.5.h Analyze the representation tradeoffs among various forms of digital information (e.g., lossy vs. lossless compression, encrypted vs. unencrypted, various image representations).
				DA1.a.6.h (+) Discuss how data sequences (e.g., binary, hexadecimal, octal) can be interpreted in a variety of forms (e.g., instructions, numbers, text, sound, image).

Content Area: Data and Analysis (DA)

Standard DA2: Students will recognize and define data in computational problems

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
DA2.a: Gather data to support computational problem solving.	DA2.a.1.e Collect simple quantitative data over time (e.g., daily temperatures or sunrise time).	DA2.a.2.i Collect quantitative data over time from multiple sources (e.g., class or group pools, individual observations of street traffic).	DA2.a.3.m Gather and organize multiple quantitative data elements using a computational tool (e.g., spreadsheet software).	DA2.a.4.h Discuss techniques used to store, process, and retrieve different amounts of information (e.g., files, databases, data warehouses).
				DA2.a.5.h (+) Use various data collection techniques for different types of computational problems (e.g., mobile device Global Positioning System (GPS), user surveys, embedded system sensors, open data sets, social media data sets).
DA2.b: Categorize and analyze data.	DA2.b.1.e Sort objects into buckets, recognizing relevant and/or irrelevant data (e.g., one of these things is not like the other).	DA2.b.2.i Choose appropriate classifications or grouping for data by shape, color, size, or other attributes.	DA2.b.3.m Develop a strategy to answer a question by using a computer to manipulate (e.g., sort, total and/or average, chart, graph) and analyze data that has been collected by the class or student.	DA2.b.4.h Apply basic techniques for locating and collecting small- and large-scale data sets (e.g., creating and distributing user surveys, accessing real-world data sets).

Content Area: Data and Analysis (DA)

Standard DA3: Students will communicate about data in computing

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
DA3.a: Communicate about data.	DA3.a.1.e Collect data over time and organize it in a chart or graph in order to make and communicate a prediction.	DA3.a.2.i Organize data into new subsets to provide different views or commonalities and present insights gained using visual representations.	DA3.a.4.m Describe how different formats of stored data represent tradeoffs between quality and size. [Clarification compare examples of music, text and/or image formats].	DA3.a.6.h Use computational tools to collect, transform, and organize data about a problem to explain to others.
		DA3.a.3.i Organize and evaluate data for its sufficiency and relevance to making accurate inferences or predictions.	DA3.a.5.m Explain the processes used to collect, transform, and analyze data to solve a problem using computational tools (e.g., use an app or spreadsheet form to collect data, decide which data to use or ignore, and choose a visualization method).	

Content Area: Data and Analysis (DA)

Standard DA4: Students will develop and use data abstractions

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
DA4.a: Model with data.	DA4.a.1.e Use a computing device to store, search, retrieve, modify, and delete information and define the information stored as data.	DA4.a.3.i Create a computational artifact to model the attributes and behaviors associated with a concept (e.g., solar system, life cycle of a plant).	DA4.a.4.m Revise computational models to more accurately reflect real-world systems (e.g., ecosystems, epidemics, spread of ideas).	DA4.a.6.h Create computational models that simulate real-world systems (e.g., ecosystems, epidemics, spread of ideas).
	DA4.a.2.e Create a model of an object or process in order to identify patterns and essential elements (e.g., water cycle, butterfly life cycle, seasonal weather patterns).		DA4.a.5.m Modify an existing computational model to emphasize key features and relationships within a system. (A model can be used to simulate events, examine theories and inferences, or make predictions).	DA4.a.7.h (+) Evaluate the ability of models and simulations to formulate, refine, and test hypotheses.
DA4.b: Identify patterns.				DA4.b.1.h (+) Use data analysis to identify significant patterns in complex systems (e.g., take existing data sets and make sense of them).

NOTE: This standard continued on next page.

Content Area: Data and Analysis (DA)

Standard DA4: Students will develop and use data abstractions (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
DA4.b: (cont'd) Identify patterns.				DA4.b.2.h (+) Identify mathematical and computational patterns through modeling and simulation (e.g., regression, queueing theory, discrete event simulation).
				DA4.b.2.h (+) Identify mathematical and computational patterns through modeling and simulation (e.g., regression, queueing theory, discrete event simulation).

Content Area: Impacts of Computing (IC)

Standard IC1: Students will understand the impact and effect computing technology has on our everyday lives

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
IC1.a: Understand the impact technology has on our everyday lives and the effects of computing on the economy and culture.	IC1.a.1.e Compare and contrast examples of how computing technology has changed the way people live, work, and interact.	IC1.a.2.i Discuss computing technologies that have changed the world and express how those technologies influence, and are influenced by, cultural practices.	IC1.a.4.m Provide examples of how computational artifacts and devices impact health and wellbeing, both positively and negatively, locally and globally (e.g., effects of globalization, and automation).	IC1.a.6.h Debate the social and economic implications associated with ethical and unethical computing practices (e.g., intellectual property rights, hacktivism, software piracy, new computers shipped with malware).
		IC1.a.3.i Generate examples of how computing can affect society, and also how societal values can shape computing choices.	IC1.a.5.m Explain how computer science fosters innovation and can enhance careers and disciplines.	IC1.a.7.h Discuss implications of the collection and large-scale analysis of information about individuals (e.g., how businesses, social media, and government collect and use personal data).
				IC1.a.8.h Compare and debate the positive and negative impacts of computing on behavior and culture (e.g., evolution from hitchhiking to ride-sharing apps, online accommodation rental services).

NOTE: This standard continued on next page.

Content Area: Impacts of Computing (IC)

Standard IC1: Students will understand the impact and effect computing technology has on our everyday lives (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
IC1.a: (cont'd) Understand the impact technology has on our everyday lives and the effects of computing on the economy and culture.				IC1.a.9.h Describe how computation shares features with art and music by translating human intention into an artifact.
				IC1.a.10.h (+) Develop criteria to evaluate the beneficial and harmful effects of computing innovations on people and society.
IC1.b: Understand the effects of computing on communication and relationships.	IC1.b.1.e Explain the differences between communicating electronically and communicating in person.	IC1.b.2.i Compare and contrast the effects of communicating electronically to communicating in person.	IC1.b.3.m Analyze and present beneficial and harmful effects of personal electronic communication and social electronic communication.	IC1.b.5.h Evaluate the negative impacts of electronic communication on personal relationships and evaluate differences between face-to-face and electronic communication.

NOTE: This standard continued on next page.

Content Area: Impacts of Computing (IC)

Standard IC1: Students will understand the impact and effect computing technology has on our everyday lives (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
IC1.b: (cont'd) Understand the effects of computing on communication and relationships.			IC1.b.4.m Describe ways in which the internet impacts global communication and collaborating.	IC1.b.6.h (+) Create a list of practices that individuals and organizations can use to encourage proper use of both electronic and face-to-face communication.
				IC1.b.7.h (+) Evaluate the negative impacts on societal discourse caused by social media and electronic communities.

Content Area: Impacts of Computing (IC)

Standard IC2: Students will experience learning within a collaborative, inclusive computing culture and explain the steps needed to ensure that all people have access to computing

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
IC2.a: Understand the effects of the digital divide.		IC2.a.1.i Brainstorm and advocate for ways in which computing devices and the internet could be made more available to all people.	IC2.a.2.m Explain the impact of the digital divide (i.e., uneven access to computing, computing education, and interfaces) on access to critical information.	IC2.a.3.h (+) Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.
IC2.b: Test and refine digital artifacts for accessibility.		IC2.b.1.i Brainstorm ways in which computing devices could be made more accessible to all users.	IC2.b.2.m Critically evaluate and redesign a computational artifact to remove barriers to universal access (e.g., using captions on images, high contrast colors, and/or larger font sizes).	IC2.b.3.h Design a user interface (e.g., web pages, mobile applications, animations) to be more inclusive and accessible, minimizing the impact of the designer's inherent bias.
IC2.c: Collaborate ethically in the creation of digital artifacts.	IC2.c.1.e Work with others as co-learners to solve a problem or reach a goal.	IC2.c.2.i Use online collaborative spaces ethically and safely to work with another student to solve a problem or reach a goal.	IC2.c.4.m Use the internet ethically and safely to work with a group of people who are not physically near to solve a problem or reach a goal.	IC2.c.5.h Ethically and safely select, observe, and contribute to global collaboration in the development of a computational artifact (e.g., contribute the resolution of a bug in an open-source project platform, or contribute an online article).

NOTE: This standard continued on next page.

Content Area: Impacts of Computing (IC)

Standard IC2: Students will experience learning within a collaborative, inclusive computing culture and explain the steps needed to ensure that all people have access to computing (cont'd)

Performance Indicators (by Grade Band)

IC2.c: (cont'd) **Collaborate ethically in the creation of digital artifacts.**		IC2.c.3.i Seek out and compare diverse perspectives, synchronously or asynchronously, to improve a project.		IC2.c.6.h Demonstrate how computing enables new forms of experience, expression, communication, and collaboration.

Content Area: Impacts of Computing (IC)

Standard IC3: Students will understand the importance of proper use of data and information in a computing society

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
IC3.a: Understand intellectual property and fair use.		IC3.a.1.i Use resources from the World Wide Web in making artifacts and recognize that the work came from others.	IC3.a.2.m Understand laws associated with digital information (e.g., intellectual property, fair use, and Creative Commons).	IC3.a.4.h Compare and contrast information access and distribution rights.
			IC3.a.3.m Describe ethical issues that relate to computing devices and networks (e.g., equity of access, security, hacking, intellectual property, copyright, Creative Commons licensing, and plagiarism).	
IC3.b: Assess the practice of digital privacy.	IC3.b.1.e Respect other students' information and refrain from accessing others' devices or accounts without permission.	IC3.b.3.i Explain problems that relate to using computing devices and networks (e.g., logging out to deter others from using your account, cyberbullying, privacy of personal information, and ownership).	IC3.b.4.m Analyze and summarize negative and positive impacts of using data and information to categorize people, predict behavior, and make recommenda-tions based on those predictions (e.g., customizing search results or targeted advertising based on previous browsing history can save search time and limit options at the same time).	IC3.b.5.h Research and understand misuses of private digital information in our society.

NOTE: This standard continued on next page.

Content Area: Impacts of Computing (IC)

Standard IC3: Students will understand the importance of proper use of data and information in a computing society (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
IC3.b: (cont'd) Assess the practice of digital privacy.	IC3.b.2.e Understand what kinds of digital information is considered private, take steps to keep their information private, and respect the privacy of other students' information.			IC3.b.6.h Debate laws regarding an individual's digital privacy and be able to explain the main arguments from multiple perspectives.
IC3.c Assess interrelationship between computing and society.				IC3.c.1.h (+) Design and implement a study that evaluates how computation has revolutionized an aspect of our culture or predicts how an aspect might evolve (e.g., education, healthcare, art/entertainment, energy).
				IC3.c.2.h (+) Debate laws and regulations that impact the development and use of software and be able to explain the main arguments from multiple perspectives.

Content Area: Networking and the Internet (NI)

Standard NI1: Students will understand the importance of security when using technology

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
NI1.a: Use secure practices for personal computing.	NI1.a.1.e Use secure practices (such as passwords) to protect private information and discuss the effects of misuse.	NI1.a.2.i Create examples of strong passwords, explain why strong passwords should be used, and demonstrate proper use and protection of personal passwords.	NI1.a.4.m Analyze and summarize security risks associated with weak passwords, lack of encryption, insecure transactions, and persistence of data.	NI1.a.6.h Provide examples of personal data that should be kept secure and the methods by which individuals keep their private data secure.
		NI1.a.3.i Remember basic concepts and facts regarding security issues with general computer use.	NI1.a.5.m Understand security issues with general computer use.	NI1.a.7.h (+) Explain security issues that might lead to compromised computer programs (e.g., circular references, ambiguous program calls, lack of error checking, and field size checking).
NI1.b: Understand the importance of institutional security.		NI1.b.1.i Give examples of information that organizations keep private as opposed to information that they make public.	NI1.b.2.m Explain the principles of information security (confidentiality, integrity, availability) and authentication techniques.	NI1.b.3.h Compare and contrast multiple viewpoints on cybersecurity (e.g., from the perspective of security experts, privacy advocates, national security).
				NI1.b.4.h Identify digital and physical strategies to secure networks and discuss the tradeoffs between ease of access and need for security.

Content Area: Networking and the Internet (NI)

Standard NI2: Students will understand how information is sent by the internet

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
NI2.a: Demonstrate how the internet works at the physical layer.	NI2.a.1.e Use a physical tool (e.g. flashlight, string) to communicate with another student.	NI2.a.3.i Model how a device on a network sends a message from one device (sender) to another (receiver) while following specific rules.	NI2.a.6.m Simulate how information is transmitted as packets through multiple devices over the internet and networks.	NI2.a.8.h Illustrate the basic components of computer networks (e.g., draw logical and topological diagrams of networks including routers, switches, servers, and end user devices; create model with string and paper).
	NI2.a.2.e Provide examples of computer use that involve the internet.	NI2.a.4.i Differentiate between using the internet and not using the internet (e.g. identify difference between local and remote computation, such as collaborating on a Google Doc in “the cloud” versus editing a local document).	NI2.a.7.m Explain, using basic terms, how a wireless or cellular network allows internet information to be transmitted from a server to a user device.	NI2.a.9.h (+) Explain ways in which the internet is decentralized and fault-tolerant.
		NI2.a.5.i Illustrate how information travels on the internet.		NI2.a.10.h (+) Simulate and discuss the issues (e.g., bandwidth, load, delay, topology) that impact network functionality (e.g., use free network simulators).

NOTE: This standard continued on next page.

Content Area: Networking and the Internet (NI)

Standard NI2: Students will understand how information is sent by the internet (cont'd)

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
NI2.b: Demonstrate how the internet works at the protocol layer.		NI2.b.1.i Act out a protocol that people use in common everyday communications (e.g., checking out a book from the library, meeting a new person, making an appointment, playing a class game, or calling a friend on the phone to invite them over).	NI2.b.2.m Define the term protocol, provide an example of protocols in daily life, and explain their use on the internet.	NI2.b.3.h Describe key protocols and underlying processes of internet-based services (e.g., http/https and Simple Mail Transfer Protocol (SMTP) or Internet Message Access Protocol (IMAP), routing protocols).
NI2.c: Demonstrate how the internet works at the addressing layer.	NI2.c.1.e Devise a system for sending a physical message to anyone in their school by using addressing techniques (e.g., address envelopes by student first name and teacher, grade, or room).	NI2.c.2.i Devise a system for sending a physical message to anyone in their school by using addressing techniques, and then draw a tree or visual representation of their addressing system, and finally act out their addressing system by sending messages.	NI2.c.3.m Explain the hierarchical structure of the Internet Domain Name System (IDNS).	NI2.c.4.h (+) Evaluate how the hierarchical nature of the Domain Name System helps the internet work efficiently.

NOTE: This standard continued on next page.

Content Area: Networking and the Internet (NI)

Standard NI2: Students will understand how information is sent by the internet

Performance Indicators (by Grade Band)

Learning Priority	K-2 (e)	3-5 (i)	6-8 (m)	9-12 (h)
NI2.d: Demonstrate and explain encryption methods.		NI2.d.1.i Communicate across a classroom using a secure method of their own design (e.g., pictures, physical movement, text).	NI2.d.2.m Encode and decode text-based messages using basic algorithms (e.g., shift cipher, substitution cipher).	NI2.d.3.h Write a program that performs basic encryption (e.g., shift cipher, substitution cipher).
				NI2.d.4.h (+) Explain the features of public key cryptography.
				NI2.d.5.h (+) Explore security policies by implementing and comparing encryption and authentication strategies (e.g., secure coding, safeguarding keys).

Section IV

Section IV

Disciplinary Literacy: Literacy for Learning in Computer Science

What is Disciplinary Literacy?

Computer scientists have unique ways of accessing and communicating information through specialized language and text specific to computer science. Students benefit from educators who understand computer science practices in order to link language skills to this complex content. Disciplinary literacy in computer science focuses on the unique ways that scientists interact with texts such as programming languages.

In Wisconsin, disciplinary literacy is defined as the confluence of content knowledge, experiences, and skills merged with the ability to read, write, listen, speak, think critically, and perform in a way that is meaningful within the context of a given field.

The *Wisconsin Academic Standards for Literacy in All Subjects* are connected to each set of content-specific standards to guide educators as they strive to help students meet the literacy challenges within each particular field of study. This national effort is referred to as disciplinary literacy.

Disciplinary literacy is important in ALL courses and subjects at all grade levels. Therefore, the *Wisconsin Academic Standards for Literacy in All Subjects* provide standards for cross-discipline literacy in **all disciplines and every grade level K-12.** This literacy focus must begin as soon as children have access to formal education and continue intentionally as college and career readiness goals advance for all children in Wisconsin.

Elementary classroom teachers build the foundational literacy skills necessary for students to access all learning. Additionally, they develop content-specific literacy skills to read, write, listen, speak, and think critically in each discipline beginning at an early age. The applicable K-5 standards help educators in Wisconsin build a ladder of skills and dispositions that lead to accelerated achievement across disciplines.

Why is Disciplinary Literacy Important?

The modern global society, of which our students are a part, requires postsecondary learning. An analysis of workforce trends by Georgetown University economist Anthony Carnevale and his colleagues found that likely 65 percent of all job openings in 2020 will require some postsecondary education. Postsecondary success depends on students' ability to comprehend and produce the kinds of complex texts found in all disciplines. Therefore, the economic future of our state, as well as our students and their success as productive citizens and critical thinkers, links to disciplinary literacy.

Textbooks, articles, manuals, and historical primary source documents create specialized challenges for learners. These texts often include abstracts, figures, tables, diagrams, and specialized vocabulary. The ideas are complex and build across a number of paragraphs requiring focus and strategic processing. To comprehend and produce this type of text, students must be immersed in the language and thinking processes of that discipline and they must be supported by an expert guide, their teacher (Carnegie Report, 2010).

The message is that literacy is integral to attainment of content knowledge and content is essential background knowledge for literacy development. This interdependent relationship exists in all disciplines.

A focus at the elementary level on foundational reading, when expanded to include engaging experiences connected to informational texts, vocabulary, and writing for content-specific purposes, builds background knowledge and skills in each discipline. This increases opportunities for success as students approach more rigorous content in those disciplines (Alliance for Excellent Education, 2011).

Reading, writing, speaking, listening, and critical thinking must be integrated into each discipline across all grades so that all students gradually build knowledge and skills toward college and career readiness. Collaboration among institutes of higher education, CESA Statewide Network, districts, schools, teachers, and family and community will guide the implementation of the standards in Wisconsin.

The Wisconsin State Standards require educators to support literacy in each classroom across the state. Since the impact of this effort is significant, it is essential that resources and supports be accessible to all educators. To build consistent understanding, DPI convened a statewide Disciplinary Literacy Leadership Team in 2011 comprised of educators from many content areas and educational backgrounds. This team was charged with examining standards, identifying the needs in the field for support, and gathering materials and resources to address those needs.

Wisconsin Foundations for Disciplinary Literacy

To guide understanding and professional learning, a set of foundations, developed in concert with Wisconsin's *Guiding Principles for Teaching and Learning*, directs Wisconsin's approach to disciplinary literacy.

The literacy skills of reading, writing, listening, speaking, and critical thinking improve when content-rich learning experiences motivate and engage students.

Our goals for Wisconsin students are:

1. Demonstrate independence.
2. Build strong content and knowledge.
3. Respond to the varying demands of audience, task, purpose and discipline.
4. Comprehend as well as critique.
5. Value evidence.
6. Use technology and digital media strategically and capably.
7. Come to understand other perspectives and cultures.

Academic learning begins in early childhood and develops across all disciplines.

Each discipline has its own specific vocabulary, text types, and ways of communicating. Children begin learning these context- and content-specific differences early in life and continue through high school and beyond. While gardening, small children observe the form and function of a root, stem, leaf, and soil; or measure, mix, and blend while baking a cake. School offers all students opportunities to develop the ability to, for example, think like a scientist, write like a historian, critique like an artist, problem solve like an auto mechanic, or analyze technological advances like a health care technician. As literacy skills develop, educators gradually shift the responsibility for reading, writing, listening, speaking, and critical thinking to students through guided supports in both individual and collaborative learning experiences.

Content knowledge is strengthened when educators integrate discipline-specific literacy into teaching and learning.

Educators help students recognize and understand the nuances of a discipline by using strategies that "make their thinking visible." They promote classroom reading, writing, listening, speaking, and critical thinking using authentic materials that support the development of content-specific knowledge. They guide students through these complex texts by using strategies that develop conceptual understanding of language and set expectations for relevant application of skills. These literacy practices deepen students' content knowledge, strategies, and skills so that their learning transfers to real-world situations.

"The ability to comprehend written texts is not a static or fixed ability, but rather one that involves a dynamic relationship between the demands of texts and prior knowledge and goals of the reader."

Educators who foster disciplinary literacy develop experiences that integrate rigorous content with relevant collaborative and creative literacy processes to motivate and engage students. Setting high expectations, they structure routines and supports that empower students to take charge of their own learning. When students work in teams to research science and mathematics concepts in the development of an invention or a graphic arts design or when they collaboratively build a blog that contains their recent marketing venture, they use specific literacy skills and strategies to solidify learning. Students need these opportunities over time to develop the precise and complex reading, writing, listening, speaking, and critical thinking skills demanded in today's careers.

Students demonstrate their content knowledge through reading, writing, listening, and speaking as part of a content-literate community. Students who are literate in a particular discipline are able to successfully read, write, and speak about that discipline and can listen to and think critically as others communicate in that community. Performance tasks that allow students to present the complexity of a content area in a way that is meaningful to the field become authentic approaches to assessing mastery within a discipline. Such tasks empower students to discover the real-world connections across disciplines and to actively participate in communities of discipline-literate peers.

What Research and Resources are Available?

The *Wisconsin Academic Standards for Literacy in All Subjects* reflect the importance of literacy in both the oral and written language and in both productive (speaking and writing) and receptive (listening and reading) discourse. Clearly, critical and precise thinking are required to develop all of these specific strategies and skills. The standards also address the learning and functioning of language in a technological, media-driven world because the language that we use is selective depending upon the context of the conversation.

The following section offers relevant research and resources to support professional learning in reading, writing, speaking, listening, and language across disciplines. Collegial conversation and learning, both cross-discipline and within-discipline, will help make the Wisconsin Academic Standards more applicable to schools and districts and will address the needs of unique programs within those contexts. A collection of online resources will continue to develop as support materials emerge.

Reading Connections

While early reading focuses on learning that letters make sounds and that words carry meaning, reading quickly develops to a point where the message taken from text depends on what the reader brings to it. *The Carnegie Report*, *Reading in the Disciplines* (2010), describes this phenomenon:

> Therefore, a musician reading a journal article that describes concepts in music theory will take more information away from the text than a music novice because of their knowledge and experience in music. As well, an individual who spends a significant amount of time reading automotive manuals will more easily navigate a cell phone manual because of familiarity with that type of text.

A chart excerpted from the Carnegie Report (2010) details a few of the generic and more discipline-specific strategies that support students as they attempt to comprehend complex text. While the generic strategies pertain across content areas, discipline-specific ones must be tailored to match the demands of the content area.

Both generic and discipline focused strategies and knowledge must be applied to the comprehension and evaluation of:

- textbooks
- journal and magazine articles
- historically situated primary documents
- full-length books
- newspaper articles
- book chapters
- multimedia and Digital Texts

Generic reading strategies*

- Monitoring comprehension
- Pre-reading
- Setting goals

- Thinking about what one already knows
- Asking questions
- Making predictions
- Testing predictions against the text
- Re-reading
- Summarizing

Discipline-specific reading strategies*

- Building prior knowledge
- Building specialized vocabulary
- Learning to deconstruct complex sentences
- Using knowledge of text structures and genres to predict main and subordinate ideas
- Mapping graphic (and mathematical) representations against explanations in the text
- Posing discipline-relevant questions
- Comparing claims and propositions across texts
- Using norms for reasoning within the discipline (i.e. what counts as evidence) to evaluate claims

*Source: Carnegie Report, (2010)

Additional resources support reading in specific subjects. *Content Counts! Developing Disciplinary Literacy Skills, K–6* by Jennifer L. Altieri (2011) is a guide for discipline-specific literacy at the elementary level and offers strategies to balance the demands of literacy while continuing to make content count and help students meet the reading, writing, speaking and listening demands of the content areas as they advance in school.

A resource by Doug Buehl (2011), *Developing Readers in the Academic Disciplines*, describes what it means to read, write, and think through a disciplinary lens in the adolescent years. This teacher-friendly guide helps connect literacy with disciplinary understandings to bridge academic knowledge gaps, frontload instruction, and build critical thinking through questioning.

Note on range and content of student reading

The *Wisconsin Academic Standards for Literacy in All Subjects* require that "students must read widely and deeply from among a broad range of high-quality, increasingly challenging...text." This type of reading—included in an intentionally developed curriculum—supports students in building a base of content-specific knowledge while developing skills to read increasingly complex text.

Writing Next: Effective Strategies to Improve Writing of Adolescents in Middle and High Schools (2007), detailed research on writing to learn, rather than only for assessment, as having a significant impact on content learning.

Wisconsin uses a three-part model for text complexity, considering qualitative, quantitative, and reader-and-task demands (see https://dpi.wi.gov/reading/professional-learning/text-complexity for more information). In addition, a well-developed collection of complex texts carefully considers representation and diversity, including diversity in the creators and topics of texts.

Writing Connections

The *Wisconsin Academic Standards for Literacy in All Subjects* call for emphasis on three types of writing: narrative, informational, and argument. Writing that presents a logical argument is especially appropriate to discipline-specific work since credible evidence differs across content areas. The ability to consider multiple perspectives, assess the validity of claims, and present a point of view is required in argumentative writing. These thinking and communication skills are "critical to college and career readiness."

The study found writing to learn was equally effective for all content areas in the study (social studies, math, and science) and at every grade (4-12).

Note on range and content of student writing

The *Wisconsin Academic Standards for Literacy in All Subjects* require that students "write routinely over extended time frames (time for research, reflection, and revision) and shorter time frames (a single sitting or a day or two) for a range of discipline-specific tasks, purposes, and audiences." This breadth and depth of writing ensures students are able to flexibly select a format to meet the needs of a specific audience and purpose. This is accomplished through significant amounts of time dedicated to varied writing.

Students must have extensive vocabularies, built through reading and explicit instruction embedded in the context of content learning. This enables them to comprehend complex texts, engage in purposeful writing and communicate effectively within a discipline.

Speaking, Listening, and Language Connections

The ability to share ideas and orally communicate with credibility in a specific academic discourse empowers students and allows access to specialized groups. In Situated Language and Learning: A Critique of Traditional Schooling, James Paul Gee (2004) describes the need to prioritize these skills so that students are at ease as they enter situations connected to a specific content area and are more likely to continue their learning in that discipline.

As expertise develops, students feel more and more comfortable applying knowledge and skills while speaking and listening in a specific discipline.

- A media course may teach students appropriate expression, tone, and rate of speech when addressing a large audience.
- Listening carefully to questions posed is a specialized skill that debate facilitators develop.
- Scientists learn to listen for bias in the perspectives presented by peers to determine the reliability of scientific outcomes.
- Artists have very specialized and specific ways of speaking about the many aspects of a piece.

A policy brief from the Alliance for Excellent Education, *Engineering Solutions to the National Crisis in Literacy* describes “a staircase of literacy demands” and emphasizes the importance of a progressive development of language and literacy over time.

The conceptual understanding of “functions” in mathematics may begin to develop in elementary school in its simplest form. As the concept develops over the years, students will use the word “function” in a meaningful way when speaking and writing to describe the mathematical concept they apply. When educators explicitly connect a mathematical term to its application and repeatedly expose students to the concept connected to the term, a specialized language becomes second nature to the mathematics classroom.

Skills in determining or clarifying the meaning of words and phrases encountered, choosing flexibly from an array of strategies, and seeing an individual word as part of a network of other words that, for example, have similar denotations but different connotations, allow students to access information and support their own learning.

Literacy in Multiple Languages

Increasing economic, security, cross-cultural, and global demands underscore the value of literacy in more than one language. Students who think, read, write, and communicate in multiple languages are an asset to our own country and can more easily interact and compete in the world at large.

English learners in our classrooms face significant challenges as they add a new language and work to grasp content at the same rate as their English-speaking peers. In a report to the Carnegie Corporation, *Double the Work: Challenges and Solutions to Acquiring Academic Literacy for Adolescent English Language Learners (2007)*, researchers found that a focus on academic literacy is crucial for English language learners' success in school. In their description of academic literacy, they include reading, writing, and oral discourse that:

- varies from subject to subject;
- requires knowledge of multiple genres of text, purposes for text use, and text media;
- is influenced by students' literacies in context outside of school; and
- is influenced by students' personal, social, and cultural experiences.

The needs of our English learners are addressed when we embed disciplinary literacy strategies into our subject area teaching. These high impact strategies and skills allow English language learners and all students to more readily access content knowledge and connect it to the prior knowledge they bring to the classroom. When educators take the initiative to understand and embed these strategies and skills, they offer additional opportunities for success to all of our students.

References

Altieri, Jennifer. 2011. *Content Counts! Developing Disciplinary Literacy Skills, K–6*. International Reading Association.

Buehl, Doug. 2011. *Developing Readers in the Academic Disciplines*. International Reading Association.

Carnevale, Anthony, Nicole Smith, and Jeff Strohl. 2013. Recovery: *Job Growth and Education Requirements Through 2020*. Washington, DC: Georgetown University Center on Education and the Workforce.

Fitzsimmons, Shannon, and Deborah J. Short. 2007. *Double the Work: Challenges and Solutions to Acquiring Academic Literacy for Adolescent English Language Learners*. New York: Carnegie Corporation.

Gee, James Paul. 2004. *Situated Language and Learning: A Critique of Traditional Schooling*. New York. Routledge.

Graham, Steve, and Dolores Perin. 2007. *Writing Next: Effective Strategies to Improve Writing of Adolescents in Middle and High Schools.* New York: Carnegie Corporation.

Haynes, Mariana. 2011. *Engineering Solutions to the National Crisis in Literacy: How to Make Good on the Promise of the Common Core State Standards*. Washington, DC: Alliance for Excellent Education.

Lee, Carol, and Anika Spratley. 2010. *Reading in the Disciplines: The Challenges of Adolescent Literacy*. New York: Carnegie Corporation.

State Superintendent's Adolescent Literacy Plan. 2008. Madison, WI: Wisconsin Department of Public Instruction.

Untitled Presentation. 2010. Washington, DC: Georgetown Center on Education and the Workforce. http://www9.georgetown.edu/grad/gppi/hpi/cew/pdfs/CEW_press_conference_ppt.pdf (accessed June 7, 2011)

Vygotsky, Lev S., 1978. *Mind in Society: The Development of Higher Psychological Processes*. 14th edition. Cambridge MA: Harvard University Press.

Section V

Connecting Computer Science to Other Wisconsin Academic Disciplines

Connecting Computer Science

In Wisconsin, the education vision is for every child to graduate college and career ready. To achieve this vision, students must develop the skills to think, read, communicate, and perform in many academic contexts. Since students must develop these specific skills, every educator must consider how students learn in their discipline.

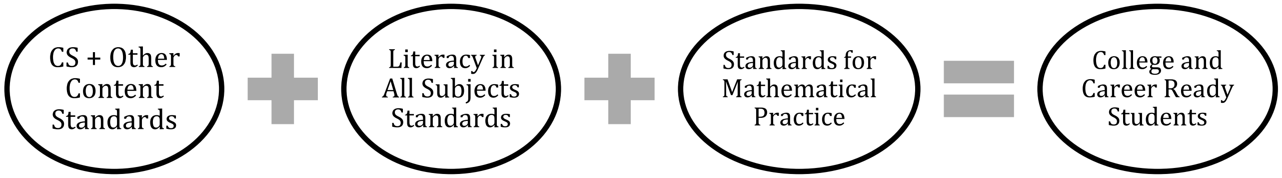
Wisconsin Academic Standards for Computer Science are designed to be focused and coherent, anchored in college and career readiness, and evidence and research-based. The Wisconsin Academic Standards signify the need to change practice in at least three areas: content, instruction, and assessments. Educators in CS must be knowledgeable in how both CS and other Wisconsin standards are addressed in their classrooms. Connections between the Computer Science and other Wisconsin Standards come in two forms.

Making the Connection: Wisconsin Academic Standards and Computer Science Content

1. Integration with disciplinary literacy (literacy in all subjects) and standards for mathematical practice:
    - **Standards and instruction:** The use of reading, writing, speaking, and listening and the application of Standards for Mathematical Practice to build and communicate knowledge specific to computer science.
    - **Assessment:** Standards should be measured through multiple assessments including performance-based assessments.
2. Integration of content and background knowledge where standards from other content areas are embedded:
    - **Standards and instruction:** The use of multiple sets of standards to create relevance of content for students; both computer science and content or standards from other subjects (such as science or mathematics).
    - **Assessment:** Standards should be measured through multiple assessments including performance-based assessments.
    - **Equivalency:** Equivalency shows a one-to-one correlation between Wisconsin State Standards or other content areas such as science and social studies and CTE standards through a state-approved equivalency process in conformity with the Wisconsin State Statute for Equivalency Credit (§ 118.33, Wis. Stats.). This is an option for Career and Technical Education (CTE) courses that prove to have sufficient academic content and are taught in a technical and applied setting.

The Connection

This visual shows the relationship of the *Wisconsin Academic Standards for Computer Science*, *Literacy in all Subjects*, and *Mathematics*. This combination leads toward college and career readiness.



Literacy in All Subjects: The Shift

Disciplinarily literacy—the ability to read, write, listen, speak, think critically, and perform in different ways and for different purposes—begins to develop early and becomes increasingly important as students pursue specialized fields of study. *Wisconsin Academic Standards for Literacy in All Subjects* strive to help students meet the literacy challenges within each particular field of study.

Disciplinary literacy will look different in every classroom based upon the nature of the academic standards addressed within the course and the types of reading, writing, speaking, and listening required to convey knowledge. Students are reading texts to gain knowledge about the discipline; teachers are engaging students with questions and performance tasks; students are writing, composing, or creating.

In Wisconsin, disciplinary literacy is defined as the confluence of content knowledge, experiences, and skills merged with the ability to read, write, listen, speak, think critically, and perform in a way that is meaningful within the context of a given field.

Wisconsin Academic Standards for Literacy in All Subjects identify the specific literacy skills that should be a part of all disciplines. The task, as experts, is to engage students to the authentic literacy activities of the discipline and teach students how to interact with content effectively.

Mathematical Practices: The Shift

The shift in mathematics processes means students are able to transfer math skills and understanding across concepts and grades. Focus allows each student to think, practice and integrate new ideas into a growing knowledge structure. Mathematical proficiency is necessary for every student. Therefore, understanding concepts and being fluent are both important.

This means teaching more than "how to get the answer" and instead support students' ability to access concepts from a number of perspectives while demonstrating conceptual understanding of core math concepts by applying them to new situations. Teachers in content areas outside of math, particularly science and CTE, ensure students are using math at all grade levels to access and make meaning from content. Educators must intentionally engage students at all levels, so they are readily able to apply mathematics in their ever-changing world.

By combining the mathematical practices and CS standards, it allows the teacher to build on students' prior learning from multiple content areas. Students are able to see the relevance of their learning in their chosen career pathway and deepen their learning by transferring skills and concepts.

"When today's students become adults, they will face new demands for mathematical proficiency that school mathematics should attempt to anticipate. Moreover, mathematics is a realm no longer restricted to a select few. All young Americans must learn to think mathematically, and they must think mathematically to learn."

—Adding It Up, National Research Council, 2001

Connecting to Other Content Area Standards

Computer science courses and programs are the quintessential convergence of standards from numerous content areas. Not only do students learn the knowledge and skills necessary for successful transition to college and careers, they also practice and apply their learning in real-life instructional situations that prepare them for specific entry-level careers and postsecondary studies. Along with CS specific standards, students are applying and reinforcing the standards learned in many other areas of study such as, career and technical education, science, arts and creativity, social studies, and mathematics. Educators should consider how standards from other content areas are incorporated into instruction and assessments within CS courses and units.

Reference

Jeremy Kilpatrick, Jane Swafford, and Bradford Findell. *2001. "Adding It Up: Helping Children Learn Mathematics."* National Research Council, Mathematics Learning Study Committee, Center for Education, Division of Behavioral and Social Sciences and Education.