

Oklahoma Academic Standards for Computer Science {High School First Public Draft}

Concept	Subconcept	By the end of 10th Grade	By the end of 12th Grade
Computing Systems	Devices	L1.CS.D.01 Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects.	Not addressed at this level.
	Hardware & Software	L1.CS.HS.01 Explain the interactions between application software, system software, and hardware.	L2.CS.HS.01 Identify and categorize roles of an operating system.
	Troubleshooting	L1.CS.T.01 Develop and apply criteria for systematic discovery of errors and systematic strategies for correction in computing systems.	L2.CS.T.01 Identify how hardware components facilitate logic, input, output and storage in computing systems.
Networks & The Internet	Network Communication & Organization	L1.NI.NCO.01 Evaluate the scalability and reliability of networks by identifying and illustrating the basic components of computer networks (e.g., routers, switches, servers, etc.) and network protocols (e.g., IP, DNS, etc.).	L2.NI.NCO.01 Describe the issues that impact network functionality (e.g., bandwidth, load, delay, topology).
	Cybersecurity	L1.NI.C.01 Compare various security measures by evaluating tradeoffs between the usability and security of a computing system.	L2.NI.C.01 Compare and refine ways in which software developers protect devices and information from unauthorized access.
		L1.NI.C.02 Illustrate how sensitive data can be affected by attacks.	
		L1.NI.C.03 Recommend security measures to address various scenarios based on the principles of information security.	
L1.NI.C.04 Explain tradeoffs when selecting and implementing cybersecurity recommendations from multiple perspectives such as the user, enterprise, and government.	Not addressed at this level.		
Data Analysis	Storage	L1.DA.S.01 Translate and compare different bit representations of real-world phenomena, such as characters, numbers, and images.	Not addressed at this level.
		L1.DA.S.02 Evaluate the tradeoffs in how data is organized and stored.	
	Collection, Visualization, & Transformation	L1.DA.CVT.01 Use tools and techniques to locate, collect and create visualizations of small and large-scale data sets (e.g., paper surveys, online data sets, etc.).	L2.DA.CVT.01 Use data analysis tools and techniques to identify patterns from complex real-world phenomena.
	Not addressed at this level.	L2.DA.CVT.02 Generate data sets that support a claim or communicates information using a variety of data collection tools and analysis techniques.	
Inference & Models	L1.DA.IM.01 Use computational models such as data analysis, pattern recognition, and/or simulations to show the relationships between collected data elements.	L2.DA.IM.01 Use models and simulations to help formulate, refine, and test scientific hypotheses.	
Algorithms	Algorithms	L1.AP.A.01 Use algorithms (e.g., sequencing, selection, iteration, recursion, etc.) to create a prototype to provide a possible solution for a common problem.	L2.AP.A.01 Describe how artificial intelligence drives many software and physical systems (e.g., autonomous robots, computer vision, pattern recognition, text analysis).
		Not addressed at this level.	L2.AP.A.02 Develop an artificial intelligence algorithm to play a game against a human opponent or solve a common problem.
			L2.AP.A.03 Critically examine and adapt classic algorithms (e.g. selection sort, insertion sort, etc).
			L2.AP.A.04 Evaluate algorithms (e.g., sorting, searching) in terms of their efficiency, correctness, and clarity.
	Variables	L1.AP.V.01 Demonstrate the use of lists to simplify solutions, generalizing computations problems instead of repeatedly using simple variables.	L2.AP.V.01 Compare and contrast simple data structures and their uses (e.g. arrays, lists, stacks, queues, maps, trees, graphs, etc.).
	Control	L1.AP.C.01 Justify the selection of specific control structures (e.g. conditionals loops, lists, etc.) when trade-offs involve implementation, readability, and program performance.	L2.AP.C.01 Trace the execution of recursive algorithms, illustrating output and changes in values of named variables.
L1.AP.C.02 Iteratively develop an event-based program that will solve a practical problem, or allow self expression.		Not addressed at this level.	

Oklahoma Academic Standards for Computer Science {High School First Public Draft}

Concept	Subconcept	By the end of 10th Grade	By the end of 12th Grade	
Algorithms & Programming	Modularity	L1.AP.M.01 Using systematic analysis and design, break down a solution into basic elements such as procedures, functions or methods.	L2.AP.M.01 Construct solutions to problems using student-created components, such as procedures, modules and/or objects.	
		L1.AP.M.02 Create computational artifacts by using common structures to organize, manipulate and/or process data.	L2.AP.M.02 Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution.	
		Not addressed at this level.	L2.AP.M.03 Create programming solutions using code reuse (e.g., libraries, APIs, collaboration software, and versioning software) and applied techniques.	
	Program Development	L1.AP.PD.01 Create software by analyzing a problem and/or process, developing a solution, testing outcomes, documenting and adapting the program for a variety of users.	L2.AP.PD.01 Compare multiple programming languages and discuss features that make them useful for solving problems and developing systems.	
		L1.AP.PD.02 Classify and define a variety of software licensing schemes (e.g. open source, freeware, commercial) and discuss the advantages and disadvantages of the different schemes in software development.	L2.AP.PD.02 Using the software life cycle process, create software that will provide solutions to a variety of users.	
		L1.AP.PD.03 Evaluate and refine computational artifacts to make them more user-friendly, efficient and/or accessible.	L2.AP.PD.03 Design software in a project team environment using integrated development environments (IDEs), versioning systems, and collaboration systems.	
		L1.AP.PD.04 Design and develop a computational artifact while working in a team.	L2.AP.PD.04 Explain security issues that might lead to compromised computer programs.	
		L1.AP.PD.05 Using visual aids and documentation, illustrate the design elements and data flow (e.g. flowcharts, pseudocode, etc.) of the development of a complex program.	L2.AP.PD.05 Develop programs for multiple computing platforms.	
		Not addressed at this level.	L2.AP.PD.06 Develop and use a series of test cases to verify that a program performs according to its design specifications.	
		Not addressed at this level.	L2.AP.PD.07 Through peer review systematically check code for correctness, usability, readability, efficiency, portability, and scalability (e.g. code review).	
		Not addressed at this level.	L2.AP.PD.08 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).	
	Impacts of Computing	Culture	L1.IC.C.01 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.	L2.IC.C.01 Evaluate the beneficial and harmful effects that computational artifacts and innovations have on society.
			L1.IC.C.02 Test and refine computational artifacts to reduce bias and equity deficits.	L2.IC.C.02 Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.
L1.IC.C.03 Demonstrate how a given algorithm applies to problems across disciplines.			L2.IC.C.03 Design and implement a study that evaluates or predicts how computation has revolutionized an aspect of our culture and how it might evolve (e.g. education, healthcare, art/entertainment, energy).	
Social Interactions		L1.IC.SI.01 Demonstrate how computing increases connectivity to people in various cultures.	Not addressed at this level.	
		L1.IC.SLE.01 Explain the beneficial and harmful effects that intellectual property laws can have on innovation.	L2.IC.SLE.01 Debate laws and regulations that impact the development and use of software.	
Safety, Law, & Ethics		L1.IC.SLE.02 Explain the privacy concerns related to the collection and large scale analysis of information about individuals (eg. how businesses, social media, and the government collects and uses data) that may not be evident to users.	Not addressed at this level.	
		L1.IC.SLE.03 Evaluate the social and economics implications as related to privacy, data, property, information, and identity in the context of safety, law, or ethics.		