Pearson Education
Emergent Learning, Computer Science: A Problem-Solving Approach, 1e ©2024
Correlation to
OK Academic Standards for Computer Science

High School		
Subconcept	**Ninth and Tenth Grades: Level 1 (L1)**	**Computer Science: A Problem-Solving Approach**
Devices (D)	L1.CS.D.01 Model how abstractions hide the underlying implementation details of computing systems embedded in everyday objects.	pp. 469-470
	Developing and Using Abstractions	
Hardware & Software (HS)	L1.CS.HS.01 Analyze the levels of abstraction and interactions between application software, system software, and hardware.	p. 149; pp. 209-212
	Developing and Using Abstractions	
Troubleshooting (T)	L1.CS.T.01 Develop and apply criteria for the systematic discovery of errors and systematic strategies for the correction of errors in computing systems.	pp. 187-191
	Testing and Refining Computational Artifacts	
Network Communication & Organization (NCO)	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).	p. 268, 533, 594
	Developing and Using Abstractions	
Cybersecurity (CY)	L1.NI.CY.01 Compare physical and cybersecurity measures by evaluating trade-offs between the usability and security of a computing system and the risks of an attack.	pp. 530-534
	Developing and Using Abstractions	
	L1.NI.CY.02 Recommend security measures to address various scenarios based on information security principles.	pp. 529-532
	Recognizing and Defining Computational Problems	
	L1.NI.CY.03 Explain trade-offs when selecting and implementing cybersecurity recommendations from multiple perspectives, such as the user, enterprise, and government.	pp. 529-533
	Communicating about Computing	
Storage (S)	L1.DA.S.01 Convert and compare different bit representations of data types, such as characters, numbers, and images.	pp. 53-57
	Developing and Using Abstractions	
	L1.DA.S.02 Evaluate the trade-offs in how data is organized and stored digitally.	pp. 57-58; 127-129
	Recognizing and Defining Computational Problems	
Collection, Visualization, & Transformation (CVT)	L1.DA.CVT.01 Use tools and techniques to locate, collect, and create visualizations of small and largescale data sets (e.g., paper surveys and online data sets).	pp. 158-165; 219-221

High School		
Subconcept	**Ninth and Tenth Grades: Level 1 (L1)**	**Computer Science: A Problem-Solving Approach**
	Students will continue to apply the standards and practices from the previous grade levels. Additional standards and practices for this	
Inference & Models (IM)	L1.DA.IM.01 Illustrate and explain the relationships between collected data elements using computational models.	pp. 207-208
	Developing and Using Abstractions	
Algorithms (A)	L1.AP.A.01 Create a prototype that uses algorithms (e. g., searching, sorting, finding shortest distance) to provide a possible solution for a real- world problem.	p. 214, 283, 287, 349
	Creating Computational Artifacts	
	Additional standards and practices for this subconcept begin in High School Level 2.	
Variables (V)	L1.AP.V.01 Demonstrate the use of lists (e.g., arrays) to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.	pp. 337-340
	Developing and Using Abstractions	
Control (C)	L1.AP.C.01 Justify the selection of specific control structures (e.g., sequence, conditionals, repetition, procedures) considering program efficiencies such as readability, performance, and memory usage.	p. 117; pp. 249-250
	Recognizing and Defining Computational Problems	
Modularity (M)	L1.AP.M.01 Decompose problems into procedures using systematic analysis and design.	pp. 187-189
	Recognizing and Defining Computational Problems	
	L1.AP.M.02 Create computational artifacts by systematically organizing, manipulating and/or processing data.	p. 115, pp. 189-191; 203
	Recognizing and Defining Computational Problems	
	Additional standards and practices for this subconcept begin in High School Level 2.	
Program Development (PD)	L1.AP.PD.01 Create software that will provide solutions to a variety of users using a software development process.	p. 195; 201; pp. 211-213
	Communicating About Computing	
	L1.AP.PD.02 Evaluate a variety of software licensing schemes (e.g., open source, freeware, commercial) and discuss the advantages and disadvantages of each scheme in software development.	p. 7; pp, 523-525
	Communicating About and Collaborating Around Computing	

High School		
Subconcept	**Ninth and Tenth Grades: Level 1 (L1)**	**Computer Science: A Problem-Solving Approach**
	L1.AP.PD.03 While working in a team, develop, test, and refine event-based programs that solve practical problems or allow self-expression.	pp. 34-36,
	Testing and Refining Computational Artifacts	
	L1.AP.PD.04 Using visual aids and documentation, illustrate the design elements and data flow (e.g., flowcharts, pseudocode) of the development of a complex program.	pp. 159-160; 217, 391
	Communicating About Computing	
	L1.AP.PD.05 Evaluate and refine computational artifacts to make them more user-friendly, efficient and/or accessible.	pp. 226-228; 517
	Testing and Refining Computational Artifacts	
	Additional standards and practices for this subconcept begin in High School Level 2.	
Culture (CU)	L1.IC.CU.01 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.	pp. 44-46
	Communicating About Computing	
	L1.IC.CU.02 Test and refine computational artifacts to ensure access to a variety of user audiences.	pp. 195-196
	Developing a Productive Computing Environment	
	L1.IC.CU.03 Demonstrate ways a given algorithm can help solve computational problems across disciplines.	p. 113; 181; 201
	Recognizing and Defining Computational Problems	
Social Interactions (SI)	L1.IC.SI.01 Demonstrate and debate how computing increases and decreases connectivity and communication among people of various cultures.	pp. 30-31
	Collaborating Around Computing	
Internet Safety, Law, & Ethics (SLE)	L1.IC.SLE.01 Describe the beneficial and harmful effects that intellectual property laws can have on innovation.	p. 525
	Communicating About Computing	
	L1.IC.SLE.02 Describe and discuss the privacy concerns related to the large-scale collection and analysis of information about individuals (e.g., how websites collect and uses data) that may not be evident to users.	pp. 530-531
	Communicating About Computing	
	L1.IC.SLE.03 Evaluate the social and economic consequences of how law and ethics interact with digital aspects of privacy, data, property, information, and identity.	pp. 44-46
	Communicating About Computing	

Pearson Education
Emergent Learning, Computer Science: A Problem-Solving Approach, 1e ©2024
Correlation to
OK Academic Standards for Computer Science

High School		
Subconcept	**Eleventh and Twelfth Grades: Level 2 (L2)**	**Computer Science: A Problem-Solving Approach**
Devices (D)	Students will continue to apply the standards and practices from the previous grade levels.	
Hardware & Software (HS)	L2.CS.HS.01 Identify and categorize the roles of a variety of operating system software.	pp. 7-11
	Communicating About Computing	
Troubleshooting (T)	L2.CS.T.01 Illustrate how understanding the ways hardware components facilitate logic, input, output, and storage in computing systems will support troubleshooting.	p. 6; 16; 19; 461
	Communicating About Computing	
Network Communication & Organization (NCO)	L2.NI.NCO.01 Describe the issues that impact network functionality (e.g., bandwidth, load, latency, topology).	p. 45
	Communicating About Computing	
Cybersecurity (CY)	L2.NI.CY.01 Compare and refine ways in which software developers protect devices and information from unauthorized access.	pp. 530-531
	Communicating about Computing	
Storage (S)	Students will continue to apply the standards and practices from the previous grade levels.	
Collection, Visualization, & Transformation (CVT)	L2.DA.CVT.01Use data analysis tools and techniques to identify patterns from complex real-world data.	pp. 54-56; 335
	L2.DA.CVT.02 Generate data sets that use a variety of data collection tools and analysis techniques to support a claim and/or communicate information.	pp. 461-462
	Communicating About Computing	
	L2.DA.IM.01 Use models and simulations to help plan, conduct, and refine investigations.	p. 423; 481
Inference & Models (IM)	L2.AP.A.01 Model and use appropriate terminology to describe how artificial intelligence algorithms drive many software and physical systems (e.g., autonomous robots, pattern recognition, text analysis).	pp. 85-86; 190
	Communicating About Computing	
Algorithms (A)	L2.AP.A.02 Develop an artificial intelligence algorithm to play a game against a human opponent or solve a real-world problem.	pp. 456-457
	Creating Computational Artifacts	
	L2.AP.A.03 Critically examine and trace classic algorithms (e.g., selection sort, insertion sort, binary search, linear search).	p. 355;594
	Developing and Using Abstractions	

High School		
Subconcept	**Eleventh and Twelfth Grades: Level 2 (L2)**	**Computer Science: A Problem-Solving Approach**
	L2.AP.A.04 Evaluate algorithms (e.g., sorting, searching) in terms of their efficiency and clarity.	pp. 194-195; pp. 355-356
	Developing and Using Abstractions	
Variables (V)	L2.AP.V.01 Compare and contrast data structures and their uses (e.g., lists, stacks, queues).	pp. 337-339
Control (C)	L2.AP.C.01 Model the execution of repetition (e.g., loops, recursion) of an algorithm illustrating output and changes in values of named variables.	pp. 272-273;
Modularity (M)	L2.AP.M.01 Construct solutions to problems using student-created components (e.g., procedures, modules, objects).	p. 178; 187-188
	Creating Computational Artifacts	
	L2.AP.M.02 Design or redesign a solution to a large-scale computational problem by identifying generalizable patterns.	pp. 187-188
	Developing and Using Abstractions	
	L2.AP.M.03 Create programming solutions by reusing existing code (e.g., libraries, Application Programming Interface (APIs), code repositories).	pp. 189-191
	Creating Computational Artifacts	
Program Development (PD)	L2.AP.PD.01 Create software that will provide solutions to a variety of users using multiple software development processes.	pp. 206-207; 451
	Creating Computational Artifacts	
	L2.AP.PD.02 Design software in a project team environment using integrated development environments (IDEs), versioning systems, and collaboration systems.	pp. 88-89; 153
	L2.AP.PD.03 Develop programs for multiple computing platforms.	pp. 209-212
	Creating Computational Artifacts	
	L2.AP.PD.04 Systematically examine code for correctness, usability, readability, efficiency, portability, and scalability through peer review.	pp. 439-440
	Testing and Refining Computational Artifacts	
	L2.AP.PD.05 Develop and use a series of test cases to verify that a program performs according to its design specifications.	pp; 388-389

Pearson Education
Emergent Learning, Computer Science: A Problem-Solving Approach, 1e ©2024
Correlation to
OK Academic Standards for Computer Science

	High School	
Subconcept	**Eleventh and Twelfth Grades: Level 2 (L2)**	**Computer Science: A Problem-Solving Approach**
	L2.AP.PD.06 Explain security issues that might lead to compromised computer programs.	p. 537
	Communicating About Computing	
	L2.AP.PD.07 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).	pp. 228-229
	Creating Computational Artifacts	
Culture (CU)	L2.IC.CU.01 Evaluate the beneficial and harmful effects that computational artifacts and innovations have on society.	pp. 226-228; 517
	L2.IC.CU.02 Evaluate the impact of location and user audience on the distribution of computing resources in a global society.	pp. 45-46
	L2.IC.CU.03 Design and implement a study that evaluates or predicts how creating, testing, and refining computational artifacts has revolutionized an aspect of our culture and how it might evolve (e.g., education, healthcare, art/entertainment, energy).	pp. 226-228; 517
Social Interactions (SI)	Students will continue to apply the standards and practices from the previous grade levels.	
Internet Safety, Law, & Ethics (SLE)	L2.IC.SLE.01 Debate laws and regulations that impact the development and use of software.	pp. 524-525; 537
	Communicating About Computing, Recognizing and Defining Computational Problems	