

## CompuScholar, Inc.

### Alignment to Alabama Digital Literacy and Computer Science Standards

9th - 12th grades

#### Alabama Course Details:

<b>Course Title:</b>	Digital Literacy and Computer Science
<b>Grade Level:</b>	9th - 12th grades
<b>Standards Link:</b>	<a href="#">2018 Textbook Draft DL and CS COS.pdf</a>

#### CompuScholar Course Details:

<b>Course Title:</b>	CompuScholar: Windows Programming with C#
<b>Course ISBN:</b>	978-0-9887070-0-9
<b>Course Year:</b>	2018

**Note 1:** Citation(s) listed may represent a subset of the instances where objectives are met throughout the course.

**Note 2:** Citation(s) refer to the "Lesson Text" elements within the course, unless otherwise noted. The course "Instructional Video" components are supplements designed to introduce or re-enforce the main lesson concepts, and the Lesson Text contains full details.

### Course Description

9th - 12th grades content for Digital Literacy and Computer Science is organized into five strands of focused study. CompuScholar's "Windows Programming with C#" introductory programming course meets these standards while teaching students fundamentals of the C# language.

### Course Standards

Computational Thinker	CITATION(S)
<b>Abstraction</b>	
1. Decompose problems into component parts, extract key information, and develop descriptive models to understand the levels of abstractions in complex systems.	Chapter 7, Lesson 3 Chapters 12, 13, 16, 17
2. Explain how computing systems are often integrated with other systems and embedded in ways that may not be apparent to the user. Examples: A medical device can be embedded inside a patient to monitor and regulate his or her health; millions of lines of code control the subsystems within an automobile (e.g., antilock braking systems, lane detection, and self-parking).	Chapter 1, Lesson 2 Chapter 2, Lesson 3

<b>Algorithms</b>	
3. Differentiate between a generalized expression of an algorithm in pseudocode and its concrete implementation in a programming language.	Chapter 7, Lesson 3
a. Explain that some algorithms do not lead to exact solutions in a reasonable amount of time and thus approximations are acceptable.	Chapter 7, Lesson 3 Chapter 14, Lesson 1
b. Compare and contrast the difference between specific control structures (sequential statements, conditional, iteration) and explain the benefits and drawbacks of choices made. Examples: Tradeoffs involving implementation, readability, and program performance.	Chapter 5, Lessons 2, 3, 4
c. Understand when a problem solution requires decisions to be made among alternatives, such as conditional “if” constructs, or when a solution needs to be iteratively processed to arrive at a result, such as iterative “loop” constructs or recursion.	Chapter 5, Lessons 2, 3, 4 Chapter 14, Lesson 2
d. Evaluate and select algorithms based on performance, reusability, and ease of implementation.	Chapter 14, Lessons 1, 3
e. Explain how more than one algorithm may solve the same problem and yet be characterized with different priorities. Examples: All self-driving cars have a common goal of taking a passenger to a designation but may have different priorities such as safety, speed, or conservation; web search engines have their own algorithms for search with their own priorities.	Chapter 14, Lessons 1, 3
4. Use and adapt classic algorithms to solve computational problems. Examples: Sorting, searching, shortest path, and data compression.	Chapter 7, Lesson 3 Chapter 14, Lessons 1, 2, 3
<b>Programming and Development</b>	
5. Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using current events.	Chapters 16, 17, 18
6. Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects, with parameters, and which return a result.	Chapters 9, 12, 13, 16, 17
7. Compare and contrast fundamental data structures and their uses. Examples: Strings, lists, arrays, stacks, queues.	Chapter 4, Lessons 1, 2, 4 Chapter 11, Lessons 1, 2 Supplemental Chapter 1, Lesson 8
8. Demonstrate code reuse by creating programming solutions using libraries and Application Programming Interfaces (APIs).	Chapter 3, Lesson 4 Chapter 6 Chapter 7, Lesson 2
9. Demonstrate the ability to verify the correctness of a program.	Chapter 10, Lesson 4 Chapter 18, Lesson/Activity 4

a. Develop and use a series of test cases to verify that a program performs according to its design specifications	Chapter 10, Lesson 4 Chapter 18, Lesson/Activity 4
b. Collaborate in a code review process to identify correctness, efficiency, scalability and readability of program code.	Chapter 10, Lesson 4 Chapter 18, Lesson/Activity 4
10. Resolve errors encountered (debug) during testing using iterative design process. Examples: Test for infinite loops, check for bad input, check edge-cases.	Chapter 10, Lesson 4 Chapter 18, Lesson/Activity 4

<b>Citizen of a Digital Culture</b>	<b>CITATION(S)</b>
<b>Safety, Privacy, and Security</b>	
11. Model and demonstrate behaviors that are safe, legal, and ethical while living, learning, and working in an interconnected digital world.	Chapter 1, Lessons 5, 6
a. Recognize user tracking methods and hazards. Examples: Cookies, WiFi packet sniffing.	Supplemental Chapter 3, Lesson/Activity 1
b. Understand how to apply techniques to mitigate effects of user tracking methods.	Supplemental Chapter 3, Lesson/Activity 1
c. Understand the ramifications of end-user license agreements and terms of service associated with granting rights to personal data and media to other entities.	Chapter 1, Lesson 5
d. Explore the relationship between online privacy and personal security. Examples: Convenience and accessibility, data mining, digital marketing, online wallets, theft of personal information.	Chapter 1, Lesson 6 Supplemental Chapter 3, Lesson/Activity 1
e. Identify consequences (physical, mental, legal, and ethical) of inappropriate digital behaviors. Examples: Cyberbullying/harassment, inappropriate sexual communications.	Chapter 1, Lessons 5, 6
f. Explain strategies to lessen the impact of negative digital behaviors and assess when to apply them.	Chapter 1, Lesson 5
12. Describe how sensitive data can be affected by malware and other attacks.	Chapter 1, Lessons 5, 6
13. Compare various security measures of a computer system. Examples: Usability, security, portability, and scalability.	Chapter 1, Lesson 6
14. Compare ways to protect devices, software, and data.	Chapter 1, Lesson 6
<b>Legal and Ethical Behavior</b>	
15. Explain the necessity for the school's Acceptable Use Policy (AUP).	Chapter 1, Lesson 5
16. Identify laws regarding the use of technology and their consequences and implications. Examples: Unmanned vehicles, net neutrality/common carriers, hacking, intellectual property, piracy, plagiarism.	Chapter 1, Lessons 5, 6

17. Discuss the ethical ramifications of malicious hacking and its impact on society. Examples: Dissemination of privileged information, ransomware.	Chapter 1, Lessons 5, 6
18. Explain the beneficial and harmful effects that intellectual property laws can have on innovation.	Chapter 1, Lesson 5
<b>Digital Identity</b>	
19. Prove that digital identity is a reflection of persistent, publicly available artifacts.	Supplemental Chapter 3, Lesson/Activity 1
20. Evaluate strategies to manage digital identity and reputation with awareness of the permanent impact of actions in a digital world.	Supplemental Chapter 3, Lesson/Activity 1
<b>Impact of Computing</b>	
21. Explain how technology facilitates the disruption of traditional institutions and services. Examples: Digital currencies, ridesharing, autonomous vehicles, retail, Internet of Things (IoT).	Supplemental Chapter 3, Lesson/Activity 2
22. Research the impact of computing technology on possible career pathways. Examples: Government, business, medicine, entertainment, education, transportation.	Supplemental Chapter 3, Lesson/Activity 2
23. Debate the positive and negative effects of computing innovations in personal, ethical, social, economic, and cultural spheres. Examples: AI/machine learning, mobile applications, automation of traditional occupational skills.	Supplemental Chapter 3, Lesson/Activity 2

<b>Global Collaborator</b>	<b>CITATION(S)</b>
<b>Creative Communication</b>	
24. Compare and contrast internet publishing platforms, including suitability for media types, target audience, and feedback mechanism.	Supplemental Chapter 3, Lesson 4
a. Apply version control capabilities within a digital tool to understand the importance of managing historical changes across suggestions made by a collaborative team.	Chapter 18, Lesson/Activity 4
<b>Digital Tools</b>	
25. Utilize a variety of digital tools to create digital artifacts across content areas.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6 Supplemental Chapter 3, Lesson 4

<b>Collaborative Research</b>	
26. Use collaborative technologies to work with others including peers, experts, or community members to examine local, national, and global issues and problems from multiple viewpoints.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6 Supplemental Chapter 3, Lesson 4
<b>Social Interactions</b>	
27. Apply tools and methods for collaboration on a project to increase connectivity among people in different cultures and career fields. Examples: Collaborative documents, webinars, teleconferencing, and virtual fieldtrips	Chapter 18 (Open Team Project) Supplemental Chapter 3, Lesson 4

<b>Computing Analyst</b>	<b>CITATION(S)</b>
<b>Data</b>	
28. Develop a model that reflects the methods, procedures and concepts used by computing devices in translating digital bits as real world phenomena, such as print characters, sound, images, and video.	Chapter 4, Lesson 5 Supplemental Chapter 1, Lesson/Activity 1
29. Summarize the role of compression (encode/decode) and encryption (encrypt/decrypt) in modifying the structure of digital artifacts and the varieties of information carried in the metadata of these artifacts.	Supplemental Chapter 1, Lesson/Activity 1
30. Evaluate the tradeoffs involved in choosing methods for the organization of data elements and the location of data storage, including the advantages and disadvantages of networked computing. Examples: Client server, peer-to-peer, cloud computing.	Supplemental Chapter 1, Lesson 2
31. Create interactive data visualizations using software tools to help others understand real-world phenomena.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6
32. Use data analysis tools and techniques to identify patterns in data representing complex systems.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6
<b>Systems</b>	
33. Evaluate the scalability and reliability of networks by describing the relationship between routers, switches, servers, topology, and addressing, as well as the issues that impact network functionality. Examples: Bandwidth, load, delay, topology	Supplemental Chapter 1, Lesson 2
a. Explain the purpose of Internet Protocol (IP) addresses and how domain names are resolved to IP addresses through a Domain Name System (DNS) server.	Supplemental Chapter 1, Lesson 3

b. Understand the need for networking protocols and examples of common protocols. Examples: HTTP, SMTP, and FTP	Supplemental Chapter 1, Lesson 4
34. Categorize the roles of operating system software.	Chapter 1, Lesson 3
35. Appraise the role of artificial intelligence in guiding software and physical systems. Examples: Predictor systems, self-driving cars.	Supplemental Chapter 3, Lesson/Activity 3
36. Explain the tradeoffs when selecting and implementing cybersecurity recommendations. Examples: Two-factor authentication, password requirements, geolocation requirements.	Chapter 1, Lesson 6
<b>Modeling and Simulation</b>	
37. Evaluate the ability of models and simulations to test and support the refinement of hypotheses.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6
a. Create and utilize models and simulations to help formulate, test, and refine a hypothesis.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6
b. Form a model of a hypothesis, testing the hypothesis by the collection and analysis of data generated by simulations. Examples: Science lab, robotics lab, manufacturing, space exploration.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6
c. Explore situations where a flawed model provided an incorrect answer.	Chapter 18 (Open Team Project) Supplemental Chapter 1, Lessons/Activities 5, 6

<b>Innovative Designer</b>	<b>CITATION(S)</b>
<b>Human/Computer Partnerships</b>	
38. Systematically design and develop programs for broad audiences by incorporating feedback from users. Examples: Games, utilities, mobile applications.	Chapter 18 (Open Project) Supplemental Chapter 2, Lesson / Activity 1
39. Identify a problem that cannot be solved by either humans or machines alone and discuss a solution for it by decomposing the task into sub-problems suited for a human or machine to accomplish. Examples: A human/computer team playing chess, forecasting weather, piloting airplanes.	Supplemental Chapter 3, Lesson/Activity 3
<b>Design Thinking</b>	
40. Use an iterative design process, including learning from mistakes, to gain a better understanding of a problem domain.	Chapter 18 (Open Project) Supplemental Chapter 1, Lessons/Activities 5, 6

Recurring Standards, All Grades	CITATION(S)
<b>Safety, Privacy, and Security</b> - 1. Identify, demonstrate, and apply personal safe use of digital devices.	Chapter 1, Lesson 6 Supplemental Chapter 3, Lesson 1
<b>Legal and Ethical Behavior</b> - 2. Recognize and demonstrate age-appropriate responsible use of digital devices and resources as outlined in school/district rules.	Chapter 1, Lesson 5
<b>Impact of Computing</b> - 3. Analyze the potential impact of computing.	Chapter 1, Lesson 1 Supplemental Chapter 3, All Lessons
<b>Systems</b> - 4. Identify and employ appropriate troubleshooting techniques used to solve computing or connectivity issues.	Chapter 10, All Lessons Chapter 18, Lesson 4
<b>Collaborative Research</b> - 5. Locate, curate, and evaluate information from digital sources to answer research questions.	Chapter 18, Activity 1 Supplemental Chapter 2, Activities 2, 4 Supplemental Chapter 3, Activities 2, 3
<b>Digital Tools</b> - 6. Produce, review, and revise authentic artifacts using appropriate digital tools.	Hands-on coding exercises produce digital applications in most chapters, including team project in Chapter 18. See also: Supplemental Chapter 2, Lessons / Activities 1, 2, 4 Supplemental Chapter 3, Lesson / Activity 4