

CSTA STANDARDS CORRELATIONS for CCR ADVANCED COMPUTER SCIENCE COURSES

Theme	Game Development & Advanced Game Development with Unreal Engine	App Development & Advanced App Development with Android Studio
Computing Systems Includes Devices, Hardware/Software, & Troubleshooting		
Networks & the Internet Includes Network Organization/Communication & Cybersecurity		
Data & Analysis Includes Storage, Collection/Visualization/Transformation, & Inference/Models		
Algorithms & Programming Includes Algorithms, Variables, Control, Modularity, & Program Development	<p>3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.</p> <p>3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p>3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.</p> <p>3A-AP-16 Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p>3A-AP-17 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.</p> <p>3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.</p> <p>3A-AP-19 Systematically design and develop programs for broad audiences by incorporating feedback from users.</p> <p>3A-AP-21 Evaluate and refine computational artifacts to make them more usable and accessible.</p> <p>3A-AP-22 Design and develop computational artifacts working in team roles using collaborative tools.</p> <p>3A-AP-23 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.</p> <p>3B-AP-11 Evaluate algorithms in terms of their efficiency, correctness, and clarity.</p> <p>3B-AP-13 Illustrate the flow of execution of a recursive algorithm.</p> <p>3B-AP-16 Demonstrate code reuse by creating programming solutions using libraries and APIs.</p> <p>3B-AP-17 Plan and develop programs for broad audiences using a software life cycle process.</p> <p>3B-AP-20 Use version control systems, integrated development environments (IDEs), and collaborative tools and practices (code documentation) in a group software project.</p> <p>3B-AP-21 Develop and use a series of test cases to verify that a program performs according to its design specifications.</p> <p>3B-AP-22 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).</p> <p>3B-AP-23 Evaluate key qualities of a program through a process such as a code review.</p>	<p>3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.</p> <p>3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.</p> <p>3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.</p> <p>3A-AP-16 Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</p> <p>3A-AP-17 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.</p> <p>3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.</p> <p>3A-AP-19 Systematically design and develop programs for broad audiences by incorporating feedback from users.</p> <p>3A-AP-21 Evaluate and refine computational artifacts to make them more usable and accessible.</p> <p>3A-AP-22 Design and develop computational artifacts working in team roles using collaborative tools.</p> <p>3A-AP-23 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.</p> <p>3B-AP-11 Evaluate algorithms in terms of their efficiency, correctness, and clarity.</p> <p>3B-AP-17 Plan and develop programs for broad audiences using a software life cycle process.</p> <p>3B-AP-22 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).</p> <p>3B-AP-23 Evaluate key qualities of a program through a process such as a code review.</p>
Impacts of Computing Includes Culture, Social Interactions, & Safety/Law/Ethics	<p>3A-IC-24 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.</p> <p>3A-IC-27 Use tools and methods for collaboration on a project to increase connectivity of people in different cultures and career fields.</p> <p>3A-IC-29 Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.</p>	<p>3A-IC-24 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.</p> <p>3A-IC-27 Use tools and methods for collaboration on a project to increase connectivity of people in different cultures and career fields.</p> <p>3A-IC-29 Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.</p>

from CSTA: Level 3A standards are recommended for all students before the end of high school; Level 3B standards are for students who wish to pursue a CS career or education beyond high school.