South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
Computing Systems			
5.CS.1.1 Select the appropriate computing device for an application (e.g., writing an essay on a laptop versus on a smartphone).			
5.CS.1.2 Explain the importance of the major hardware components of the computing device (e.g., input and output devices, processors).			
5.CS.2.1 Justify the use of different computing devices for relevant tasks.			
5.CS.2.2 Explore and compare multiple software applications (e.g., word processor, spreadsheet, presentation software, web browser).			
5.CS.3.1 Respond appropriately to various error messages (e.g., “webpage not found;” “incorrect password”).			
5.CS.3.2 Identify the computing device components that may cause various problems.			
Networks & the Internet			
5.NI.1.1 Identify the advantages and disadvantages of various network types (e.g., wired, Wi-Fi, cellular data).			
5.NI.2.1 Recognize video conferencing as a communication avenue.			
5.NI.2.2 Modify search criteria and use advanced search tactics to improve search results.			
5.NI.2.3 Utilize websites that are appropriate sources of research.			
Data and Analysis			
5.DA.1.1 Save and retrieve files on computing devices.			
5.DA.1.2 Recognize how text, images, and sounds are represented as binary numbers in computing devices.			
5.DA.2.1 Compare and contrast tools for collecting data.			
5.DA.2.2 Determine the most effective way to represent a given data set (e.g., bar graphs, line plots).			
5.DA.3.1 Compare and contrast models (e.g., graphs, tables) for data analysis.			
5.DA.4.1 Discuss accuracy based on data available.			
Algorithms and Programming			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
5.AP.1.1 Execute a sequence of instructions (i.e., algorithm) that mimic a daily task.			
5.AP.2.1 Recognize that a sequence of steps can be repeated.	[1]		
5.AP.2.2 Identify the result of a conditional statement (e.g., in the statement, “If it is dark, then turn on the light,” the result is the lights turning on).	[2]		
5.AP.3.1 Compose multiple levels of simple tasks (e.g., eating breakfast can include going to the table, sitting down in a chair, and picking up a spoon; brushing your teeth; walking to the bus stop) to make a more complex task.			
5.AP.3.2 Decompose a complex task of higher complexity (e.g., cooking a meal) into simple tasks (e.g., selecting a recipe, getting the ingredients, preparing the food, and serving the meal, where the task of getting the ingredients can be decomposed into writing a shopping list, going to a store, selecting and buying the ingredients, and going home).	[3]		
5.AP.4.1 Use a visual language to design and test a program that solves a simple task (e.g., online coding activity).	[4]		
Impacts of Computing			
5.IC.1.1 Discuss the positive and negative impacts of computing on society.			
5.IC.2.1 Demonstrate an understanding of the relevance and appropriateness of various electronic information sources (e.g., online databases such as Discus; web search engines).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
Computing Systems			
6.CS.1.1 Identify and describe the key functional components (e.g., input devices, output devices, processor, operating system, software applications, memory, storage) of a computer.			
6.CS.1.2 Identify relevant problems and how they are solved using computer science and various types of computing devices(e.g., directions to a location can be obtained through Global Position Systems (GPS) and/or online maps).			
6.CS.2.1 Understand various ways software is acquired and installed.			
6.CS.3.1 Identify the source of a problem using a systematic process (i.e., troubleshooting).	[5]		
Networks & the Internet			
6.NI.1.1 Identify and define hardware required to connect to a network (e.g., connect a school tablet or computer to Wi-Fi, network, or internet).			
6.NI.1.2 Define an IP address and show an example.			
6.NI.1.3 Identify a Uniform Resource Locator (URL).			
6.NI.1.4 Define a packet and explain how they are used to transmit data across a network.			
6.NI.2.1 Identify common security risks associated with using computer networks (e.g., compromised passwords, phishing, viruses).			
6.NI.2.2 Identify how individuals and organizations protect data and information from security risks associated with using computer networks.			
Data and Analysis			
6.DA.1.1 Identify the file extensions (e.g., .ppt, .pdf, .mp3) associated with software programs.			
6.DA.2.1 Explore real-world data collection (e.g., identification number at lunch; teacher taking attendance; grocery store shopping card).			
6.DA.3.1 Explain how large data sets are represented graphically (e.g., frequency plots, bar graphs).			
6.DA.3.2 Represent one set of numerical data (e.g., histograms, box plots, dot plots).			
Algorithms and Programming			
6.AP.1.1 Recognize that there are multiple ways to sequence instructions that can lead to the same result.			
6.AP.1.2 Interpret pseudocode and flowcharts.	[6]		

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
6.AP.2.1 Select appropriate coding control structures to skip or repeat instructions.	[7]		
6.AP.3.1 Discuss the parts of a program (e.g., components of creating a video game include keeping score, determining winners/losers, moving characters, designing game art, and advancing levels).	[8]		
6.AP.4.1 Use a beginner coding language (e.g., drag-and-drop, block-based) to design and code a simple program that solves a problem	[9]		
6.AP.5.1 Recognize variables that represent information (e.g., age, first name).	[10]		
6.AP.5.2 Recognize variables can represent different types of data (e.g., numbers, words, colors, images).	[11]		
Impacts of Computing			
6.IC.1.1 Explore how computer science is and can be used to solve problems in students' daily lives (e.g., "Internet of Things," smart appliances, smart cars).	[12]		
6.IC.1.2 Discover positive and negative impacts of computing on society (e.g., personal, health, workforce, economy, education, culture, environment).			
6.IC.2.1 Identify current communication methods and computing devices.			
6.IC.3.1 Identify guidelines for safely using the internet.			
6.IC.4.1 Identify unethical and illegal behavior.			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
Computing Systems			
7.CS.1.1 Explore an expanded definition of computing devices (e.g., "Internet of Things," wearable technology, robotics).	[13]		
7.CS.1.2 Analyze relevant problems and how they are solved using computer science and various types of computing devices (e.g., Global Positioning System (GPS) and online maps provide guided step-by-step directions to locations).			
7.CS.2.1 Describe processing speed and storage capacity using standard units of measure (e.g., 3 TB hard drive, 256 GB cell phone, 3.8 GHz processor).			
7.CS.3.1 Understand and communicate solutions to various computing problems (e.g., computing device is frozen; webpage does not load; application does not launch; keyboard does not work).			
7.CS.3.2 Understand how rebooting a computing device can solve problems.			
Networks & the Internet			
7.NI.1.1 Identify and compare types of networks (i.e., Local Area Networks (LANs) and Wide Area Networks (WANs)).			
7.NI.1.2 Define and understand how the internet is a network of Wide Area Networks (WANs).			
7.NI.1.3 Compare and contrast network topologies (e.g., ring, star, mesh).			
7.NI.2.1 Identify software methods for protecting data transmitted across networks (e.g. anti-virus software).			
7.NI.2.2 Identify physical methods for securing computing devices (e.g., biometric-thumb reader, computer lock, restricted access rooms, hardware firewall).			
Data and Analysis			
7.DA.1.1 Describe how a picture, audio, and video are stored digitally (e.g., Red, Green, and Blue (RGB), pixels, .wav).			
7.DA.2.1 Identify computing devices that assist with data collection (i.e., thermometers, barcode scanners, biometrics, sensors, radio-frequency identification (RFID), wearable technology).			
7.DA.3.1 Create various graphical representations of large data sets (e.g., frequency plots, bar graphs, presentation software).			
7.DA.3.2 Represent two sets of numerical data (e.g., histograms, box plots, dot plots).			
Algorithms and Programming			
7.AP.1.1 Write sequences of instructions for others to perform tasks.	[14]		

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
7.AP.1.2 Suggest changes to the sequence of instructions that can lead to the same result (e.g., explore different ways to tying shoes).			
7.AP.1.3 Write clear instructions using pseudocode.	[15]		
7.AP.2.1 Write code using control structures to skip or repeat instructions.	[16]		
7.AP.3.1 Decompose a problem into smaller parts.	[17]		
7.AP.3.2 Identify the parts of a program (e.g., components of creating a video game include keeping score, determining winners/losers, moving characters, designing game art, and advancing level).	[18]		
7.AP.4.1 Use a beginner coding language (e.g., drag-and-drop, block-based) to design and code a moderately complex program that solves a problem.	[19]		
7.AP.5.1 Identify variables as a representation for information.	[20]		
7.AP.5.2 Discuss the differences between the types of data (e.g., characters, integers, decimals).	[21]		
Impacts of Computing			
7.IC.1.1 Understand how computer science is and can be used to solve problems in students' daily lives (e.g., voter identification website, online tax filing).	[22]		
7.IC.1.2 Compare positive and negative impacts of computing on society (e.g., personal, health, workforce, economy, education, culture, environment).			
7.IC.2.1 Describe current communication methods and computing devices.			
7.IC.3.1 Understand precautions to protect personal information (i.e., password strength, anti-virus software).			
7.IC.4.1 Understand the consequences of unethical and illegal behavior online (e.g., social media, gaming, cyberbullying).			
7.IC.5.1 Discuss and understand factors that affect access to computing resources locally, nationally, and globally (e.g., geographical location, socioeconomic status, government structure).			
7.IC.6.1 Explain how computer science plays a role in every industry.			
7.IC.7.1 Understand and communicate the changes in computing and computer science over time.			
7.IC.7.2 Understand and communicate the history and development of the internet.			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
Computing Systems			
8.CS.1.1 Compare and contrast relevant problems and how they are solved using computer science and various types of computing devices (e.g., Global Positioning System (GPS) and online maps include different features, including real-time traffic, satellite images, construction and accident notifications).			
8.CS.2.1 Understand that computers receive and process data as a series of on and off signals (i.e., binary data).	[23]		
8.CS.2.2 Determine appropriate hardware, operating systems, and software based upon the needs of users in various career fields (e.g., computing devices used by professional video producers and students differ).			
8.CS.3.1 Understand computer hardware and software compatibility (e.g., applications designed for Android devices cannot run on iOS devices).			
8.CS.3.2 Identify appropriate resources for troubleshooting hardware and software problems (e.g., user manuals, online searches, technology support services).			
Networks & the Internet			
8.NI.1.1 Identify a protocol as a set of rules, and identify common protocols (e.g., Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Internet Protocol (IP), Transmission Control Protocol (TCP)).			
8.NI.1.2 Diagram a small network using a switch and a router.			
8.NI.1.3 Identify the best network topology given a problem (e.g., mesh, tree, ring).			
8.NI.2.1 Discuss and understand recent events and trends regarding cybercrimes (i.e., identity theft, hacking).			
8.NI.2.2 Discuss and understand the impact of computing copyright issues (i.e., music and software piracy; plagiarism).			
Data and Analysis			
8.DA.1.1 Discuss how text, images, and sounds are represented using binary numbers in computing devices.			
8.DA.1.2 Compare and contrast characteristics of a variety of file formats (e.g., software compatibility, file size, compressed and uncompressed files, transparency).			
8.DA.1.3 Compare and contrast current storage mediums and their uses (e.g., flash drives, hard drives, networks, cloud).			
8.DA.2.1 Compare and contrast computing devices that assist with data collection (i.e., thermometers, barcode scanners, biometrics, sensors, radio-frequency identification (RFID), wearable technology).			
8.DA.3.1 Identify components of infographics that can be used to represent numerical data (e.g., scatterplots).			
8.DA.3.2 Make inferences based on collected data (e.g., online video watching history used to recommend new videos users may like).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
8.DA.3.3 Explain how models are used to predict specific behaviors and/or outcomes (e.g., weather data presented in a model used to predict future weather conditions and activity).			
Algorithms and Programming			
8.AP.1.1 Modify a sequence of instructions to solve problems.			
8.AP.1.2 Make changes to the sequence of instructions that can lead to the same result.			
8.AP.1.3 Write clear instructions using flowcharts.	[24]		
8.AP.2.1 Modify an algorithm using conditionals and iteration.	[25]		
8.AP.3.1 Decompose a problem into functional parts.			
8.AP.3.2 Compose a program with multiple parts.	[26]		
8.AP.4.1 Use a beginner coding language (e.g., drag-and-drop, block-based) to design and code a complex program that solves a problem.	[27]		
8.AP.5.1 Compare and contrast variables that change or are constant.	[28]		
8.AP.5.2 Identify the variables needed to solve a given problem (i.e., information that needs to be tracked).	[29]		
Impacts of Computing			
8.IC.1.1 Justify how computer science is and can be used to solve problems in students' daily lives (e.g., mobile applications to accomplish tasks or solve problems in a neighborhood; remote traffic control).			
8.IC.1.2 Analyze positive and negative impacts of computing on society (e.g., personal, health, workforce, economy, education, culture, environment).			
8.IC.2.1 Compare and contrast current communication methods and computing devices.			
8.IC.3.1 Identify risks associated with sharing information digitally (e.g., phishing, identity theft, hacking).			
8.IC.4.1 Investigate recent laws that have been created to govern computer use (e.g., privacy, piracy, censorship, intellectual property).			
8.IC.5.1 Investigate historical and current trends of underrepresentation in computer science (e.g., race, ethnicity, gender, socioeconomic status).			
8.IC.5.2 Recognize computer scientists from underrepresented populations who have advanced computing.			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
8.IC.5.3 Explain how the lack of diverse perspectives and backgrounds restricts possible solutions to computational problems (e.g., first iteration of Google Maps included only driving directions, but later public transit and walking directions were added).			
8.IC.6.1 Identify traditional and nontraditional careers that use computer science (e.g., computer science in agriculture, medical, and public safety fields).			
8.IC.6.2 Relate the five disciplines of computing (i.e., computer science, software engineering, information technology, information systems, and computer engineering) to careers in various industries (e.g., advancements in healthcare, national security, and transportation).			
8.IC.7.1 Analyze the impact of computing and computer science over time (e.g., advantages such as faster, more efficient completion of tasks and access to the information; disadvantages such as fewer human jobs due to automation).			
8.IC.7.2 Understand the historical impact and future potential of exponential growth in computing (i.e., Moore's Law).			
8.IC.7.3 Identify and describe emerging technologies (e.g., virtual reality, biometrics, health monitoring systems).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
Computing Systems			
HS1.CS.1.1 Analyze the impact that computing devices have in real-world settings (e.g., traffic lights, medical devices, facial recognition).			
HS2.CS.1.1 Investigate how a problem is systematically solved through the selection and integration of hardware and software components.			
HS3.CS.1.1 Recommend modifications for existing computing devices and software to improve functionality for end users.			
HS4.CS.1.1 Develop a solution to a given problem using appropriate hardware and software (e.g., sensor devices, Wi-Fi capabilities, specialized displays, runtime modules, operating systems, application programming interfaces (APIs)).			
HS1.CS.1.2 Compare and contrast the elements of a computing system by examining hardware elements for their intended use (e.g., input-output (I/O) devices, random access memory (RAM), read only memory (ROM), storage devices, motherboards, and processors including the arithmetic logic unit (ALU), control unit, registers, cache memory, example implementations of some of these components using logic gates) (Virginia, 2017).			
HS2.CS.1.2 Analyze how various hardware and software layers provide simplifying abstractions (e.g., a redundant array of independent disks (RAID) controller hiding details of data storage on multiple disks, an operating system hiding details of virtual memory, the presentation layer of a network hiding details about encryption).			
HS3.CS.1.2 Justify hardware and software selections for specific applications by evaluating the components (e.g., databases, sensors, application programming interfaces (APIs)) of various computing devices (e.g., desktops, laptops, tablets, smartphones, specialized devices like global positioning systems (GPSs)).			
HS4.CS.1.2 Cite evidence of how selecting appropriate hardware and software components enhances user interfaces to provide better solutions for real-world problems.			
HS1.CS.1.3 Compare and contrast the strengths and weaknesses of various popular operating systems, (e.g., Windows, Windows Server, OS-X, Android, iOS, UNIX, Linux).			
HS2.CS.1.3 Analyze the roles of operating system software components (e.g., memory management, data storage and retrieval, process management, access control) in a specific solution (CSTA, 2017).			
HS3.CS.1.3 Investigate how the components of an operating system provide simplifying abstractions for a user or developer (e.g., the basic inputoutput system (BIOS), user interface, memory management, process management, file systems, devices, drivers).			
HS4.CS.1.3 Defend the choice of an appropriate operating system based on the requirements of a given computer system or project.			
HS1.CS.2.1 Interpret various types of error messages from various sources (e.g., operating systems, applications, application programming interfaces (APIs)) to assist in solving common computer malfunctions.			
HS2.CS.2.1 Research credible sources of information that can be used for complex troubleshooting strategies (e.g., modifying system settings, correcting connectivity problems).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
HS3.CS.2.1 Solve common computer malfunctions or describe the problem accurately, using technical vocabulary, so that others can solve it.			
HS4.CS.2.1 Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors (CSTA, 2017)	[30]		
Networks & the Internet			
HS1.NI.1.1 Describe how hardware, software, and protocols work together for transmitting data across networks.			
HS2.NI.1.1 Compare and contrast common network protocols (e.g., Internet Protocol (IP), File Transfer Protocol (FTP), Transmission Control Protocol (TCP), Domain Name System (DNS), Hypertext Transfer Protocol (HTTP), Hypertext Transfer Protocol with Secure Sockets Layer (HTTPS), Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP), Internet Message Access Protocol (IMAP), Telnet, Secure Shell Protocol (SSH)) (Arkansas, 2016).			
HS3.NI.1.1 Explain each layer of the Open Systems Interconnect (OSI) Model (Arkansas, 2016).			
HS4.NI.1.1 Trace data through the Open Systems Interconnect (OSI) Model.			
HS1.NI.1.2 Model the relationship between routers, switches, servers, topology, and addressing (CSTA, 2017).			
HS2.NI.1.2 Analyze the tradeoffs and hierarchical structures of various common network topologies (Arkansas, 2016).			
HS3.NI.1.2 Design a practical, efficient, and secure network solution (e.g., small office network) (Arkansas, 2016).			
HS4.NI.1.2 Evaluate the scalability and reliability of a network (e.g., bandwidth, load, delay, topology) (CSTA, 2017).			
HS1.NI.2.1 Reflect on case studies or current events in which governments or organizations experienced data leaks or data loss as a result of cybersecurity attacks (CSTA, 2017).			
HS2.NI.2.1 Evaluate how sensitive data can be affected by malware and other attacks (e.g., denial-of-service attacks, ransomware, viruses, worms, spyware, phishing) (CSTA, 2017).			
HS3.NI.2.1 Research security measures (i.e., hardware, software, and practices that control access to data and systems) to combat a variety of cybersecurity vulnerabilities (CSTA, 2017).			
HS4.NI.2.1 Recommend security measures (i.e., hardware, software, and practices that control access to data and systems) to address various scenarios based on factors such as efficiency, feasibility, and ethical impacts (CSTA, 2017).			
HS1.NI.2.2 Identify best practices of software development that improve computer security and protect devices and information from unauthorized access (e.g., encryption, authentication strategies, secure coding, safeguarding keys) (CSTA, 2017).			
HS2.NI.2.2 Compare and contrast ways software developers protect devices and information from unauthorized access (e.g., encryption, authentication strategies, secure coding, safeguarding keys) (CSTA, 2017).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
HS3.NI.2.2 Evaluate various security measures, considering tradeoffs between the usability and security of a computing system. (e.g., a web filter that prevents access to many educational sites but keeps a campus' network safe) (CSTA, 2017).			
HS4.NI.2.2 Select and justify cybersecurity recommendations (i.e., hardware, software, and practices that control access to data and systems) appropriate for an intended audience and purpose (CSTA, 2017).			
Data and Analysis			
HS1.DA.1.1 Describe the various data collection methods, data analysis tools, and data representation tools.			
HS2.DA.1.1 Compare and contrast the various data collection methods, data analysis tools, and data representation tools.			
HS3.DA.1.1 Explain how different collection methods and tools influence the amount and quality of the data that is observed and recorded.			
HS4.DA.1.1 Justify the choice of a data collection method, data analysis tool, and data representation tool over alternate options.			
HS1.DA.1.2 Describe the various data storage tools and data organization methods.			
HS2.DA.1.2 Compare and contrast the various data storage tools and data organization methods.			
HS3.DA.1.2 Justify choices on how data elements are organized and where data is stored considering cost, speed, reliability, accessibility, privacy, and integrity (e.g., local storage, portable storage, cloud storage).			
HS4.DA.1.2 Evaluate the data storage needs of a computing solution (e.g., file compression).			
HS1.DA.1.3 Distinguish between various methods of data representation (i.e., analog, digital, binary)	[31]		
HS2.DA.1.3 Translate between various methods of data representation (i.e., analog, digital, ASCII, binary).			
HS1.DA.2.1 Describe the properties of a data set that could be used to explore a realworld phenomenon or support a claim.			
HS2.DA.2.1 Compare and contrast data sets that could be used to explore a real-world phenomenon or support a claim.			
HS3.DA.2.1 Create data sets that could be used to explore a realworld phenomenon or support a claim.			
HS4.DA.2.1 Evaluate the use of large data sets to explore a realworld phenomenon or support a claim.			
HS1.DA.2.2 Identify various types of computational models and their uses for data composed of multiple data elements that relate to one another (e.g., population data may contain information about age, gender, height) (K – 12 Framework, 2016).			
HS2.DA.2.2 Compare and contrast various types of computational models and their uses for data composed of multiple data elements that relate to one another (e.g., population data may contain information about age, gender, height) (K – 12 Framework, 2016).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
HS3.DA.2.2 Evaluate the limitations of a computational model and the accuracy of inferences.			
HS4.DA.2.2 Create a computational model using large data sets, make inferences, and address the limitations of the model.			
HS1.DA.3.1 Identify a data set that could be used to solve a real-world problem.			
HS2.DA.3.1 Evaluate how the same data set can be visualized and reconstructed to support multiple sides of an issue.			
HS3.DA.3.1 Construct a data visualization to solve a real-world problem using software tools or programming (e.g., generated scatter, bar, and line charts).			
HS4.DA.3.1 Analyze patterns in a data visualization then select a collection tool to validate a claim or share information with a group of people.			
HS1.DA.3.2 Organize collected data to communicate the solution to a real-world phenomenon and support a claim.			
HS2.DA.3.2 Compare and contrast data visualizations for exploring real-world phenomenon or supporting a claim.			
HS3.DA.3.2 Evaluate possible computational models for data visualizations that aid in solving a variety of problems.			
HS4.DA.3.2 Create a computational model for data visualization.			
Algorithms and Programming			
HS1.AP.1.1 Create flowcharts and/or pseudocode to express a problem or idea as an algorithm.	[32]		
HS2.AP.1.1 Create algorithms to solve computational problems that have an application in the real world (e.g., local community, church, civic organization, school, home life).	[33]		
HS3.AP.1.1 Adapt predefined algorithms to solve computational problems.			
HS4.AP.1.1 Evaluate algorithms in terms of efficiency, correctness, and clarity (CSTA, 2017).			
HS1.AP.2.1 Trace the flow of execution of a program that uses a combination of control structures (e.g., conditionals, loops, event handlers, recursion).	[34]		
HS2.AP.2.1 Design and iteratively develop programs that combine control structures (e.g., conditionals, loops, event handlers, recursion).	[35]		
HS3.AP.2.1 Justify the selection of specific control structures explaining the benefits and drawbacks of choices made (e.g., tradeoffs involving implementation, readability, and program performance).			
HS1.AP.2.2 Trace the flow of execution of a program that uses a variety of programming constructs (e.g., procedures, modules, objects).	[36]		

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
HS2.AP.2.2 Design a solution through systematic analysis using programming constructs (e.g., procedures, modules, objects).	[37]		
HS3.AP.2.2 Justify the selection of specific programming constructs, explaining the benefits and drawbacks of choices made on the program's execution.			
HS1.AP.3.1 Decompose tasks into smaller, reusable parts to facilitate the design, implementation, and review of programs.			
HS2.AP.3.1 Develop code to solve the smaller parts of a decomposed task that can be reused to solve similar problems (e.g., procedures, functions, objects).	[38]		
HS3.AP.3.1 Build a complex solution to a problem that incorporates reusable code (e.g., student created, application programming interfaces (APIs), libraries).	[39]		
HS4.AP.3.1 Justify the selection of modular parts in the creation of a complex solution.			
HS1.AP.4.1 Plan and develop programs for a variety of audiences using a process that incorporates development, feedback, and revision.			
HS2.AP.4.1 Plan and develop a program that addresses potential security issues.			
HS3.AP.4.1 Plan and develop a program that is accessible across multiple computing platforms (e.g., iOS, Unix, Windows, web-based).			
HS4.AP.4.1 Evaluate a program through a review process (e.g., code review, beta testing, pilot group).	[40]		
HS1.AP.4.2 Seek and incorporate feedback to refine a solution (e.g., users, team members, code review, teachers)			
HS2.AP.4.2 Systematically test programs using a range of test cases to meet design specifications (e.g., specific outcomes, functionality, user interface, error handling) (CSTA, 2017).			
HS3.AP.4.2 Evaluate and refine programs to make them more usable, functional, and accessible.			
HS4.AP.4.2 Implement version control to track program refinements.			
HS1.AP.4.3 Recognize the variety of documentation methods available while developing a program (e.g., inline comments, procedure header, purposeful naming).	[41]		
HS2.AP.4.3 Document programs in order to make them easier to follow, test, and debug.	[42]		
HS3.AP.4.3 Document programs that use non-user-created resources (e.g., code, media, libraries) giving attribution to the original creator.			
HS4.AP.4.3 Justify design decisions by documenting the design process of complex programs (e.g., developer journal, digital portfolio, presentation).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
HS1.AP.4.4 Examine licenses (i.e., permissions) that limit or restrict use of resources (e.g., freeware, shareware, open source, creative commons).			
HS2.AP.4.4 Discuss the implications of using licensed resources in a developed solution.			
HS3.AP.4.4 Develop a systematic solution that incorporates licensed resources.			
HS4.AP.4.4 Research the process for licensing student-created resources.			
HS1.AP.5.1 Justify and use appropriate data types (i.e., primitive and non-primitive) in simple programs.	[43]		
HS2.AP.5.1 Determine when data structures (e.g., lists, arrays, tuples, stacks, queues, structures) are more appropriate than simple data types and incorporate them in programs.		[44]	
HS3.AP.5.1 Determine when external data structures (e.g., databases, flat files) are appropriate and incorporate them in programs.			
HS4.AP.5.1 Justify how data structures and abstraction are used to manage program complexity			
Impacts of Computing			
HS1.IC.1.1 Research computing solutions to problems in different countries, considering the personal, ethical, social, economic, and cultural impact (e.g., the use of drones to deliver blood and medical supplies in countries in Africa, the use of Uber in India to address traffic congestion).			
HS2.IC.1.1 Compare and contrast the efficiency, feasibility, and ethical impacts of deploying the same computing solution in various countries.			
HS3.IC.1.1 Hypothesize the future impacts of computing across the world (e.g., personal, health, workforce, economy, education, culture, environment).			
HS1.IC.1.2 Research traditional and non-traditional computer science careers.			
HS2.IC.1.2 Identify a computer science career in a nontraditional computer science industry for each of the five computing disciplines (i.e., computer science, software engineering, information technology, information systems, computer engineering).			
HS3.IC.1.2 Research how computing is used in non-traditional computer science careers (e.g., sensors on soldiers' or firefighters' uniforms, robots to detect and diffuse explosive devices).			
HS1.IC.2.1 Select the most appropriate means of communication for given situations (e.g., personal versus professional communication, communication with teachers and employers).			
HS2.IC.2.1 Discuss how social media and computing devices have positively and negatively impacted communication.			
HS3.IC.2.1 Justify proper and improper use of social media and computing devices (e.g., role-playing and example scenarios).			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
HS4.IC.2.1 Create rules of etiquette for proper use of social media and computing devices.			
HS1.IC.2.2 Discuss issues related to personal security (e.g., personal, financial, professional).			
HS2.IC.2.2 Define and visually display students' digital footprint			
HS3.IC.2.2 Analyze the relationship between students' digital footprint and personal security.			
HS4.IC.2.2 Recommend methods to protect digital information in different situations (e.g., traveling to other countries, two-factor authentication, encryption).			
HS1.IC.2.3 Explain the implications of proper and improper use of social media (e.g., college admissions, employment, cyberbullying laws).			
HS2.IC.2.3 Identify ethical and legal computing practices.			
HS3.IC.2.3 Distinguish among ethical, unethical, legal, and illegal computing practices (e.g., fair-use, illegal music/video downloads, sharing copyrighted pictures/videos, black-hat hacking, white-hat hacking).			
HS4.IC.2.3 Investigate how computer use and digital privacy are governed across the globe (e.g., government regulations for computer use in the United State, Canada, China, North Korea, and Russia).			
HS1.IC.3.1 Identify factors (e.g., net neutrality, government regulations, infrastructure, funding) that impact equitable access to computing resources for underrepresented groups (e.g., race, ethnicity, gender, geographic location, socioeconomic status).			
HS2.IC.3.1 Research current efforts to provide equitable access to computing resources for underrepresented groups (e.g., race, ethnicity, gender, geographic location, socioeconomic status)			
HS3.IC.3.1 Evaluate the effectiveness of current efforts to provide equitable access to computing resources for underrepresented groups (e.g., race, ethnicity, gender, geographic location, socioeconomic status).			
HS4.IC.3.1 Design a solution to improve equitable access to computing resources for underrepresented groups (e.g. classroom, school, neighborhood).			
HS1.IC.3.2 Identify computer scientists from underrepresented groups and their specific contributions (e.g., African-American, Latino, women, disabled).			
HS1.IC 3.3 Identify the advantages and disadvantages of diverse perspectives and backgrounds when solving computational problems.			
HS2.IC.3.3 Evaluate existing computing solutions according to inclusivity or non-inclusivity (e.g., sight and hearing impairment, ethnicity, age).			
HS3.IC.3.3 Recommend modifications to make a current computing solution more inclusive for all users.			
HS1.IC.4.1 Discuss how Moore's Law has impacted computing.			

South Carolina CS Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
HS2.IC.4.1 Discuss the advantages and disadvantages of advancing and emerging technologies over time (e.g., the impacts of artificial intelligence, virtual reality, and biometrics on productivity, job loss, inventions, quality of life, and globalization).			
HS3.IC.4.1 Hypothesize problems that the next generation of computing will solve.			
HS1.IC.4.2 Define and provide examples of big data (e.g., information collected from social media or smartphone use)			
HS2.IC.4.2 Research how big data is used to solve computing problems.			

[1] This begins in Mission 4 when function is called multiple times

[2] This is accomplished by creating a flowchart which are introduced in the teachers' manual

[3] This starts with Mission 6

[4] All of our missions are online coding activities

[5] This is introduced in Mission 2 and the teachers' manual

[6] This is introduced in the teachers' manual

[7] All missions do this

[8] This is the pseudocode that is introduced in the teachers' manual

[9] All missions do this

[10] 3.8 introduces the use of variables

[11] Mission 4 introduces data types

[12] This is discussed at the end of each mission

[13] All of our missions do this

[14] This is pseudocode which is introduced in the teachers' manual

[15] This is introduced in the teachers' manual

[16] All missions do this

[17] Our missions are designed this way to take care of small parts and in the end you have an entire program

[18] This is the flowcharts that are introduced in the teachers' manual

[19] All missions do this

[20] 3.8 introduces the use of variables

[21] Mission 4 introduces the different data types

[22] This is discussed at the end of each mission

[23] Mission 2 discusses this

[24] This is introduced in the teachers' manual

[25] This begins in Mission 4.7

[26] Mission 6 begins coding multiple parts

[27] All missions do this

[28] Mission 6 begins changing variables instead of leaving them constant like previous missions

[29] 3.8 begins the use of variables

[30] Mission 2 and the teachers' manual introduce troubleshooting techniques

[31] This is discussed any time a new sensor is introduced

[32] Flowcharts and pseudocodes are introduced in the teachers' manual

[33] These are the remixes depending on teacher rubrics

[34] This would be flowcharts starting at Mission 4 that has conditionals and Mission 6 that has loops

[35] This would be remixes starting at Mission 4 that has conditionals and Mission 6 that has loops

[36] This would be flowcharts starting at Mission 4 that has conditionals and Mission 6 that has loops

[37] This would be the process of creating a remix

[38] Function use begins in Mission 4 and Mission 9 begins the self created functions

[39] All missions use and discuss the use of libraries

[40] Code Tracing Charts are introduced in the teachers' manual

[41] 5.5 discusses descriptive naming of variables and the use of comments

[42] 5.5 introduces the use of comments

[43] Data types begin in Mission 4

[44] Mission 7 begins the use of lists