

Standards numbers: grade.concept.subconcept.#		PK	Kindergarten	1st Grade	2nd Grade
Concept	Subconcept				
Computing Systems	Devices	PK.CS.D.01 With guidance, demonstrate how to operate a computing device.	PK.CS.D.01 With guidance, demonstrate how to operate a computing device.	PK.CS.D.01 With guidance, demonstrate how to operate a computing device.	PK.CS.D.01 With guidance, demonstrate how to operate a computing device.
	Hardware & Software	PK.CS.HS.01 With guidance, use appropriate terminology to locate and identify common computing devices and components in a variety of environments (e.g. turn on, navigate, open/close programs/apps).	K.CS.HS.01 Use appropriate terminology to locate and identify common computing devices and components in a variety of environments (e.g. turn on, navigate, open/close programs/apps).	1.CS.HS.01 Use appropriate terminology in identifying and describing the function of common computing devices and components. (e.g. use an app to draw on the screen, use software to write a story or control robots).	2.CS.HS.01 Model the use of components of a computing system, it's basic functions, peripherals (connected hardware), and storage features.
		PK.CS.HS.02 With guidance, correctly use software that controls computing devices (e.g. desktop computer, laptop computer, tablet device, monitor, mouse).	K.CS.HS.02 Identify and use software that controls computing devices (e.g. desktop computer, laptop computer, tablet device, monitor, keyboard, mouse, printer).	1.CS.HS.02 With guidance select and use appropriate software/apps for an intended outcome. (e.g. desktop computer, laptop computer, tablet device, monitor, keyboard, mouse, printer).	2.CS.HS.02 Self select and use appropriate software/apps for an intended outcome. (e.g. using the hard drive, memory/storage, printers, scanners, wireless and cabled connections, and cloud storage).
	Troubleshooting	PK.CS.T.01 Recognize that computing systems might not work as expected and with guidance can identify simple hardware or software problems (e.g. volume turned down on headphones, monitor turned off).	K.CS.T.01 Recognize that computing systems might not work as expected and use accurate terminology to identify simple hardware or software problems (e.g. volume turned down on headphones, monitor turned off).	1.CS.T.01 Identify and describe basic hardware and software problems using accurate terminology (app or program is not working as expected, no sound is coming from the device, caps lock turned on, WiFi not working).	2.CS.T.01 Using accurate terminology, identify and resolve simple hardware and software problems and strategies for solving these problems.
Networks & the Internet	Network Communication & Organization	PK.NI.NCO.01 Recognize that computing devices are connected via wired or wireless networks so that they can communicate with each other.	K.NI.NCO.01 Use computing devices to connect with people or other devices using a network to communicate, access, and share information as a class (e.g. the internet, video conferencing, email, file transfer).	1.NI.NCO.01 Recognize that by connecting computing devices together they can share information using a network (e.g. wired or wireless network).	2.NI.NCO.01 Use computing devices to share information and communicate with others using a network.
	Cybersecurity	PK.NI.C.01 Recognize that passwords are private and should be kept secret.	K.NI.C.01 Use a form of secure access to protect private information and discuss the effects of password misuse (e.g. logging into a device, educational websites, authentication, thumbprint recognition).	1.NI.C.01 Identify what authentication methods (passwords) are; explain why they are not shared; and discuss what makes a password strong. Independently, use passwords to access technological devices, apps, etc.	2.NI.C.01 Demonstrate use of strong authentication methods to access and protect devices and data. Understand the effects of retaining password privacy.
Data Analysis	Storage	PK.DA.S.01 Know that the computing device can save information as data that can be searched, modified, and saved or deleted (e.g. save photos, files, or videos).	K.DA.S.01 With guidance, demonstrate that computing devices can save information as data that can be searched, modified, and saved or deleted (e.g. save photos, files, or videos).	1.DA.S.01 With guidance locate, open, modify, delete and save an existing file, use appropriate file-naming conventions, and recognize that the file exists within an organizational structure (drive, folder, file).	2.DA.S.01 Manipulate existing files while use appropriate file-naming conventions. With guidance, develop and modify an organizational structure by creating, copying, moving, and deleting files and folders.
	Collection	PK.DA.C.01 Students understand that data about themselves and the world around them is collected, used, and organized in a meaningful way.	K.DA.C.01 Students learn how data about themselves and the world around them is collected, used, and organized in a meaningful way.	1.DA.C.01 With guidance, collect data and present it two different ways (chart or graph).	2.DA.C.01 With guidance, collect and present the same data in various visual formats.
	Visualization & Transformation	PK.DA.CVT.01 Students represent collected data in a visual way. (e.g. charts, graphs, tables).	K.DA.CVT.01 Students represent collected data in a visual way through a computing device (e.g. charts, graphs, tables).	1.DA.CVT.01 With guidance, identify and interpret data from a chart or graph (visualization) in order to make a prediction, with or without a computing device.	2.DA.CVT.01 Collect data over time and organize it on a chart or graph in order to make a prediction
	Inference & Models	PK.DA.IM.01 Students look for patterns in data, make predictions, and make a model (e.g. make predictions on weather data, butterfly life cycle, etc.) and present in a picture graph or pattern.	K.DA.IM.01 Students look for patterns in data, make predictions, make a model, and draw conclusions (e.g. make predictions on weather data, butterfly life cycle, etc.) and present in a picture graph or pattern.	1.DA.IM.01 Create a model of an object or process in order to identify patterns and essential elements. (e.g. water table, butterfly life cycle, seasonal weather patterns).	2.DA.IM.01 Use patterns in data to make inferences or predictions based on data collected from users or simulations.
Programming	Algorithms	PK.AP.A.01 With guidance, construct and execute algorithms (set of step-by-step instructions) that includes sequencing and simple loops to accomplish a task, with or without a computing device (e.g. verbally, kinesthetically, with robot devices or a programming language, block coding).	K.AP.A.01 Construct and execute algorithms (set of step-by-step instructions) that includes sequencing and simple loops to accomplish a task, both independently, collaboratively, with or without a computing device (e.g. verbally, kinesthetically, with robot devices or a programming language, block coding).	1.AP.A.01 With guidance, model daily processes and follow algorithms (sets of step-by-step instructions) for complete tasks verbally, kinesthetically, with robot devices, or a programming language.	2.AP.A.01 Both independently and collaboratively construct and follow algorithms that include sequencing and simple loops to accomplish a task verbally, kinesthetically, with robot devices, or a programming language.
	Variables	PK.AP.V.01 With guidance, understand that numbers represent different types of data using numbers or other symbols (e.g. thumbs up/thumbs down for yes/no color by number, arrows for direction, encoding/decoding a word using numbers or pictographs).	K.AP.V.01 With guidance, recognize that numbers represent different types of data using numbers or other symbols (e.g. thumbs up/thumbs down for yes/no color by number, arrows for direction, encoding/decoding a word using numbers or pictographs).	1.AP.V.01 With guidance, model the way that programs store and manipulate data by using numbers or other symbols to represent information (e.g. thumbs up/thumbs down for yes/no, use arrows when writing algorithms to represent direction, or encode and decode words using numbers, pictographs, or other symbols to represent letters or words).	2.AP.V.01 Use and model the way a computer program stores, accesses, and manipulates data that is represented as a variable.

Algorithms and	Control	PK.AP.C.01 With guidance, create programs to accomplish tasks using a programming language, robot device, or unplugged activity that includes sequencing events and simple loops (e.g. emphasizing beginning, middle, and end; collaborative programming).	K.AP.C.01 With guidance, independently or collaboratively create programs to accomplish tasks using a programming language, robot device, or unplugged activity that includes sequencing events and simple loops (e.g. emphasizing beginning, middle, and end; collaborative programming).	1.AP.C.01 With guidance, independently, or collaboratively construct algorithms (sets of step-by-step instructions) to accomplish tasks using a programming language, robot device, or unplugged activity that includes sequencing and repetition, to express ideas or address a problem.	2.AP.C.01 Independently and collaboratively create programs to accomplish tasks using a programming language such as block based programming using a robot device, or unplugged activity that includes simple loops, sequencing, and repetition.	
	Modularity	PK.AP.M.01 With guidance, decompose (break down) a larger problem into smaller subproblems.	K.AP.M.01 With guidance, decompose (break down) a larger problem into smaller subproblems or combine simple tasks to make something more complex.	1.AP.M.01 Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.	2.AP.M.01. Independently decompose (break down) a larger problem into smaller subproblems and steps needed to solve those problems.	
	Program Development		PK.AP.PD.01 Create a design document to illustrate thoughts, ideas, and stories in a sequential manner.	K.AP.PD.01 Create a design document to illustrate thoughts, ideas and stories in a sequential manner (e.g. storyboard, mindmap, sequential graphic organizer).	1.AP.PD.01 Independently or with guidance, create a grade-level appropriate artifact to illustrate thoughts, ideas, or stories in a sequential (step-by-step) manner (e.g. story map, storyboard, and sequential graphic organizer).	2.AP.PD.01 independently create a grade-level appropriate artifact to illustrate thoughts, ideas, or stories in a sequential (step-by- step) manner (e.g., story map, storyboard, and sequential graphic organizer).
			PK.AP.PD.02 Recognize that digital items can be owned and that proper credit needs to be given (e.g. using code, music, pictures).	K.AP.PD.02 With guidance, give credit to ideas, creations, and solutions of others while developing algorithms (e.g. using code, music, pictures).	1.AP.PD.02 Independently or with guidance give credit to ideas, creations and solutions of others while writing and/or developing programs.	2.AP.PD.02. Give credit to ideas, creation (such as code, music, or pictures) and solutions of others while writing and developing programs.
			PK.AP.PD.03 With guidance, construct, execute, and debug (identify and fix) algorithms using a programming language and or an unplugged activity that includes sequencing (e.g. use block based programming).	K.AP.PD.03 With guidance, independently or collaboratively construct, execute, and debug (identify and fix) algorithms using a programming language and or an unplugged activity that includes sequencing (e.g. use block based programming).	1.AP.PD.03 With guidance, independently, or collaboratively debug (identify and fix) programs using a programming language and/or unplugged activity that includes sequencing and repetition.	2.AP.PD.03. Independently and collaboratively analyze and debug (fix) an algorithm using a programming language and/or unplugged activity that includes sequencing and simple loops.
	PK.AP.PD.04 With guidance, use correct terminology in the development of an algorithm to solve a simple problem (e.g. beginning, middle, end).	K.AP.PD.04 Use correct terminology in the development of an algorithm to solve a simple problem (e.g. beginning, middle, end).	1.AP.PD.04 Use correct terminology (first, second, third) and explain the choices made in the development or an algorithm to solve a simple problem.	2.AP.PD.04. Use correct terminology (debug, program input/output, code) to explain the development of an algorithm to solve a problem in an unplugged activity, hands on manipulatives, or a programming language.		
Impacts of Computing	Culture	PK.IC.C.01 Understand different ways in which types of technologies are used in your daily life.	K.IC.C.01 Understand different ways in which types of technologies are used in your daily life.	1.IC.C.01 Identify how people use different types of technologies in their daily work and personal lives.	2.IC.C.01 Recognize and describe how different technologies used daily in work and at home are used to solve problems or make work and life easier.	
	Social Interactions	PK.IC.SI.01 With guidance understand what would be appropriate while participating in an online environment.	K.IC.SI.01 With guidance identify appropriate manners while participating in an online environment.	1.IC.SI.01 With guidance, identify appropriate and inappropriate behavior. Act responsibly while participating in an online community and know how to report concerns.	2.IC.SI.01 Aid in developing an appropriate code of conduct, explain and practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	
	History	PK.IC.H.01 Understand that computing technology has changed and improved the way people live, work, and interact.	K.IC.H.01 Discuss examples of how computing technology has changed and improved the way people live, work, and interact.	1.IC.H.01 Compare how people live and work before and after the implementation or adoption of new computing technology.	2.IC.H.0. Recognize how technologies have changed the world, and explore how the needs of society have impacted the changes in technology.	
	Safety, Law, & Ethics	PK.IC.SLE.01 With guidance understand responsible digital citizenship (legal and ethical behaviors) in the use of technology systems and software.	K.IC.SLE.01 Practice responsible digital citizenship (legal and ethical behaviors) in the use of technology systems and software.	1.IC.SLE.01 Practice responsible digital citizenship (legal and ethical behaviors) in the use of technology systems and software. Keep login information private, and log off of devices appropriately.	2.IC.SLE.01 Practice responsible digital citizenship in all technology use. Understand digital data has intellectual property rights (belongs to others) and it can not be claimed as your own.	
	Community Partnerships	PK.IC.CP.01 Discuss the fact that a wide range of jobs require knowledge or use of computer science.	K.IC.CP.01 Understand that a wide range of jobs require knowledge or use of computer science.	1.IC.CP.01 Compare and contrast examples of how computing technology has changed and improved the way people live, work, and interact.	2.IC.CP.01. Investigate how computer science has impacted your daily life and the jobs in your community and the world around you.	
Concept	Subconcept	3rd Grade	4th Grade	5th Grade		
Systems	Devices	3.CS.D.01 Identify how computing devices can be connected to other devices to extend their capabilities.	4.CS.D.01 Identify and explain how computing devices can be connected to other devices to extend their capabilities.	5.CS.D.01 Model and communicate how computing devices can be connected to other devices to extend their capabilities.		
	Hardware and Software	3.CS.HS.01 Model how information flows through hardware and software to accomplish tasks.	4.CS.HS.01 Model how information is translated, transmitted, and processed in order to flow through hardware and software.	5.CS.HS.01 Model that information is translated into binary between software and hardware to accomplish tasks.		

Computing	Troubleshooting	3.CS.T.01 Identify, using accurate terminology, simple hardware and software problems and strategies for solving these problems (e.g., reboot device, check for power, close and reopen applications, check network availability, and discuss problems with peers and adults).	4.CS.T.01 Identify, using accurate terminology, simple hardware and software problems that may occur during everyday use, discuss problems with peers and adults, and apply strategies for solving these problems (e.g., rebooting the device, checking the power, force shut down of an application).	5.CS.T.01 Using accurate terminology, identify simple hardware and software problems that may occur during everyday use.	
	Networks & the Internet	3.NI.NCO.01 Model how a device on a network sends and receives information.	4.NI.NCO.01 Explain how information is sent and received across physical or wireless paths. (It is broken down into smaller pieces called packets and transmitted from one location to another.)	5.NI.NCO.01 Model how information is broken down into smaller pieces and transmitted through multiple devices over networks and the internet, and how these pieces are reassembled at the destination.	
Networks & the Internet	Cybersecurity	3.NI.C.01 Identify problems that relate to inappropriate use of computing devices and networks.	4.NI.C.01 Identify and explain issues related to responsible use of technology and information, and describe personal consequences of inappropriate use.	5.NI.C.01 Discuss real-world cybersecurity problems and identify strategies for how personal information can be protected.	
	Data Analysis	Storage	3.DA.S.01 Compare and contrast the formats and storage requirements for different types of information (e.g., music, video, images, and text).	4.DA.S.01 Classify different storage locations (physical, shared, or cloud) based on the type of file, storage requirements (file size, availability, available memory), and sharing requirements.	5.DA.S.01 Evaluate trade-offs of file types, storage requirements, and sharing requirements, including comparisons of availability and quality.
Data Analysis	Collection	3.DA.C.01 Gather data to solve a problem or answer a question.	4.DA.C.01 Gather and manipulate data using the appropriate digital tool.	5.DA.C.01 Select the appropriate tool to collect data that solves a problem	
	Visualization & Transformation	3.DA.VT.01 Create a simple data visualization (e.g., graphs, charts and infographics) based on data collected by or provided to student.	4.DA.VT.01 Organize and present collected data visually to highlight comparisons.	5.DA.VT.01 Organize and present collected data to highlight comparisons and support a claim.	
	Inference and Models	3.DA.IM.01 Utilize data to make predictions and discuss whether there is adequate data to make reliable predictions.	4.DA.IM.01 Determine how the accuracy of conclusions are influenced by the amount and relevance of the data collected.	5.DA.IM.01 Use data to discover or propose cause and effect relationships, predict outcomes, or communicate an idea.	
Algorithms and Programming	Algorithms	3.AP.A.01 Compare multiple algorithms for the same task.	4.AP.A.01 Analyze and refine multiple algorithms for the same task.	5.AP.A.01 Analyze and refine multiple algorithms for the same task and determine which algorithm is the most efficient.	
	Variables	3.A.V.01 Utilize and create simple programs that use variables to store and modify grade level appropriate data.	4.AP.V.01 Utilize, create, and modify programs that use variables, with grade level appropriate data.	5.A.V.01 Utilize and create programs that create, use, modify, and combine variables with grade level appropriate data	
	Control	3.AP.C.01 Create simple programs using a programming language that utilize sequencing, repetition, conditionals, and variables to solve a problem or express ideas both independently and collaboratively.	4.AP.C.01 Create programs using a programming language that utilize sequencing, repetition, conditionals and variables using math operations manipulate values to solve a problem or express ideas both independently and collaboratively.	5.AP.C.01 Create programs using a programming language that utilize sequencing, repetition, conditionals, event handlers, and variables using math operations to manipulate values to solve a problem or express ideas both independently and collaboratively.	
	Modularity	3.AP.M.01 Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.	4.AP.M.01 Decompose large problems into smaller, manageable subproblems to facilitate the program development process.	5.AP.M.01 Decompose large problems into smaller, more manageable subproblems. Then form algorithms to solve each subproblem	
		3.AP.M.02 With grade appropriate complexity, modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.	4.AP.M.02 With grade appropriate complexity, modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.	5.AP.M.02 With grade appropriate complexity, modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.	
	Program Development	3.AP.PD.01 Create a plan using an iterative process to plan the development of a program while solving simple problems (e.g., storyboard, flowchart, pseudo-code, story map).	4.AP.PD.01 Create a plan using an iterative process to plan the development of a program that includes user preferences while solving simple problems.	5.AP.PD.01 Create a plan using an iterative process for the development of a program that includes others' perspectives and user preferences while solving simple problems.	
		3.AP.PD.02 Use proper citations and document when ideas are borrowed and changed for their own use (e.g., using pictures created by others, using music created by others, remixing programming projects).	4.AP.PD.02 Use proper citations and document when ideas are borrowed and changed for their own use (e.g., using pictures created by others, using music created by others, remixing programming projects).	5.AP.PD.02 Use proper citations and document when ideas are borrowed and changed for their own use (e.g., using pictures created by others, using music created by others, remixing programming projects).	
		3.AP.PD.03 Analyze and debug a program that includes sequencing, repetition and variables in a programming language.	4.AP.PD.03 Analyze, debug, and create a program that includes sequencing, repetition and variables in a programming language.	5.AP.PD.03 Analyze, debug, and create a program that includes sequencing, repetition and variables in a programming language.	
		3.AP.PD.04 Communicate and explain your program development using comments, presentations and demonstrations.	4.AP.PD.04 Communicate and explain your program development using comments, presentations and demonstrations.	5.AP.PD.04 Communicate and explain your program development using comments, presentations and demonstrations.	

Impacts of Computing	Culture	3.IC.C.01 Identify possible problems and how computing devices have built in features for increasing accessibility to all users.	4.IC.C.01 Brainstorm problems and ways to improve computing devices to increase accessibility to all users.	5.IC.C.01 Develop, test, and refine digital artifacts to improve accessibility and usability for a computing device or program.
	Social Interactions	3.IC.SI.01 Develop a code of conduct, explain, and practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior. (Digital Citizenship)	4.IC.SI.01 Develop a code of conduct, explain, and practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior. (Digital Citizenship)	5.IC.SI.01 Develop a code of conduct, explain, and practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior. (Digital Citizenship)
		3.IC.SI.02 Identify how computational products may be, or have been, improved to incorporate diverse perspectives.	4.IC.SI.02 As a team, consider each others perspectives on improving a computational product.	5.IC.SI.02 As a team, collaborate with people and resources outside of your normal space to include diverse perspectives to improve computational products.
	History	3.IC.H.01 Identify computing technologies that have changed the world, and express how those technologies influence, and are influenced by, society.	4.IC.H.01 Identify and give examples of computing technologies that have changed the world, and express how those technologies influence, and are influenced by, society.	4.IC.H.01 Identify and explain the evolution of computing technologies that have changed the world.
	Safety, Law, & Ethics	3.IC.SLE.01 Identify types of digital data that may have intellectual property rights that prevent copying or require attribution.	4.IC.SLE.01 Discuss the social impact of violating intellectual property rights.	5.IC.SLE.01 Observe intellectual property rights and give appropriate credit when using resources.
	Community Partnerships	3.IC.CP.01 Explore the connections between computer science and other fields.	4.IC.CP.01 Explore the connections between computer science and other fields.	5.IC.CP.01 Explore the connections between computer science and other fields.
Concept	Subconcept	6th Grade	7th Grade	8th Grade
Computing Systems	Devices	6.CS.D.01 Evaluate existing computing devices and recommend improvements to design based on analysis of personal interaction with the device.	7.CS.D.01 Evaluate existing computing devices and recommend improvements to design based on analysis of how other users interact with the device.	8.CS.D.01 Develop and implement a process to evaluate existing computing devices and recommend improvements to design based on analysis of how other users interact with the device.
	Hardware & Software	6.CS.HS.01 Compare the multiple considerations and potential tradeoffs of software and hardware, such as functionality, cost, size, speed, accessibility, and aesthetics	7.CS.HS.01 Evaluate and recommend improvements to a computing system involving multiple considerations and potential tradeoffs of software and hardware, such as functionality, cost, size, speed, accessibility, and aesthetics	8.CS.HS.01 Model a computing system involving multiple considerations and potential tradeoffs of software and hardware, such as functionality, cost, size, speed, accessibility, and aesthetics
	Troubleshooting	6.CS.T.01 Identify increasingly complex software and hardware problems with computing devices and their components.	7.CS.T.01 Identify and fix increasingly complex software and hardware problems with computing devices and their components.	8.CS.T.01 Systematically identify, fix, and document increasingly complex software and hardware problems with computing devices and their components.
Networks & the Internet	Network Communication & Organization	6.NI.NCO.01 Model a simple protocol for transferring information using packets.	7.NI.NCO.01 Explain how a system responds when a packet is lost and the effect it has on the transferred information.	8.NI.NCO.01 Explain protocols and their importance to data transmission; model how packets are broken down into smaller pieces and how they are delivered.
	Cybersecurity	6.NI.C.01 Identify existing cybersecurity concerns with the Internet and systems it uses.	7.NI.C.01 Explain how to protect electronic information, both physical (e.g. hard drive) and digital, identify cybersecurity concerns and options to address issues with the Internet and the systems it uses.	8.NI.C.01 Evaluate physical and digital procedures that could be implemented to protect electronic data/information; explain the impacts of hacking, ransomware, scams, fake scans, and ethical/legal concerns.
		6.NI.C.02 Explain the importance of secured websites and describe how one method of encryption works.	7.NI.C.02 Identify and explain two or more methods of encryption used to ensure and secure the transmission of information.	8.NI.C.02 Compare the advantages and disadvantages of multiple methods of encryption to model the secure transmission of information.
Data Analysis	Storage	6.DA.S.01 Identify how the same data can be represented in multiple ways.	7.DA.S.01 Create multiple representations of data.	8.DA.S.01 Analyze multiple methods of representing data and choose the most appropriate method for representing data.
	Collection	6.DA.C.01 Distinguish between different types of data and computational tools and how this effects the accuracy of the data. (for example, surveys versus sensor data)	7.DA.C.01 Compare between different types of data choose the appropriate computational tool to use	8.DA.C.01 Develop, implement, and refine a process that utilizes computational tools to collect meaningful data.
	Visualization & Transformation	6.DA.VT.01 Represent data using different encoding systems (e.g. binary, Unicode, Morse code, shorthand, student-created code)	7.DA.VT.01 Evaluate ways Data can be transformed to remove errors, highlight or expose relationship, and/or make it easier for computers to process	8.DA.VT.01 Develop, implement, and refine a process to make data more useful and reliable.
	Inference & Models	6.DA.IM.01 Use models and simulations to formulate, refine, and test hypotheses.	7.DA.IM.01 Discuss the correctness of a model representing a system by comparing the model's generated results with observed data from the modeled system.	8.DA.IM.01 Refine computational models based on the data generated by the models.

Algorithms and Programming	Algorithms	6.AP.A.01 Use an existing algorithm in natural language or pseudocode to solve complex problems.	7.AP.A.01 Select and modify an existing algorithm in natural language or pseudocode to solve complex problems.	8.AP.A.01 Design algorithms in natural language, flow and control diagrams, comments within code, and/or pseudocode to solve complex problems.	
	Variables	6.AP.V.01 Create programs that use variables to store and modify grade level appropriate data.	7.AP.V.01 Create programs using variables with purposeful and thoughtful naming conventions for identifiers to improve program readability	8.AP.V.01 Create programs using variables with purposeful and thoughtful naming conventions for identifiers to improve program readability	
	Control	6.AP.C.01 Develop programs that utilize combinations of repetition, conditionals, and the manipulation of variables representing different data types.	7.AP.C.01 Develop programs that utilize combinations of repetition, compound conditionals, and the manipulation of variables representing different data types.	8.AP.C.01 Develop programs that utilize combinations of nested repetition, compound conditionals, procedures without parameters, and the manipulation of variables representing different data types.	
	Modularity	6.AP.M.01 Decompose problems into parts to facilitate the design, implementation, and review of programs.	7.AP.M.01 Decompose problems into parts to facilitate the design, implementation, and review of increasingly complex programs.	8.AP.M.01 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of complex programs.	
	Program Development	6.AP.PD.01 Seek and incorporate feedback from team members to refine a solution to a problem.	7.AP.PD.01 Seek and incorporate feedback from team members and users to refine a solution to a problem.	8.AP.PD.01 Seek and incorporate feedback from team members and users to refine a solution to a problem that meets the needs of diverse users.	
		6.AP.PD.02 Incorporate existing code, media, and libraries into original programs and give attribution.	7.AP.PD.02 Incorporate existing code, media, and libraries into original programs of increasing complexity and give attribution.	8.AP.PD.02 Incorporate existing code, media, and libraries into original programs of increasing complexity and give attribution.	
		6.AP.PD.03 Test and refine programs using teacher provided inputs.	7.AP.PD.03 Test and refine programs using a variety of student created inputs.	8.AP.PD.03 Systematically test and refine programs using a range of student created inputs.	
		6.AP.PD.04 Break down tasks and follow an individual timeline when developing a computational artifact.	7.AP.PD.04 Distribute tasks and maintain a project timeline when collaboratively developing computational artifacts.	8.AP.PD.04 Explain how effective communication between participants is required for successful collaboration when developing computational artifacts.	
		6.AP.PD.05 Document text-based programs in order to make them easier to follow, test, and debug.	7.AP.PD.05 Document text-based programs of increasing complexity in order to make them easier to follow, test, and debug.	8.AP.PD.05 Document text-based programs of increasing complexity in order to make them easier to follow, test, and debug.	
	Impacts of Computing	Culture	6.IC.C.01 Explain how computing impacts people's everyday activities.	7.IC.C.01 Assess how computing impacts innovation in other fields.	8.IC.C.01 Describe the trade-offs associated with computing technologies (e.g. automation), explaining their effects on economies and global societies, and explore careers related to the field of computer science.
6.IC.C.02 Identify and discuss the technology proficiencies needed in the classroom and the workplace, and how to meet the needs of diverse users.			7.IC.C.02 Relate the distribution of computing resources in a global society to issues of equity, access, and power.	8.IC.C.02 Evaluate and improve the design of existing technologies to meet the needs of diverse users and increase accessibility and usability.	
				8.IC.C.03 Evaluate how technology can be used to distort, exaggerate, and misrepresent information.	
Social Interactions		6.IC.SI.01 Individually and collaboratively develop and conduct an online survey that seeks input from a broad audience. Describe and use safe, appropriate, and responsible practices (netiquette) when participating in online communities (e.g., discussion groups, blogs, social networking sites).	7.IC.SI.01 Individually and collaboratively use advanced tools to design and create online content (e.g., digital portfolio, multimedia, blog, web page). Describe and use safe, appropriate, and responsible practices (netiquette) when participating in online communities (e.g., discussion groups, blogs, social networking sites).	8.IC.SI.01 Communicate and publish key ideas and details individually or collaboratively in a way that informs, persuades, and/or entertains using a variety of digital tools and media-rich resources. Describe and use safe, appropriate, and responsible practices (netiquette) when participating in online communities (e.g., discussion groups, blogs, social networking sites).	
History					
Safety, Law, & Ethics		6.IC.SLE.01 Differentiate between appropriate and inappropriate content on the Internet, and identify unethical and illegal online behavior.	7.IC.SLE.01 Explain the connection between the longevity of data on the Internet, personal online identity, and personal privacy.	8.IC.SLE.01 Discuss the social impacts and ethical considerations associated with cybersecurity, including the positive and malicious purposes of hacking.	
Community Partnerships		6.IC.CP.01 Communicate with students about the impact of Computer Science across diverse career fields.	7.IC.CP.01 Research with students about the impact of Computer Science across diverse career fields.	8.IC.CP.01 Formulate a solution for a problem or issue by gathering input from local / regional industry members	
Concept	Subconcept	9th & 10th Grade	11th & 12th Grade		

Computing Systems	Devices	L1.CS.D.01 Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects.	L2.CS.D.01 Use abstractions in a collaborative group to make changes to an existing device or program.		
	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.		
		L1.CS.HS.02 Evaluate and utilize a computer system (hardware & software) for a given purpose.	L2.CS.HS.02 Compare computer systems and determine advantages and drawbacks of each system.		
Troubleshooting	L1.CS.T.01 Develop and apply criteria for systematic discovery of errors and systematic strategies for correction of errors 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, latency, topology).		
		L1.NI.C.01 Compare physical and cybersecurity measures by evaluating trade-offs 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.		
	Cybersecurity	L1.NI.C.02 Illustrate how sensitive data can be affected by attacks.	L2.NI.C.02 Use encryption and decryption algorithms to transmit/ receive an encrypted message.		
		L1.NI.C.03 Recommend security measures to address various scenarios based on information security principles.	L2.NI.C.03 Develop and implement network security protocols to protect specific network types.		
		L1.NI.C.04 Explain trade-offs when selecting and implementing cybersecurity recommendations from multiple perspectives such as the user, enterprise, and government.	L2.NI.C.04 Evaluate various case studies in forensic computing.		
Data Analysis	Storage	L1.DA.S.01 Analyze storage types and locations.	L2.DA.S.01 Translate and compare different bit representations of data types, such as characters, numbers, and images.		
		L1.DA.S.02 Evaluate the trade-offs in how data is organized and stored digitally.	L2.DA.S.02 Analyze file systems created for keeping track of files on the hard disk.		
	Collection	L1.DA.C.01 Use tools and techniques to locate, collect, small and largescale data sets.	L2.DA.C.01 Use data analysis tools and techniques to identify patterns from complex real-world data.		
	Visualization & Transformation	L1.DA.C.02 Prepare and create visualizations of small and largescale data sets (e.g. spreadsheets, charts and graphs).	L2.DA.C.02 Create visualizations and generate data sets that use a variety of data collection tools and analysis techniques to support a claim and/or communicate information.		
	Inference & Models	L1.DA.IM.01 Show the relationships between collected data elements using computational models.	L2.DA.IM.01 Use models and simulations to help formulate, refine, and test scientific hypotheses. (e.g. flocking behaviors, life cycles, etc.)		
Algorithms		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.	L2.AP.A.01 Describe how artificial intelligence algorithms drive many software and physical systems (e. g., autonomous robots, computer vision, pattern recognition, text analysis).		
		L1.AP.A.02 Design and develop a software artifact working in a team.	L2.AP.A.02 Develop an artificial intelligence algorithm to play a game against a human opponent or solve a real-world problem.		
		L1.AP.A.03 Design algorithms using sequence, selection, and iteration	L2.AP.A.03 Critically examine and trace classic algorithms (e.g., selection sort, insertion sort, binary search, linear search).		
		L1.AP.A.04 Illustrate the flow of execution of a recursive algorithm.	L2.AP.A.04 Evaluate algorithms (e.g., sorting, searching) in terms of their efficiency and clarity.		
Variables		L1.AP.V.01 Evaluate variable scope and roles to simplify and solve problems. (e.g., global, local, one-way flag, accumulator, etc.)	L2.AP.V.01 Compare and contrast simple data structures and their uses (e.g., lists, stacks, queues).		

Algorithms and Programs	Control	L1.AP.V.02 Differentiate variable types and reasoning for each use. (e.g., int, float, boolean)	L2.AP.V.02 Demonstrate the use of lists (e.g., arrays) to simplify solutions, generalizing computational problems instead of repeatedly using primitive variables.		
		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.	L2.AP.C.01 Trace the execution of repetition (e.g., loops, recursion), illustrating output and changes in values of named variables.		
		L1.AP.M.01 Break down a solution into procedures using systematic analysis and design.	L2.AP.M.01 Construct solutions to problems using multiple objects from one class.		
	Modularity	L1.AP.M.02 Create computational artifacts by systematically organizing, manipulating and/or processing data.	L2.AP.M.02 Design or redesign a solution to a largescale computational problem by identifying generalizable patterns.		
		L1.AP.M.03 Construct solutions to problems using student-created components (e.g., procedures, modules, objects).	L2.AP.M.03 Create programming solutions by reusing existing code (e.g., libraries, Application Programming Interface (APIs), code repositories).		
		L1.AP.PD.01 Create software by analyzing a problem and/or process, developing and documenting a solution, testing outcomes, and adapting the program for a variety of users.	L2.AP.PD.01 Create software that will provide solutions to a variety of users using the software life cycle process		
	Program Development	L1.AP.M.01 Break down a solution into procedures using systematic analysis and design.	L2.AP.PD.02 Design software in a project team environment using integrated development environments (IDEs), versioning systems, and collaboration systems.		
		L1.AP.PD.03 While working in a diverse team, develop, test, and refine event-based programs that solve practical problems or allow self expression.	This could also address the difference between procedural coding and OOP.		
		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.	L2.AP.PD.04 Systematically check code for correctness, usability, readability, efficiency, portability, and scalability through peer review.		
		L1.AP.PD.05 Evaluate and refine computational artifacts to make them more user-friendly, efficient and/or accessible.	L2.AP.PD.05 Develop and use a series of test cases to verify that a program performs according to its design specifications.		
L2.AP.PD.06 Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).		L2.AP.PD.06 Explain security issues that might lead to compromised computer programs.			
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 among people of various cultures.	L2.IC.SI.01 Compare and contrast the benefits and drawbacks of social media.		
	History	L1.IC.H.01 Hypothesize the impact of the innovations of computing systems for the next decade.	L2.IC.H.01 Analyze trends of computing and how those trends have changed over time.		
	Safety, Law, &	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.		
		L1.IC.SLE.02 Compare and contrast various software licensing schemes (e.g. open source, freeware, commercial)	L2.IC.SLE.02 Determine ways to test the validity of information located online.		

Ethics	Ethics	L1.IC.SLE.03 Explain the privacy concerns related to the large-scale collection and analysis of information about individuals (e.g., how businesses, social media, and the government collects and uses data) that may not be evident to users.	L2.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.		
	Community Partnerships	L1.IC.CP.01 Explore computing systems in local industries.	L2.IC.CP.01 Collaborate with local industry partners to design and implement a viable mentorship.		