

Missouri Standards Alignment with CodeX Curriculum

By the end of eighth grade:	Unit 1	Unit 2	Unit 3
Computing Systems			
5.CS.HS.01 Model that information is translated into bits in order to transmit and process between software to accomplish tasks.			
5.CS.T.01 Identify, using accurate terminology, simple hardware and software problems that may occur during everyday use. Discuss problems with peers and adults, apply strategies for solving these problems and explain why the strategy should work.			
Networks & the Internet			
5.NI.NCO.01 Model how information is broken down into packets, transmitted through multiple computing devices over networks and the internet and reassembled at the destination.			
5.NI.C.01 Analyze the credibility of digital information (e.g., comparing multiplatform accounts and sources, the author's point of view).			
5.NI.C.02 Discuss cybersecurity problems caused by information that is published for different reasons (e.g., inform, advertise, persuade, harm).			
Data and Analysis			
5.DA.S.01 Evaluate trade-offs, including availability and quality, based on the type of file, storage requirements (e.g., file size, availability, available memory) and sharing requirements.			
5.DA.CVT.01 Organize and present collected data to highlight comparisons and support a claim.			
5.DA.IM.01 Use reliable data to highlight or propose cause and effect relationships, predict outcomes or communicate an idea.			
Algorithms and Programming			
5.APA.01 Compare and simplify multiple algorithms (sets of step-by-step instructions) for accomplishing the same task verbally and kinesthetically, with robot devices or a programming language, then determine which is the most efficient.			
5.AP.V.01 Create programs that use variables to store and modify grade level appropriate data.	[1]		
5.APC.01 Create a program using control structures (e.g., sequence, conditionals, interactive-looping), event handlers and variables to solve a problem or express ideas both independently and collaboratively.	[2]		
5.APM.01 Decompose (break down) large problems into smaller, manageable sub-problems and then into a precise sequence of instructions.	[3]		
5.APM.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]		
5.APPD.01 Use an iterative and collaborative process to plan the development of a program that includes other perspectives and user preferences while solving simple problems.			
5.APPD.02 Observe intellectual property rights and give appropriate attribution when creating or remixing programs.			
5.APPD.03 Analyze, examine, create and debug a program that includes sequencing, repetition, conditionals and variables in a programming language	[5]		
5.APPD.04 Communicate and explain your program development using comments, presentations and interactive demonstrations.	[6]		
Impacts of Computing			
5.IC.C.01 Give examples and explain how computing technologies have changed the world and express how computing technologies influence, and are influenced by, cultural practices.			
5.IC.C.02 Develop, test and refine digital artifacts to improve accessibility and usability.			
5.IC.SI.01 Develop a code of conduct, explain and practice grade-level appropriate behavior and responsibilities while participating in an online community (e.g., talking safely online, promoting good digital citizens, privacy settings, cyberbullying). Identify and report inappropriate behavior and know how to report cyberbullying.			
5.IC.SLE.01 Observe intellectual property rights and give appropriate credit when using resources.			
5. IC.SLE.02 Continue to discuss and understand the implications of positive and negative digital footprints and that they never go away.			

Missouri Standards Alignment with CodeX Curriculum

By the end of eighth grade:	Unit 1	Unit 2	Unit 3
Computing Systems			
6-8.CS.D.01 Evaluate the design of computing devices, based on the characteristics of each device and how users interact with it, to improve the overall user experience			
6-8.CS.HS.01 Design projects that combine hardware and software components to collect and exchange data.			
6-8.CS.T.01 Develop a systematic troubleshooting routine to identify the problem, research solutions and fix problems with computing devices, components and software.	[7]		
Networks & the Internet			
6-8.NI.NCO.01 Model the different ways that data is transferred across a network and the protocols used to transmit the data.			
6-8.NI.C.01 Recognize and determine computer threats and be able to identify programs and methods to protect electronic information.			
6-8.NI.C.02 Demonstrate how data is transmitted through multiple methods of encryption.			
Data and Analysis			
6-8.DA.S.01 Represent data using multiple encoding schemes.			
6-8.DA.VT.01 Collect data using computational tools and display it for the end user in an easy to understand way.			
Algorithms and Programming			
6-8.DA.IM.01 Analyze methods to refine computational models based on received data.			
6-8.APA.01 Design algorithms with flow charts and/or pseudocode to show solutions to complex problems.	[8]		
6-8.AP.V.01 Create clearly named variables to store and manipulate information.	[9]		
6-8.APC.01 Design and develop combinations of control structures, nested loops and compound conditionals.	[10]		
6-8.APM.01 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.			
6-8.APM.02 Create procedures with parameters to organize code and make it easier to reuse.		[11]	
6-8.AP.PD.01 Use flowcharts and/or pseudocode to solve problems using algorithms.	[12]		
6-8.AP.PD.02 Use feedback from team members and users to refine solutions to meet user needs.			
6-8.AP.PD.03 Give proper attribution to code, media, etc. that is used in their programs.			
6-8.AP.PD.04 Systematically test and refine programs using a range of test cases.			
6-8.AP.PD.05 Manage project tasks and timelines when collaboratively developing computational artifacts.			
Impacts of Computing			
6-8.IC.C.01 Compare tradeoffs associated with computing technologies that have impacted people's activities, careers and lives when solving global problems using the power of computing.			
6-8.IC.C.02 Discuss issues of bias and accessibility in the design of existing technologies.			
6-8.IC.SI.01 Collaborate with many contributors through strategies such as crowdsourcing or surveys when creating a computational artifact.			
6-8.IC.SLE.01 Describe tradeoffs between allowing information to be public and keeping information private and secure.			

Missouri Standards Alignment with CodeX Curriculum

	Unit 1	Unit 2	Unit 3
Computing Systems			
9-10.CS.D.01 Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects.			
9-10.CS.HS.01 Explain the abstraction and interactions between application software, system software and hardware.			
9-10.CS.T.01 Develop, communicate and apply systematic troubleshooting strategies for correction of errors in computing systems.	[13]		
Networks & the Internet			
9-10.NI.NC0.01 Evaluate the scalability and reliability of networks by identifying and illustrating the basic components of computer networks (e.g., routers, switches, servers) and network protocols (e.g., IP, DNS).			
9-10.NI.NC0.02 Describe the issues that impact network functionality (e.g., bandwidth, load, delay, topology).			
9-10.NI.C.01 Compare physical and cybersecurity measures by evaluating trade-offs between the usability and security of a computing system.			
9-10.NI.C.02 Illustrate how sensitive data can be affected by attacks.			
9-10.NI.C.03 Recommend security measures to address various scenarios based on information security principles.			
9-10.NI.C.04 Explain trade-offs when selecting and implementing cybersecurity recommendations from multiple perspectives such as the user, enterprise and government.			
Data and Analysis			
9-10.DA.S.01 Translate and compare different bit representations of data types, such as characters, numbers and images.			
9-10.DA.S.02 Evaluate the trade-offs in how data is organized and stored digitally.			
9-10.DA.CVT.01 Create data visualizations to help others better understand real-world phenomena.			
9-10.DA.CVT.02 Explain the insights and knowledge gained from digitally processed data by using appropriate visualizations, notions and precise language.			
9-10.DA.CVT.03 Evaluate and refine computational artifacts to make them more usable and accessible.			
9-10.DA.IM.01 Show the relationships between collected data elements using computational models.			
9-10.DA.IM.02 Refine computational models to better represent the relationships among different elements of data collected from a phenomenon or process.			
Algorithms and Programming			
9-10.APA.01 Create a prototype that uses algorithms (e.g., searching, sorting, finding shortest distance) to provide a possible solution for a real-world problem.	[14]		
9-10.APV.01 Create problem solutions that utilize primitive variables (e.g., strings, ints, Booleans, doubles).	[15]		
9-10.APV.02 Demonstrate the use of advanced variables (e.g., lists, arrays, objects) to simplify solutions, generalizing computational problems instead of repeatedly using primitive variables.		[16]	
9-10.APC.01 Apply the concepts of specific control structures (e.g., sequence, conditionals, repetition, procedures) considering program efficiencies such as readability, performance and memory usage.			
9-10.APM.01 Break down a solution into procedures using systematic analysis and design utilizing functional abstraction.			
9-10.APM.02 Create computational artifacts (file, graphic, video, audio) by systematically organizing, manipulating and/or processing data.			
9-10.APPD.01 Using visual aids and documentation, illustrate the design elements and data flow (e.g., flowcharts, pseudocode) of the development of a program.	[17]		
9-10.APPD.02 Create a program by analyzing a problem and/or process, developing and documenting a solution, testing outcomes, debugging errors and adapting the program for a variety of users.			
9-10.APPD.04 While collaborating in a team, develop, test and refine programs that solve practical problems or allow selfexpression			
9-10.APPD.05 Evaluate and refine computational artifacts to make them more user-friendly, efficient and/or accessible.			
Impacts of Computing			
9-10.IC.C.01 Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.			
9-10.IC.C.02 Test and refine computational artifacts to reduce bias and equity deficits.			
9-10.IC.C.03 Demonstrate ways a given algorithm applies to problems across disciplines.			
9-10.IC.SI.01 Demonstrate through collaboration on a project how computing increases connectivity among people of various cultures			
9-10.IC.SI.02 Explain how the degrees of communication afforded by computing have impacted the nature and content of career fields.			
9-10.IC.SLE.01 Explain the beneficial and harmful effects that intellectual property laws can have on innovation.			
9-10.IC.SLE.02 Explain the privacy concerns related to the collection and analysis of information about individuals that may not be evident to users.			
9-10.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.			
9-10.APSLE.04 Define and classify a variety of software licensing schemes (e.g., open source, freeware, commercial) and discuss the advantages and disadvantages of each scheme in software development.			
9-10.IC.SLE.05 Identify and explain the potential impacts and implications of emerging technologies on larger social economic and political structures with evidence from credible sources.			

- [1] 3.8 introduces the use of variables
- [2] Mission 4 begins the use of all of these
- [3] These are the flowcharts and pseudocodes introduced in the teachers' manual
- [4] Remixes are introduced in Mission 4
- [5] These start in Mission 4
- [6] 5.5 introduces comments
Pseudocodes and flowcharts are introduced in the teachers' manual
- [7] Mission 2 introduces troubleshooting techniques as does the teachers' manual
- [8] These are introduced in the teachers' manual
- [9] 3.8 introduces the use of variables
5.5 discusses descriptive naming of variables
- [10] Mission 6 introduces but does not discuss nested loops
Mission 9 introduces compound conditionals
- [11] Mission 7 begins the discussion on code reuse
Mission 10 describes it even more and discusses organizing code to make the reuse easier
- [12] These are introduced in the teachers' manual
- [13] Mission 2 and the teachers' manual discuss troubleshooting
- [14] These are the remixes that are introduced in Mission 4
- [15] Mission 4 introduces different data types for variable use
- [16] 7.5 introduces the use of lists
- [17] These are introduced in the teachers' manuals