Course Title: Introduction to Computer Science (formally referred to as STEM)
Content Area: Computer Science
Grade Level: 8th-Grade

Scope and Sequence

Unit or Topic	CSTA Standards & Practices	Length of Time	Key Content	Assessment Tools	Scaffolding Strategies (Interventions, Special Education)	Resources & Materials
What is Computing?	1B-CS-01 (P7.2) 1B-CS-02 (P4.4) 1B-CS-03 (P6.2) 1B-IC-18 (P7.1) 2-CS-02 (P.5.1) 2-CS-03 (P6.2) 2-IC-20 (P7.2)	2-3 weeks	• Explore the concepts of *computer* and *computing* • Explore the history of computers and research famous computer innovators • Explore roles that computers play in our lives • Demystify and learn the function of the parts of a personal computer – learn the terminology of *hardware* components necessary for the purchase of a personal computer, hardware vs. *software*, *input* and *output* devices • Troubleshooting - identify and fix an issue with a computing device • Explore the idea of *intelligence* • Explore the future of computing – examples: artificial intelligence, new ways of storing data etc.	**Diagnostic** Entry/Exit Tickets **Formative** Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share **Summative** Teacher-Created Assessments, Checklists, Performance-Based Projects, Rubrics	- Hands-on Models/Demonstrations - Describe concepts in multiple ways - Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	OneNote Teacher-created lesson plans, activities, supplements and assessments model desktop PCs (at least 6 of the same model) relevant technology news articles and web 2.0 tools
Problem-Solving & Data	2-DA-07 (P4.0) 3A-DA-07 (P4.1)	2-3 weeks	• Introduce data collection and problem-solving • Apply and reinforce the 4-step problem-solving process	**Diagnostic** Entry/Exit Tickets **Formative**	- Hands-on Models/Demonstrations - Describe concepts in multiple ways	OneNote Teacher-created lesson plans, activities,

			• Explore how digital data is represented and encoded • Count in binary • Convert from binary to decimal • Encode characters in binary using ACSII codes • Introduce hexadecimal system • Create and manipulate digital images (pixel images) • Introduce RGB and mix levels of RGB • Introduce linear and binary algorithms • Explore sorted/unsorted lists and sorting algorithms • Introduce minimal spanning trees and how graphic can be used to solve problems	Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share **Summative** Teacher-Created Assessments, Checklists, Performance-Based Projects, Rubrics	- Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	supplements and assessments Programmable LED matrices relevant web 2.0 tools
Cryptography	2-NI-05 (P.7.2) 2-NI-06 (P4.4) 2-IC-20 (P7.2) 2-IC-23 (P7.2)	2-3 weeks	• Cryptography, Cryptology, Cryptanalysis – Explore the need for privacy and security • Explain general encryption with data and keys • History of Cryptography – beginnings, substitution ciphers, changes over time, 256-bit encryption, explore the Enigma • Data Encryption – explore the CIA triad • Explore, apply, decrypt and crack Caesar Ciphers • Explore Vigenere Ciphers with JavaScript program	**Diagnostic** Entry/Exit Tickets **Formative** Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share **Summative** Teacher-Created Assessments,	- Hands-on Models/Demonstrations - Describe concepts in multiple ways - Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	OneNote Teacher-created lesson plans, activities, supplements and assessments relevant web 2.0 tools

				Checklists, Performance-Based Projects, Rubrics		
Introduction to Programming & Designing For Impact	2-AP-10 (P4.4, P4.1) 2-AP-11 (P5.1, P5.2) 2-AP-13 (P3.2) 2-AP-14 (P.4.1, P4.3) 2-AP-16 (P4.2, P5.2, P7.3) 2-AP-17 (P6.1) 2-AP-18 (P2.2) 2-IC-21 (P1.2) 2-IC-22 (P2.4, P5.2)	5-6 weeks	• Introduce the concept of a computer program as a set of instructions • Introduce the Block programming language, including the basic terms utilized in the language • Practice using the basic features of App Inventor in the context of creating a simple program • Practice the concept of event-driven programming • Explore and practice debugging and refactoring strategies • Introduce and apply the coding concepts of loops, variables, user input, conditionals, parameters and AND/OR and randomness • Introduce and apply the concept of abstraction and custom blocks or functions • Creating an engaging interface • Apply the Design Process – Empathy, Define, Ideate, Prototype, Build • Explore designing for impact – accessibility issues, rapid prototyping, user testing, digital surveys	**Diagnostic** Entry/Exit Tickets **Formative** Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share **Summative** Teacher-Created Assessments, Checklists, Performance-Based Projects, Rubrics	- Hands-on Models/Demonstrations - Describe concepts in multiple ways - Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	OneNote Teacher-created lesson plans, activities, supplements and assessments Access to MIT App Inventor on teacher laptop, all student laptops, and all classroom android tablets

			• Build an app to solve a problem			
Introduction to Robotics with Raspberry Pi	1A-AP-08 (P4.4) 1A-AP-11 (P3.2) 1A-AP-15 (P.7.2) 1A-CS-02 (P7.2) 1B-AP-09 (P5.2) 1B-AP-11 (P3.2) 1B-AP-17 (P7.2) 1B-CS-01 (P7.2) 2-AP-10 (4.4, P4.1) 2-AP-11 (P5.1, P5.2) 2-AP-13 (P3.2) 2-AP-19 (P7.2)	5-6 weeks	• Introduce physical computing • Explore robotics – identify the criteria that makes a machine a robot • Apply goal setting, code commenting, and pseudocode • Analog vs. Digital • Extend programming concepts with Raspberry Pi builds • Connecting external components with Raspberry Pi • CNC Farming with FarmBot	**Diagnostic** Entry/Exit Tickets **Formative** Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share **Summative** Teacher-Created Assessments, Checklists, Performance-Based Projects, Rubrics	- Hands-on Models/Demonstrations - Describe concepts in multiple ways - Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	OneNote Teacher-created lesson plans, activities, supplements and assessments Class pack of Raspberry Pi and assorted components (at least 13 sets) Access to FarmBot software on teacher laptop and all student laptops
The Internet	2-NI-04 (P4.4) 2-IC-20 (P7.2) 2-IC-23 (P7.2)	2-3 weeks	• Introduce the structure of the internet • Explore the difference between IPv4 and IPv6 – why are we running out of addresses? • Explore the different levels of the internet • Trace a website request from a server, through the network, and to your computer • Data transmission – How are internet packets able to	**Diagnostic** Entry/Exit Tickets **Formative** Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share	- Hands-on Models/Demonstrations - Describe concepts in multiple ways - Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	OneNote Teacher-created lesson plans, activities, supplements and assessments Relevant technology news articles and web 2.0 tools

			find their way onto your computer? • Explain how a request from a computer travels through the various levels of servers to reach and return the correct webpages and resources • Hardware – Explore the role of routers, Importance of protocols, Explore how data is able to be transmitted across the ocean by underwater cables	**Summative** Teacher-Created Assessments, Checklists, Performance-Based Projects, Rubrics		
Digital Citizenship & Cyber Hygiene (optional)	2-IN-04 (P4.4) 2-IC-20 (P7.2) 2-IC-21 (P1.2) 2-IC-23 (P7.2)	1-2 weeks	• Digital footprint and reputation • Cyberbullying • Internet Safety • Privacy and Security	**Diagnostic** Entry/Exit Tickets **Formative** Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share **Summative** Teacher-Created Assessments, Checklists, Performance-Based Projects, Rubrics	- Hands-on Models/Demonstrations - Describe concepts in multiple ways - Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	OneNote Teacher-created lesson plans, activities, supplements and assessments relevant technology news articles and web 2.0 tools
Web Design (optional)	2-NI-04 (P.4.4)	2-3 weeks	• Creating webpages using HTML including links,	**Diagnostic**	- Hands-on Models/Demonstrations	OneNote

			images, lists, tables, and inline styling • Styling webpages with CSS including CSS rules, CSS classes, and CSS IDs • Exploring how webpages are requested and delivered • Building a personal portfolio website	Entry/Exit Tickets **Formative** Student Observations, Homework, Reflection Journals, Discussions, Informal Presentations, Projects, Think/Pair/Share **Summative** Teacher-Created Assessments, Checklists, Performance-Based Projects, Rubrics	- Describe concepts in multiple ways - Breaking down large tasks into smaller chunks - Adjust pacing - Visual Aids - Front loading new vocabulary - Activating prior knowledge - Peer-to-peer discussion time - Guided and Independent Practice - Check for understanding - Graphic Organizers	Teacher-created lesson plans, activities, supplements and assessments Access to Adobe Dreamweaver or equivalent on teacher laptop and all student laptops