Exploring Computer Science Scope & Sequence

Days	Unit	Standard(s)/Outcome(s)	Essential/Guiding Questions
9	*1 - Basics of the Law*	*1. Identify consequences of unethical and illegal conduct.* *2. Compare/contrast the national constitution with state constitutions and local charters.* *3. Analyze specific cases by stating the facts, finding the legal questions, applying the laws, and resolving the issues.* *4. Differentiate between cases that belong within the jurisdiction of the federal and state court systems.* *5. Compare the role of the juvenile court with the role of other courts within a state.* *6. Explain the role of the national and state appellate courts.*	*1. How will an understanding of the U.S. legal system impact my life?* *2. What are the potential outcomes of ethical behavior? Unethical behavior?* *3. What is the importance of the historic perspective when examining the structure of the U.S. court system?*
8	**1 – Problem Solving and Computers**	*1. Identify the four steps of the problem solving process* *2. Apply the problem solving process* *3. Identify what makes a computer*	

		4. *Identify the inputs and outputs of common computing devices* 5. *Develop an algorithm for processing information* 6. *Determine a possible source of a given input, and why that input should be stored on a device* 7. *Design an app to solve a problem* **Intro to Problem Solving** **The Problem Solving Process** **Exploring Problem Solving** **What is a Computer?** **Input and Output** **Processing** **Apps and Storage** **Project: Propose an App**	
14	**2 – Web Development**	1. *Identify the purpose of a website from the perspective of both users and creators.* 2. *Identify websites as a form of personal expression* 3. *Identify the purpose of html and its role in web design, and how to build our skills in Web Lab.* 4. *Explain how to use html tags*	

		to change a website appearance and structure content. 5. *Discern what is safe to publish and share online.* 6. *Use list tags* 7. *Explain the nature of copyright and how we can use it.* 8. *Experience how proper formatting and documentation is used in html to improve communication and bug finding.* 9. *Explore techniques for linking multiple web pages together.* 10. *Describe CSS and how to use it effectively in web design.* 11. *Use CSS to change sizes, positions, and create element borders through rules that govern the entire website.* 12. *Engage with web search techniques and evaluating website trustworthiness.* 13. *Apply color and style to a web site* 14. *Implement incremental improvements to a web site* **Exploring Websites**	

		Websites for Expression **Intro to HTML** **Headers** **Digital Footprint** **Lists** **Intellectual Property** **Clean Coding and Debugging** **Project: Multipage Website** **Styling Text with CSS** **Styling Elements with CSS** **Sources and Search Engines** **RGB Colors and Classes** **Project: Personal Portfolio Website**	
18	**3 – Animations and Games**	1. *Identify how Computer Science is used in a field of entertainment* 2. *Communicate how to draw an image in Game Lab, accounting for shape position, color, and order* 3. *Use the Game Lab IDE to plot different colored shapes on the screen and overlap them* 4. *Use and reason about drawing commands with multiple parameters and use random numbers in a program* 5. *Identify and use a variable as a way*	

		to label and reference a value in a program 6. *Assign a sprite to a variable and update its properties* 7. *Animate a sprite.* 8. *Describe the connection between updating a sprite's location properties and sprite movement on the screen* 9. *Explain how a specific movement is achieved* 10. *Develop complex Boolean statements* 11. *Use conditionals to react to changes in variables and sprite properties* 12. *Demonstrate how to move sprites in response to keyboard input* 13. *Differentiate between conditions that are true once per interaction, and those that remain true through the duration of an interaction* 14. *Demonstrate the programing skills you have acquired* 15. *Explain the advantages of simplifying code using higher level blocks* 16. *Illustrate how abstractions help to manage the*	

		complexity of code 17. *Explain how individual programming constructs can be combined to create more complex behavior* 18. *Illustrate how abstractions can be built upon to develop even further abstractions* 19. *Create and use functions in a program to improve the readability of their programs* 20. *Demonstrate knowledge of core programming constructs necessary to build different components of a game* 21. *Develop and implement a plan for creating a piece of software* **Programming for Entertainment** **Plotting Shapes** **Drawing in the Game Lab** **Parameters and Randomization** **Variables** **Sprites** **The Draw Loop** **Counter Pattern (Unplugged)** **Moving Sprites** **Booleans (Unplugged)** **Booleans and Conditionals** **Conditionals and User Input**	

		Complex Conditionals **Project: Interactive Card** **Velocity** **Collision Detection** **Complex Sprite Movement** **Collisions** **Functions** **The Game Design Process** **Project: Design/Build a Game**	
16	**4 – The Design Process**	1. *Evaluate an object for how well its design meets a given set of needs* 2. *Critique a design through the perspective of a user profile* 3. *Create meaningful categories from a collection of ideas* 4. *Identify the user needs a prototype was designed to address* 5. *Categorize and prioritize user feedback for an app* 6. *Analyze interview notes to develop follow-up questions* 7. *Design the functionality of an app to address the specific needs of a user* 8. *Identify ways in which apps can effect social change* 9. *Identify the user needs being addressed by an app*	

		10. *Communicate the design and intended use of program* 11. *Test a prototype with a user, recording the results* 12. *Translate a paper prototype into a digital format* 13. *Write programs that respond to user input* 14. *Run a user test on an app and record what users say about their minimum viable product* 15. *Prioritize the bugs and features according to impact and ease of implementation* 16. *Present technical information clearly to non-technical users* **Analysis of Design** **Understanding Your User** **User-Centered Design** **User Interfaces** **Feedback and Testing** **Identifying Users Needs** **Project: Paper Prototype** **Designing Apps for Good** **Market Research** **Paper Prototypes** **Prototype Testing** **Digital Design** **Linking Screens** **Testing the App**	

		Improving and Iterating **Project: App Presentation**	
10	**5 – Data and Society**	1. *Provide examples of how representing data in different ways can affect its ability to solve different problems* 2. *Create, use, and provide feedback on a system for representing information* 3. *Use the ASCII system to encode and decode text information in binary* 4. *Create and manipulate binary patterns to represent black and white images* 5. *Extend a representation system based on patterns.* 6. *Apply a method of encryption to ensure the secure transmission of data.* 7. *Use multiple binary systems to decode information.* 8. *Create a generalized representation system for many instances of a complex type of information* 9. *Identify and collect relevant data to help solve a problem* 10. *Distinguish between data that users intentionally and*	

		unintentionally produce. 11. *Explain why a set of data must be cleaned before a computer can use it.* 12. *Use tables and visualizations summarizing data to support a decision* 13. *Use cross tabulation to find patterns and relationships in data* 14. *Design an algorithm for making decisions using data as inputs* 15. *Determine appropriate sources of data needed to solve a problem* **Representation Matters** **Patterns and Representations** **ASCII and Binary Representations** **Representing Images** **Combining Representations** **Create a Representation** **Problem Solving and Data** **Problem Solving with Big Data** **Structuring Data** **Making Decisions with Data** **Interpreting Data** **Automating Data Decisions** **Project: Make a Recommendation**	

18	**6 – Physical Computing**	1. *Identify computing innovations within a given field* 2. *Write programs that change multiple elements on a single screen instead of changing screens* 3. *Use code to control a physical device* 4. *Attach an event handler to a hardware input* 5. *Develop programs that respond to analog input* 6. *Create a function that uses parameters to generalize behavior* 7. *Access an element in an array using its index* 8. *Create and modify an array* 9. *Modify the exit condition of a for loop to control how many times it repeats* 10. *Recognize the use and need for accelerometer orientation* 11. *Use parameters to generalize the purpose of a function* 12. *Create and debug simple circuits* 13. *Develop an interactive physical prototype that combines software and*	

		hardware 14. *Independently scope the features of a piece of software* 15. *Prototype a physical computing device* 16. *Implement a plan for developing a piece of software that integrates hardware inputs and outputs* **Innovations in Computing** **Designing Screens with Code** **The Circuit Playground** **Input (Unplugged)** **Board Events** **Getting Properties** **Analog Input** **The Program Design Process** **Project: Make a Game** **Arrays and Color LEDs** **Making Music** **Arrays and For Loops** **Accelerometer** **Functions with Parameters** **Circuits and Physical Prototypes** **Project: Prototype an Innovation**	

		Last Days **Arduino / Edison Robots Exploration** **Review** **Final Exam**	