CS Standards:

- K-2
- 3-5
- 6-8

K-2

Big Ideas

It seems like K-2 can mostly be worked into existing curriculum for other subjects. It's mostly about students learning to use technology and use it responsibly.
Getting students started early on using the Google suite and digital collaboration meets most of these.

Data and Information (DI)

K-2.DI.1 Use technology resources to solve age-appropriate problems and communicate thoughts, ideas, or stories in a step-by-step manner.

- Create simple scratch stories with conversations and sequencing
    - **Animate a Name Scratch Project** (plugged)
    - **Five Things About Me** (plugged)
    - **Get Loopy** (plugged)

K-2.DI.2 Understand how to arrange (sort) information into useful order, such as sorting students by birth date, without using a computer.

- **The sorting activity done with Michelle on day 1**

K-2.DI.3 Recognize that software is created to control computer operations.

- **The "What is coding?" discussion that we had on day 2**
- **Covered when we introduced sequencing**
    - Sequencing is all about telling the computer what to do *in the order we want it to happen*. Computers only do what we tell them to do. Students can see this when they build different projects in Scratch. The computer only does what you tell it to do.
    - **Drawing Activity** (unplugged)

Computing Devices and Systems (CD)

K-2.CD.1 Use standard input and output devices to operate computers and other Technologies.

- **This happens every time students use a computer or a digital device.**

Programs and Algorithms (PA)

K-2.PA.1 Use technology and developmentally appropriate multimedia resources to conduct age-appropriate research and support learning across the curriculum.

- Use Google to find images and videos for a presentation about a topic
    - **Instead of 5 Things About Me, create 5 Things About Genetics**
    - **Instead of Animate a Name, animate a spelling or vocab word**

K-2.PA.2 Create developmentally appropriate multimedia products with support from teachers, family members, or student partners.

- **Create a Scratch project**
    - Multimedia is content that uses a combination of different content forms such as text, audio, images, animations, video and interactive content.
    - Incorporate other subject areas in your Scratch project.
        - Examples:
            - Research a science topic
            - Code a Project with Spanish text
            - History topic/story - retold in a digital format
    - Science fair, digital style
- Additional option:
    - Create Google Slides or videos

K-2.PA.3 Arrange information using concept mapping tools and a set of statements that accomplish a simple task.

- **Peanut Butter Sandwich** (unplugged)
- **Drawing Activity** (unplugged)

Networking and Communication (NC)

K-2.NC.1 Use technology to work cooperatively and collaboratively with peers, teachers, and Others.

- **Paired Programming**

K-2.NC.2 Gather information and communicate electronically with others with support from teachers, family members, or student partners.

- Send basic emails?
- Make a Google form?
- CMS (Class Dojo?)
- Scratch comments on projects

Impact and Culture (IC)

K-2.IC.1 Practice responsible digital citizenship (legal and ethical behaviors) in the use of Technology.

- **Routine discussions about online safety, digital citizenship, and digital literacy**
- **Digital resource: https://beinternetawesome.withgoogle.com/en_us**
- Role playing about digital citizenship behaviors (unplugged)

K-2.IC.2 Identify positive and negative social and ethical behaviors for using technology.

- **Routine discussions about online safety, digital citizenship, and digital literacy**
- **Digital resource: https://beinternetawesome.withgoogle.com/en_us**

3-5

Big Ideas

Here we start to introduce more of the actual ideas behind tech. Binary, algorithms, programming, networking, the internet, AI. Students will need to start to create programs, and use digital tools much more heavily. A fair amount can still be worked into other classes, but there are some specific programming things that will need their own explicit time.

Data and Information (DI)

3-5.DI.1 Understand and use the basic steps in algorithmic problem solving (e.g., problem statement and exploration, examination of sample instances, design, implementation, and testing).

- **Walk a Chair** (unplugged)

3-5.DI.2 Develop a simple understanding of an algorithm (e.g., search, sequence of events, or sorting) using computer-free exercises.

- **Simon Says activity** (unplugged)
- **Cup Stacking** (unplugged)
- **Walk a Chair** (unplugged)

3-5.CD.3 Apply troubleshooting strategies for identifying simple hardware and software problems that may occur during use.

- **Troubleshooting difficulties in Scratch**

3-5.DI.4 Describe how a simulation can be used to solve a problem.

- Simulation software?

3-5.DI.5 Understand the connections between computer science and other fields.

- Brainstorm and show connections
- Computers are part of everything

Computing Devices and Systems (CD)

3-5.CD.1 Demonstrate proficiency with keyboards and other input and output devices.

- Typing skills practice
- Voice to text practice (google docs, phones, etc…)
- Daily computer/device use fulfills this

3-5.CD.2 Understand the pervasiveness of computers and computing in daily life (e.g., voicemail,
downloading videos and audio files, microwave ovens, thermostats, wireless Internet, mobile computing devices, GPS systems).

- Discuss IoT
- Show examples of IoT
- Think of what life would look like without computers
- Project where students identify every computer they interact with for a whole day
- Discussion of the development and spread of computer technology from 1960s to today

3-5.CD.3 Apply troubleshooting strategies for identifying simple hardware and software problems that may occur during use.

- Create a help guide for older people to use technology
- Teach students to help their parents/grandparents troubleshoot
- Daily working with technology helps meet this
- This is a behavior that teachers can model for students every day, and is more of an implicitly taught standard than an explicitly taught standard

3-5.CD.4 Recognize that computers model intelligent behavior (as found in robotics, speech and language recognition, and computer animation).

- Discuss neural networks?
- AI developments?
- Discuss the limitations of computer's intelligent behavior and what humans are better than computers at (image and pattern recognition)

- Talk about/demonstrate some AI (like Siri, Cortana) and compare/contrast them to humans.

Programs and Algorithms (PA)

3-5.PA.1 Use technology resources (e.g., calculators, data collection probes, mobile devices, videos, educational software, and web tools) for problem-solving and self-directed learning, and general-purpose productivity tools and peripherals to support personal productivity, remediate skill deficits, facilitate learning, and individual/collaborative writing, communication, and publishing activities.

- **Create a Scratch project.**
- **Code.org**

3-5.PA.2 Use digital tools to gather, manipulate, and modify data for use by a program.

- Could use a survey in Google Forms
- Could use Vernier probes
- Clickers to count people
- Internet use data?

3-5.PA.3 Implement problem solutions using a block-based visual programming language.

- **Use Scratch to create programs**

Networking and Communication (NC)

3-5.NC.1 Use online resources (e.g., email, online discussions, collaborative web environments) to participate in collaborative problem-solving activities for the purpose of developing solutions or products.

- **Use Scratch to create programs**
- Other options:
    - Trello for designing
    - Teach students to write emails
    - Google Docs

3-5.NC.2 Use productivity technology tools (e.g., word processing, spreadsheet, presentation software) for individual and collaborative writing, communication, and publishing activities.

- Using the Google suite (or Microsoft, or Apple, or whatever system) in daily class work meets this standard
- Google Docs
- Google Sheets
- Google Slides
- Google Drive

Impact and Culture (IC)

3-5.IC.1 Discuss basic issues related to responsible use of technology and information, and the consequences of inappropriate use.

- **Routine discussions about online safety, digital citizenship, and digital literacy**
- **Digital resource: https://beinternetawesome.withgoogle.com/en_us**
- Expansion opportunity:
    - Discuss news articles about tech abuses

3-5.IC.2 Identify the impact of technology (e.g., social networking, cyber bullying, mobile computing and communication, web technologies, cyber security, and virtualization) on personal life and society.

- Research impacts of Snapchat/social networking apps
- Read about how screens affect the brain
- Research the history of computing
- **Be Internet Awesome touches on some of these things**

3-5.IC.3 Evaluate the accuracy, relevance, appropriateness, comprehensiveness, and biases that occur in electronic information sources.

- Fake news investigation
- Teach skills of filtering for fake news and scams
- **Be Internet Awesome touches on some of these things**

3-5.IC.4 Understand ethical issues that relate to computers and networks (e.g., equity of access, security, privacy, copyright, and intellectual property).

- Discuss security and privacy tradeoff
- Discuss what stealing/ownership looks like in a digital world
- **Be Internet Awesome touches on some of these things**

6-8

Big Ideas

It seems like some of these standards can be met in other classes like math/science, but these require a dedicated class time. How do computers work, how to program/think algorithmically, giving and receiving feedback, creating programs. Probably could be met in 1 semester of each year.

Data and Information (DI)

6-8.DI.1 Use the basic steps in algorithmic problem-solving to design solutions (e.g., problem statement and exploration, examination of sample instances, design, implementing a solution, testing, and evaluation).

- **Use Scratch to create programs**
    - Students use computational thinking to break down the coding project that they would like to create into smaller pieces. After determining the steps to build their project or "solve the problem" students are able to build the project.
    - Choose the correct step to fill in the step missing from {set of procedures}. Put the following steps in the correct order to successfully complete the program/solve the problem.
- **Walk a Square** (unplugged)

6-8.DI.2 Describe the process of parallelization as it relates to problem solving.

- **Use Scratch to create programs**
    - Students learn the concept of parallelization (the use of two or more processors, cores, or computers in combination to solve a single problem) in the functionality of their own games/animations. We help students to program one sprite to complete multiple commands simultaneously. We are able to demonstrate to students that by this functionality working in their own games, they are able to see how parallelism works in computers and draw larger connections to programs that they are possibly more familiar with such as social media and Google. We utilize unplugged computer science activities here as well where students are able to physically demonstrate parallelization as a class.
- **Parallelization was used in 2nd half of "Get Loopy!" Scratch project.**
- The concept of parallelism is also cross curricular and relates to math concepts learned by the students in earlier grades English/Language arts concepts (7.RL.3.1)

6-8.DI.3 Represent data in a variety of ways (e.g., text, sounds, pictures, and numbers), and use different visual representations of problems, structures, and data (e.g., graphs, charts, network diagrams, flowcharts).

- Google sheets + charts
- Google slides to create presentations
- **Use scratch to create digital representations of problems**
- **Use variables in scratch to represent data in numbers and text**

6-8.DI.4 Understand the notion of hierarchy and abstraction in computing including high level languages, translation, instruction set, and logic circuits.

- Building things in hierarchies is very common in computer software
    - hierarchy – an organizational structure in which items are ranked according to levels of importance
    - **Compare and contrast high level languages to low level languages**
- **Group Discussion: "What is Abstraction?"**
    - Abstraction is a new representation of a thing, a system, or a problem that helpfully reframes a problem by hiding details irrelevant to the question at hand.
    - Abstraction – the process of reducing complexity by focusing on the main idea. By hiding details irrelevant to the question at hand and bringing together related and useful details, abstraction reduces complexity and allows one to focus on the problem.
- **Create functions with students as an example of abstractions**
    - **Mad Libs** (unplugged)

6-8.DI.5 Demonstrate interdisciplinary applications of computational thinking and interact with content-specific models and simulations to support learning and research.

- **Use Scratch to create programs**
    - Computational thinking – the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent, for example: a computer
    - Cross-curricular projects

Computing Devices and Systems (CD)

6-8.CD.1 Demonstrate an understanding of the relationship between hardware and software.

- Software runs on hardware. Students use software to write software
- Which of these is hardware/software? Describe the interaction of hardware and software. Compare/contrast hardware and software.
- Students describe the difference between hardware and software and how they work together to provide functional technology.
- software is a collection of code installed onto your computer's hard drive.
- Discuss software vs. hardware - use Google Slide activity from SBCS

6-8.CD.2 Apply troubleshooting strategies to identify and solve routine hardware and software problems that occur during everyday computer use.

- Make how-to guides
- Help older people solve tech issues
- Plug it in, Power cycle, etc…
- This is a standard that is covered and should be modeled daily and in all usage of technology.

6-8.CD.3 Describe the major components and functions of computer systems and network.

- Students describe the concepts of systems and networks in everyday life, not just as applied to technology. Students describe the actual components and functions of computer systems and networks.
- Describe how networks function.
- What is a common non-technology network used in daily life?
- IoT connection
- How does the internet work activity (Khan Academy videos)

6-8.CD.4 Describe what distinguishes humans from machines focusing on human intelligence versus machine intelligence and ways we can communicate, as well as ways in which computers use models of intelligent behavior (e.g., robot motion, speech and language understanding, and computer vision).

- Students discuss what makes a computer intelligent.
- Computers use code to communicate.
- Create a simple chatbot in scratch - discuss the differences between chatbots and people
- Discuss and explore the difference between computer intelligence and human intelligence
- Discuss machine learning/demo a learning vision algorithm? (Google AIY Vision kit)

Programs and Algorithms (PA)

6-8.PA.1 Select appropriate tools and technology resources to support learning and personal productivity, publish individual products, and design, develop, and publish data, accomplish a variety of tasks, and solve problems.

- Tools being used by each grade level.
- How code gets online.
- The entire program. Databases. Students learn how to research what they don't know
- Need to use technology-based tools. Use "generic" programs (spreadsheet vs. excel), word-processing, email, slideshow software.
- Google Suite
- This is a standard that cuts across all classes, and is probably covered if students are using technology

6-8.PA.2 Implement problem solutions using a programming language that includes looping behavior, conditional statements, logic, expressions, variables, and functions.

- **Creating Scratch projects and learning the basic concepts of programming.**
    - Students put the lines/blocks of code into the proper order to complete a desired result.

        - Students practice exercises where they are given lines of code and understand what's going to happen.
            - **Loops- Coding Challenge**
- **Walk a Square** (unplugged)
- Cup Stacking (unplugged)
- Red Light, Green light (unplugged)

6-8.PA.3 Demonstrate dispositions amenable to open-ended problem solving and programming (e.g., comfort with complexity, persistence, brainstorming, adaptability, patience, propensity to tinker, creativity, accepting challenge).

- **Use Scratch to create programs.**
- This is something that happens daily in class, learning to problem solve

Networking and Communication (NC)

6-8.NC.1 Collaboratively design, develop, publish, and present products (e.g., videos, podcasts, websites) using technology resources that demonstrate and communicate curriculum concepts.

- **Chatbot** (plugged)
- **Project Presentations**
- Additional options:
    - Collaborate to develop a website
    - Collaborate to create videos/podcasts/slides presentations

6-8.NC.2 Exhibit dispositions necessary for collaboration: providing useful feedback, integrating feedback, understanding and accepting multiple perspectives, socialization.

- **Given specific feedback, students incorporate the feedback to improve a given sample of work.**
- **Giving feedback to other students on projects.**
- Working with different groups of people throughout the year.

Impact and Culture (IC)

6-8.IC.1 Exhibit legal and ethical behaviors when using technology and information and discuss the consequences of misuse.

- Students explain the concepts of copyright and terms of use. Given a scenario concerning technology, students determine whether it is legal and/or ethical. Students select the legal and ethical behaviors in different situations. Students identify the consequences of unethical/illegal behavior concerning technology.
- Discuss/teach cybersecurity like strong passwords
- Discuss/teach cyber attacks and tough ethical situations

- https://www.commonsense.org/education/lesson/scams-and-schemes-6-8
- **Be Internet Awesome touches on these things at a low level**

6-8.IC.2 Analyze the positive and negative impacts of technology on one's personal life, society, and our culture.

- Digital Life 101
- Describe life without technology
- Research the effects of tech on kids
- Screen time research

6-8.IC.3 Evaluate the accuracy, relevance, appropriateness, comprehensiveness, and biases that occur in electronic information sources.

- Explain how to know whether a web site/source is accurate/valid/useful for…
- Scam detector for online information
- Teach kids to be skeptical of everything
- **Be Internet Awesome touches on these things at a low level**

6-8.IC.4 Describe ethical issues that relate to computers and networks (e.g., security, privacy, ownership, and information sharing), and discuss how unequal distribution of technological resources in a global economy raises issues of equity, access, and power.

- A Creator's Rights Information Hiding Public Key Encryption Scout Patrol
- Copyright laws
- Music sharing
- Cybersecurity topics
- Cyberbullying
- Hacking
- **Be Internet Awesome touches on these things at a low level**