



**School
District of the
City of
St. Charles**

K-12 Computer Science

Approved by the Board of Education
June 12, 2025



K-12 Computer Science Curriculum Committee

Assistant Superintendent of Curriculum & Instruction
Dr. Earl Draper

Curriculum Committee Chairperson
Dr. Tina Lauer, Curriculum & Instruction Coordinator

Curriculum Developers
Dariyn Day, K-4 computer science teacher
Erica Eugea, K-4 computer science teacher
Delilah vanderGeest, K-4 computer science teacher
Abbey Guice, 5-6 computer science teacher
Stacey Stubits, 7-8 computer science teacher
Dana Buchanan, 7-8 computer science teacher
Alex Messina, High School computer science teacher
Ashley Howard, High School computer science teacher
Justin Heinen, High School computer science teacher

TABLE OF CONTENTS

Table of Contents	2
District Mission Statement	4
District Vision	4
District Values	4
District Goals	5
Philosophical Foundations	5
K-12 Computer Science Rationale	6
K-12 Computer Science Program Goals	6
K-8 Computer Science Course Descriptions	8
K-8 Computer Science Curriculum	10

District Mission

Engage. Empower. Inspire.
Every Learner, Every Day!

District Vision

The City of St. Charles School District will be a leader of academic excellence that prepares students to be successful in all aspects of life.

District Values

We, the City of St. Charles School District community of students, parents, staff, and patrons, value:

- Academic Excellence
 - We use research-based best practices to help all students reach their highest potential
 - We engage students through active learning and collaboration to foster lifelong achievement
 - We set measurable goals, personalize instruction, and celebrate every student's growth.
- Student Centered
 - We inspire learners to embrace continuous growth and improvement.
 - We develop the whole child academically, socially, emotionally, and behaviorally.
 - We equip students with the skills and knowledge for success in SCSD and beyond.
- Integrity
 - We align our actions to our values.
 - We are transparent and honest in our communication.
 - We hold ourselves accountable for doing the right thing.
- Diversity
 - We respect, honor, and celebrate every member of our community within & beyond the classroom.
 - We acknowledge and value the unique qualities in everyone's learning journey.
 - We embrace our diversity as a source of strength and unity.
- Community
 - We collaborate across all community levels to build an inclusive and supportive environment that empowers every student.
 - We cultivate opportunities for our students to connect and contribute within the district and beyond.
 - We foster open communication through diverse channels to engage all stakeholders.

District Goals

1. Academic Performance
 - Attendance
 - Student Achievement
 - Success Ready Students
2. Culture
 - Student Support
 - Sense of Belonging
 - Community Involvement
3. Facilities, Support, and Instructional Resource
 - School Configuration
 - Facilities
 - Staffing
 - Finance

K-12 Computer Science Rationale and Philosophy

Computer Science education equips students with essential skills to thrive in today's rapidly evolving digital world and competitive job market. These subjects foster critical thinking, problem-solving, and adaptability, providing students with a strong foundation in financial literacy, entrepreneurial thinking, and innovation. By developing these competencies, students can create personal brands, make informed financial decisions, and adapt to changing technological landscapes. This education prepares students to contribute meaningfully to society, collaborate effectively, and become responsible, resourceful individuals in both personal and professional contexts.

The City of St. Charles R-VI School District is dedicated to producing adept and sought-after future employees. Our comprehensive technology curriculum equips students with essential digital citizenship skills while fostering their problem-solving, critical thinking, creativity, and collaboration abilities. By cultivating these skills, we prepare students for future academic and professional success in a technology-driven world.

K-12 Computer Science Program Goals

K-4 Goals:

- Navigate the digital world safely and responsibly.
- Collaborate with peers and communicate respectfully online.
- Protect their private information in online communities.
- Break down complex problems into smaller, manageable parts and use problem-solving skills to develop innovative solutions.
- Present their ideas clearly and effectively using appropriate digital tools and applications.

5-6 Goals:

- Develop systematic routines to solve problems.
- Follow grade-level appropriate behaviors while participating in an online community.
- Use the engineering design process to plan the development of a program.
- Use algorithms to simplify code.
- Create a document or presentation with advanced formatting operations.
- Make decisions to troubleshoot problems.
- Compare tradeoffs associated with security and computing technologies.
- Make decisions about their coding options.
- Collect and display data.

7-8 Goals:

- Analyze issues and develop effective solutions.
- Foster critical thinking and creativity by designing and innovating new ideas.
- Apply technical skills through hands-on projects, utilizing tools and understanding how systems work.
- Collaborate effectively by enhancing teamwork and communication in engineering challenges.
- Develop project management skills by planning, organizing, and managing time efficiently to complete complex tasks in both daily life and future careers.
- Follow a design process that can help effectively develop a physical computing solution.
- Ask questions and define problems.
- Develop and use models.
- Use mathematics and computational thinking.
- Construct explanations and design solutions.
- Obtain, evaluate, and communicate information.
- Use iterative design.

- Apply problem-solving strategies, critical thinking, and engineering design process to overcome real-world challenges.
- Work effectively in teams, utilizing clear communication and collaborative skills to solve problems, distribute tasks, and refine designs.
- Analyze problems and break them down into logical steps applying basic programming concepts to automate tasks or develop creative solutions.
- Troubleshoot issues, refine solutions, and iterate on designs independently, demonstrating resilience and creativity when faced with unexpected challenges.
- Recognize how robotics and automation impact industries and begin to explore career pathways to personal interests in STEM fields.
- Effectively represent, encode, and visually organize data using various methods so that they can analyze, communicate, and apply information in different contexts

9-12 Goals:

- How to create apps that rely on the use of variables, data structures, operators, iteration, and abstraction.
- Understand the impact of computer science on society.
- Problem decomposition and iterative design are enhanced using scrum programming.
- Understand the importance of user-centered design and professional conduct.
- Assess security needs of a system and protect a system from possible threats.
- Collaborate as a team to analyze network traffic and implement protection measures to secure the network.
- Recognize patterns of a cybersecurity event, develop a plan for remediation, and document all evidence and procedures of remediation.
- Creatively design solutions to problems by deconstructing them into smaller, more manageable issues.
- Extrapolate real world information into a data representation, then store and access that information in a computer system.
- Provide computational instructions for a computer to complete a task with a variety of data input.
- Illustrate how the internet is a group of systems that are all interacting by using a shared procedure to transfer data.
- Present how computational solutions can be massively impactful to our societies by allowing for previously impossible solutions.
- Design and implement modular solutions using object-oriented principles.
- Analyze and evaluate processes for efficiency and clarity.
- Utilize data structures such as arrays and ArrayLists to manage and process data.
- Apply logical thinking to simplify, and solve complex problems.
- Collaborate effectively to develop, test, and debug programs.

K-12 Computer Science Course Descriptions

- **Kindergarten - Fourth Grade Computers** - The Elementary Computer Science course introduces students to essential concepts in digital citizenship, algorithms and programming, and digital publication. Students will learn how to navigate the digital world safely and responsibly. They will understand how to communicate respectfully online and protect their private information. Through engaging activities, students will develop problem-solving and critical thinking skills by learning basic programming concepts. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.
- **Fifth Grade Computers** - In fifth grade, students will expand on the skills they already learned in grades K-4. They will focus on digital citizenship, coding, and digital publication. They will understand how to communicate respectfully online and protect their private information. Students will understand the implications of the digital footprint and that it will never go away. They will also expand their coding skills and include sequencing, repetition, conditionals, and variables in their coding programs. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.
- **Sixth Grade Computers** - Students in sixth grade will expand on the skills they already learned in grades K-5. They will focus on digital citizenship, coding, and digital publication. While learning to be digital citizens, they will focus on cybersecurity and troubleshooting. They will create variables, loops, conditional statements, and control structures while creating their own coding programs. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.
- **Middle School Robotics** (grades 7 or 8) - This semester-long class is a beginning course in robotics. We will be utilizing various robotics kits and materials. The objective of this course is to introduce the student to basic programming as well as problem-solving strategies. This course will involve students in robot development, building, and programming. Students will work hands-on in teams to design, build, program, and document their progress. Topics may include motor control, gear ratios, torque, friction, sensors, program loops, decision-making, and timing sequences.
- **PLTW Design and Modeling** (grades 7 or 8) - This semester-long Project Lead the Way course is a beginning course for engineering. Students discover the design process and develop an understanding of the influence of creativity and innovation in their lives. Students learn a variety of collaboration, critical thinking, creativity and communication skills. Students also learn to use tools to create projects. They are then challenged and empowered to use and apply what they've learned throughout the unit to design a therapeutic device for a child who has a medical issue. This course allows students to explore all three PLTW pathways---Engineering, Computer Science, and Biomedical---in a fun and hands-on experience.
- **PLTW Computer Science for Innovators and Makers** (grades 7 or 8) - Computer Science for Innovators and Makers teaches students that programming goes beyond the virtual world into the physical world. Students are challenged to creatively use sensors and actuators to develop systems that interact with their environment. Designing algorithms and using computational thinking practices, they code and upload programs to microcontrollers that perform a variety of authentic tasks. The unit broadens students' understanding of computer science concepts through meaningful applications. Teams select and solve a personally relevant problem related to wearable technology, interactive art, or mechanical devices. Throughout the unit, students learn about programming for the physical world by blending hardware design and software development,

allowing students to discover computer science concepts and skills by creating personally relevant, tangible, and shareable projects.

- **PLTW Computer Cybersecurity** - Students in Cybersecurity establish an ethical code of conduct while learning to defend data in today's complex cyberworld. Cybersecurity introduces the tools and concepts of cybersecurity and encourages students to create solutions that allow people to share computing resources while protecting privacy. Nationally, computational resources are vulnerable and frequently attacked; in Cybersecurity, students solve problems by understanding and closing these vulnerabilities. This course raises students' knowledge of and commitment to ethical computing behavior. It also aims to develop students' skills as consumers, friends, citizens, and employees who can effectively contribute to communities with a dependable cyber-infrastructure that moves and processes information safely.
- **PLTW Computer Science Essentials** - In Computer Science Essentials, students will start with visual, block-based programming and smoothly transition to text-based languages like Python®. They will apply computational thinking practices, enrich their vocabulary, and engage in collaborative projects, reflecting the methodologies of computing professionals. Ultimately, they'll develop products that tackle topics and challenges significant to them. Computer Science Essentials helps students create a strong foundation to advance to Computer Science Principles, Computer Science A, and beyond.
- **PLTW Computer Science Principles (AP)** - Students explore and become inspired by career paths that utilize computing, discover tools that foster creativity and collaboration, and use what they've learned to tackle challenges. Students create apps, automate tasks, find patterns in data, and interpret simulations. Students collaborate to create and present solutions that can improve people's lives. Students will have the option to take the AP exam at the end of the course. Please contact your preferred college(s) for more information on AP exam acceptance for college credit.
- **PLTW Computer Science A (AP)** - This course provides students an introduction to college-level programming as well as the use of the Java programming language. This course covers essential topics for Java, as well as basic programming, such as variable creation and use, class creation and use, a variety of data structures and data storage. By the end, students will have a working understanding of Java and can create their own programs and apps. Students will have the option to take the AP exam at the end of the course. Please contact your preferred college(s) for more information on AP exam acceptance for college credit.

Kindergarten Computer Science <i>Course Overview</i>	
Grade level(s): Kindergarten	Credits earned: N/A (taught 30 minutes per week at elementary)
Course Description	
The Elementary Computer Science course introduces students to essential concepts in digital citizenship, algorithms and programming, and digital publication. Students will learn how to navigate the digital world safely and responsibly. They will understand how to communicate respectfully online and protect their private information. Through engaging activities, students will develop problem-solving and critical thinking skills by learning basic programming concepts. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> ● Navigate the digital world safely and responsibly. ● Collaborate with peers and communicate respectfully online. ● Protect their private information in online communities. ● Break down complex problems into smaller, manageable parts and use problem-solving skills to develop innovative solutions. ● Present their ideas clearly and effectively using appropriate digital tools and applications.	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards ● K-2 Computing Systems: ○ Hardware & Software 02 ○ Troubleshooting 01 ● K-2 Networks & the Internet: ○ Cybersecurity 01 ● K-2 Algorithms & Programming: ○ Algorithms 01 ○ Control 01 ○ Program Development 03 ● K-2 Impacts of Computing: ○ Social Interactions 01 ○ Safety, Law & Ethics 01	

Unit 1: Digital Citizenship***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards ● K-2 Networks & the Internet: ○ Cybersecurity 01 ● K-2 Impacts of Computing: ○ Social Interactions 01 ○ Safety, Law & Ethics 01	<i>Students will understand that...</i> ● Navigating the digital world safely involves recognizing and avoiding potential online risks. ● Protecting their private information online is crucial for maintaining personal security and privacy. ● Responsible digital behavior includes respecting others' viewpoints and digital property, as well as practicing empathy and kindness in online interactions.	<i>Students will consider...</i> ● What is a password? ● Why should we not share our passwords with others? ● How can I be a good digital citizen?
Learning Targets		
<i>Students will...</i> ● Access digital devices, websites, and apps with a teacher's guidance. ● Know what a password is and why they should not share theirs with others. ● Know when and why to take breaks when using a device. ● Know why it's important to be aware and respectful of people while using devices. ● Explain rules for traveling safely on the internet.		
Unit Duration:		
1st trimester		

Unit 2: Coding

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards • K-2 Algorithms & Programming: ○ Algorithms 01 ○ Control 01 ○ Program Development 03	<i>Students will understand that...</i> • Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. • Effective collaboration and respectful communication are essential for successfully creating programs. • Breaking down complex problems into smaller, manageable parts is crucial for developing innovative solutions. • Debugging algorithms using a programming language involves a systematic approach to identifying, analyzing, and resolving errors to improve program functionality.	<i>Students will consider...</i> • How can I create and follow an algorithm to complete a task? • How can I create a program with a sequence of instructions to accomplish a task? • How can I debug an algorithm?
Learning Targets		
<i>Students will...</i> • Model daily processes and follow algorithms to complete tasks. • Create and debug programs to accomplish tasks by themselves, with partners, or with a teacher's guidance.		
Unit Duration:		
2nd trimester		

Unit 3: Digital Publication

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards - K-2 Computing Systems: - Hardware & Software 02 - Troubleshooting 01	<i>Students will understand that...</i> - Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. - Different software applications serve various purposes and can help them accomplish specific tasks effectively. - Troubleshooting involves following a series of steps to identify and resolve issues, requiring logical thinking and persistence.	<i>Students will consider...</i> - How can I select the appropriate digital tool or application to complete a task? - How can I clearly present my ideas using a variety of digital tools? - What can I do when my device or software are not working like I expect them to?
Learning Targets		
<i>Students will...</i> - Explore software that can be used to accomplish a variety of tasks, such as publishing, writing, or presenting ideas visually. - Use various digital tools to publish their ideas, including text and images. - Use accurate terminology to identify simple hardware or software problems. - Follow a series of steps to resolve simple hardware or software issues.		
Unit Duration:		
3rd trimester		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
1	Digital Citizenship: Computer Lab Expectations	I can be safe, respectful, and responsible in the computer lab.	Computer Lab Expectations Slides	Observe students as they practice the expectations and recognize students demonstrating the desired behaviors.
2	Digital Citizenship: Balance Is Important	I know when and why to take breaks when using my device.	Common Sense Media: Balance Is Important	Observe students as they discuss the scenarios in the lesson with a partner. Students should demonstrate that they can find balance between their offline and online lives.
3	Digital Citizenship: Pause for People	- I know the meaning of the word device. - I know why it's important to be aware and respectful of people while using their devices.	Common Sense Media: Pause for People	Observe students as they discuss the scenarios in the lesson with a partner. Students should demonstrate that they can transition from using a device to offline activities.
4	Digital Citizenship: Managing Device Distractions	I know the meaning of the word device.	Common Sense Media: Managing Device Distractions	Observe students as they discuss ways to manage online distractions when using their devices.
5	Digital Citizenship: Safety in My Online Neighborhood	- I know the meaning of the word online. - I can explain the rules for traveling safely on the internet.	Common Sense Media: Safety in My Online Neighborhood	Students should recite the following internet safety rules included in the lesson: - Always ask your parents/teacher first. - Only talk to people you know. - Stick to places that are right for you.
6	Digital Citizenship: What Is a Password	- I know the meaning of the word password. - I know what a password is and why I should not share it with others.	NetSmartz Kids Password Rap	Provide students with the opportunity to explain what a password is and why they shouldn't share it with others, including their friends or classmates.
7	Digital Citizenship: Caring for Our Devices	I know the meaning of the word device.	Common Sense Media: Caring for Our Devices	Observe students as they model how to appropriately care for their devices.
8-10	Digital Citizenship:	I can access digital devices, websites, and apps with a teacher's guidance.	Google Classroom	Observe students as they log on to their devices, navigate to the Google Classroom app, and open the links to websites shared

				with them.
11	Coding: Learn to Drag and Drop	I can follow a step-by-step set of instructions to complete a task with guidance from my teacher.	Code.org Course A Lesson 2: Learn to Drag and Drop	Online puzzles
12	Coding: Happy Maps	- I know the meanings of the words algorithm, programming, and debugging. - I can follow a step-by-step set of instructions to complete a task with guidance from my teacher.	Code.org Course A Lesson 3: Happy Maps	Student handout (included in the lesson plan online)
13-14	Coding: Sequencing with Scrat	- I know the meanings of the following words: algorithm, program, sequencing, and debug. - I can create and debug programs to accomplish a task by myself, with a partner, or with my teacher's guidance.	Code.org Course A Lesson 4: Sequencing with Scrat	Online puzzles
15-16	Coding: Programming with Scrat	- I know the meanings of the following words: algorithm, program, programming, bug, and debug. - I can create and debug programs to accomplish a task by myself, with a partner, or with my teacher's guidance.	Code.org Course A Lesson 5: Programming with Scrat	Online puzzles
17-18	Coding: Programming with Rey and BB-8	- I know the meanings of the following words: algorithm, program, programming, bug, and debug. - I can create and debug programs to accomplish a task by myself, with a partner, or with my teacher's guidance.	Code.org Course A Lesson 6: Programming with Rey and BB-8	Online puzzles
19	Coding: Happy Loops	- I know the meaning of the word algorithm. - I can model daily processes and follow algorithms to complete tasks.	Code.org Course A Lesson 7: Happy Loops	Partner turn and talks/class discussions

20	Coding: Loops with Scrat	• I know the meaning of the word algorithm. • I can create and debug programs to accomplish a task by myself, with a partner, or with my teacher's guidance.	• Code.org Course A Lesson 8: Loops with Scrat	Online puzzles
21	Digital Publication: Hardware & Software	• I know the meanings of the words hardware and software.	• Scratch Garden: Hardware & Software video • Hardware & software picture cards	Observe students as they sort pictures of hardware and software.
22-24	Digital Publication: Keyboard Awareness	• I know the meanings of the words hardware, software, and troubleshooting. • I can recognize when a computing system is not working as expected. • I can use accurate terminology to identify simple hardware or software problems, such as the volume is turned down, the device is powered off, or the keyboard is not working.	• Keyboarding Zoo online game • Keyboarding Zoo 2 online game • Cup Stacking Typing online game	Observe students as they play online games and note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys.
25-26	Digital Publication: Word Processing Apps	• I know the meanings of the words hardware, software, and troubleshooting. • With guidance, I can use a given software to complete a task, such as publishing a digital story. • I can use accurate terminology to identify simple hardware or software problems, such as the device is powered off or the keyboard is not working.	• Google Docs	Observe students as they use a word processing app to digitally publish a provided story. Note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys. Also note students that need additional support or reteaching as they describe simple hardware and software problems using the correct terminology.
27-28	Digital Publication: Presentations	• I know the meanings of the words hardware, software, and troubleshooting. • With guidance, I can use a given software to complete a task, such as using a simple presentation tool. • I can use accurate terminology to identify simple hardware or	• Google Slides	Observe students as they create a simple presentation. Note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys. Also note students that need additional support or reteaching as they describe simple hardware and software problems using the correct terminology.

		software problems, such as the device is powered off or the keyboard is not working.		
29-30	Digital Publication: Choosing the Appropriate Software	• With guidance, I can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or using a simple presentation tool.	• Google Docs • Google Slides	Observe students as they select the appropriate app to publish a story or create a presentation.

Grade Level/Course: Computer Science- Kindergarten	Standard(s): - MO Computer Science Standards - K-2 Networks & the Internet: - Cybersecurity 01 - K-2 Impacts of Computing: - Social Interactions 01 - Safety, Law & Ethics 01 Domain: Digital Citizenship
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Observe students as they participate in turn and talk discussions with partners during the lessons from Common Sense Media. - Observe students as they model the schoolwide expectations for using a device safely, respectfully, and responsibly. - Observe students as they practice logging on to their devices using their passwords and accessing the appropriate apps and websites.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students know what passwords are and why they should not share them with others. - Students know when and why to take breaks when using their devices. - Students know why it's important to be aware and respectful of people while using their devices. - Students can explain rules for traveling safely on the internet.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. Students know the meanings of the words password, device, and online.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students can access digital devices, websites, and apps with a teacher's guidance.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science- Kindergarten	Standard(s): - MO Computer Science Standards - K-2 Algorithms & Programming: - Algorithms 01 - Control 01 - Program Development 03 Domain: Coding
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Code.org Course A online puzzle completion - Partner turn and talk discussions - Student handouts
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students can model daily processes and follow algorithms to complete tasks. - Students can create and debug programs to accomplish a task by themselves, with a partner, or with a teacher's guidance.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. Students know the meanings of the following words: algorithm, program, programming, sequencing, bug, and debug.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students can follow a step-by-step set of instructions to complete a task with guidance from a teacher.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science- Kindergarten	Standard(s): - MO Computer Science Standards - K-2 Computing Systems: - Hardware & Software 02 - Troubleshooting 01 Domain: Digital Publication
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Students will type a decodable story that has been provided to them using Google Docs. - Students will create a presentation using Google Slides that includes text and images. - Students will recognize when a computing system is not working and describe simple hardware and software problems using accurate terminology.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - With guidance, students can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or using a simple presentation tool. - Students can use accurate terminology to identify simple hardware or software problems, such as the volume is turned down, the device is powered off, or the keyboard is not working.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - With guidance, students can use a given software to complete a task, such as publishing a digital story or using a simple presentation tool. - Students can recognize when a computing system is not working as expected.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students demonstrate understanding of the meanings of the following words: software, hardware, and troubleshooting.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

First Grade Computer Science <i>Course Overview</i>	
Grade level(s): First Grade	Credits earned: N/A (taught 30 minutes per week at elementary)
Course Description	
The Elementary Computer Science course introduces students to essential concepts in digital citizenship, algorithms and programming, and digital publication. Students will learn how to navigate the digital world safely and responsibly. They will understand how to communicate respectfully online and protect their private information. Through engaging activities, students will develop problem-solving and critical thinking skills by learning basic programming concepts. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> ● Navigate the digital world safely and responsibly. ● Collaborate with peers and communicate respectfully online. ● Protect their private information in online communities. ● Break down complex problems into smaller, manageable parts and use problem-solving skills to develop innovative solutions. ● Present their ideas clearly and effectively using appropriate digital tools and applications.	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards ● K-2 Computing Systems: ○ Hardware & Software 02 ○ Troubleshooting 01 ● K-2 Networks & the Internet: ○ Cybersecurity 01 ● K-2 Algorithms & Programming: ○ Algorithms 01 ○ Control 01 ○ Program Development 03 ● K-2 Impacts of Computing: ○ Social Interactions 01 ○ Safety, Law & Ethics 01	

Unit 1: Digital Citizenship***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards ● K-2 Networks & the Internet: ○ Cybersecurity 01 ● K-2 Impacts of Computing: ○ Social Interactions 01 ○ Safety, Law & Ethics 01	<i>Students will understand that...</i> ● Navigating the digital world safely involves recognizing and avoiding potential online risks. ● Protecting their private information online is crucial for maintaining personal security and privacy. ● Responsible digital behavior includes respecting others' viewpoints and digital property, as well as practicing empathy and kindness in online interactions.	<i>Students will consider...</i> ● What is a password? ● What makes a password strong? ● Why should we not share our passwords with others? ● How can I be a good digital citizen? ● What is the difference between personal and private information? ● Why should I protect my private information when I go online?
Learning Targets		
<i>Students will...</i> ● Access digital devices, websites, and apps independently. ● Identify appropriate and inappropriate online behavior, with guidance. ● Know what makes passwords strong and why they should not share their passwords with others. ● Be safe, respectful, and responsible while working with others online. ● Know how to report concerns of cyberbullying.		
Unit Duration:		
1st trimester		

Unit 2: Coding

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards - K-2 Algorithms & Programming: - Algorithms 01 - Control 01 - Program Development 03	<i>Students will understand that...</i> - Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. - Effective collaboration and respectful communication are essential for successfully creating programs. - Breaking down complex problems into smaller, manageable parts is crucial for developing innovative solutions. - Debugging algorithms using a programming language involves a systematic approach to identifying, analyzing, and resolving errors to improve program functionality.	<i>Students will consider...</i> - How can I create and follow an algorithm to complete a task? - How can I use sequencing and repetition to create a program to complete a task? - How can I debug a program?
Learning Targets		
<i>Students will...</i> - Model daily processes and follow algorithms to complete tasks independently or with support from a teacher or other students. - Create programs that include sequencing and repetition to accomplish a task independently, collaboratively, or with a teacher's guidance. - Debug programs that include sequencing and simple loops to accomplish a task independently, collaboratively, or with a teacher's guidance.		
Unit Duration:		
2nd trimester		

Unit 3: Digital Publication***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards ● K-2 Computing Systems: ○ Hardware & Software 02 ○ Troubleshooting 01	<i>Students will understand that...</i> ● Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. ● Different software applications serve various purposes and can help them accomplish specific tasks effectively. ● Troubleshooting involves following a series of steps to identify and resolve issues, requiring logical thinking and persistence.	<i>Students will consider...</i> ● How can I select the appropriate digital tool or application to complete a task? ● How can I clearly present my ideas using a variety of digital tools? ● How can I troubleshoot simple hardware or software problems?
Learning Targets		
<i>Students will...</i> ● Explore software that can be used to accomplish a variety of tasks, such as publishing, writing, or presenting ideas visually. ● Use various digital tools to publish their ideas, including text and images. ● Use accurate terminology to identify simple hardware or software problems. ● Follow a series of steps to resolve simple hardware or software issues. ● Locate, open, modify, and save an existing file using appropriate file naming conventions.		
Unit Duration:		
3rd trimester		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
1	Digital Citizenship: Computer Lab Expectations	I can be safe, respectful, and responsible in the computer lab.	Computer Lab Expectations	Observe students as they practice the expectations and recognize students demonstrating the desired behaviors.
2	Digital Citizenship: Pause and Think Online	- I know the meanings of the words device and online. - I can be safe, respectful, and responsible online.	Common Sense Media: Pause and Think Online	Observe students as they participate in turn and talk discussions during the lesson. Students should provide responses to the given scenarios that demonstrate they can be safe, respectful, and responsible online.
3	Digital Citizenship: How Technology Makes You Feel	With my teacher's help, I can identify safe and unsafe behavior online.	Common Sense Media: How Technology Makes You Feel	Observe students as they participate in turn and talk discussions during the lesson. Students should provide responses to the given scenarios that demonstrate they can identify safe and unsafe behaviors online and respond appropriately. For example, if a video with violence begins playing, the student can close the tab, close the device, or get help from an adult. Watching violent videos can be unsafe and cause them to experience uncomfortable feelings, such as feeling scared, sad, or worried.
4	Digital Citizenship: Internet Traffic Light	I can be safe, respectful, and responsible online.	Common Sense Media: Internet Traffic Light	Observe students as they participate in turn and talk discussions during the lesson. Students should provide responses to the given scenarios that demonstrate they can be safe, respectful, and responsible online by only visiting "green" websites.
5	Digital Citizenship: Private Versus Personal Information	- I know the meanings of the terms personal information and private information. - I can stay safe online.	Mitzi and the Big Bad Nosy Wolf by Teresa Bateman	Private Versus Personal Information worksheet

6	Digital Citizenship: Private Versus Personal Information Cont.	- I know the meanings of the terms personal information and private information. - I can stay safe online.	Private Versus Personal Information Slides from Common Sense Media	Ozobot Private Versus Personal Information Activity
7	Digital Citizenship: Cyberbullying	- I know the meaning of the word cyberbully. - I know how to report cyberbullying.	Cyber.org: Don't Be Mean Behind the Screen	Observe students as they play the game included in the lesson. Students should demonstrate that they recognize examples of cyberbullying and know how to report any concerns they have to a trusted adult.
8	Digital Citizenship: Digital Footprints	- I know the meaning of the words online and digital footprint. - I can be safe, respectful, and responsible online.	Cyber.org: Online Behavior	Students will trace their footprints and draw pictures of safe online activities in their footprints. Their pictures should demonstrate that they can protect their digital footprints by visiting safe sites and not sharing private information.
9	Digital Citizenship: Passwords	- I know the meaning of the word password. - I know what makes a password strong and why I should not share my passwords with others.	Cyber.org: Passwords and Beyond	Students will demonstrate that they can create a strong password as they create a sample password following the step-by-step process provided in the lesson.
10	Digital Citizenship: Internet Safety	- I can access digital devices, apps, and websites independently. - I know how to be safe, respectful, and responsible while working with others online.	BrainPOP Jr: Internet Safety	Observe students as they access the video and lesson activities online. Students may complete the easy or hard quiz provided in the lesson to demonstrate their ability to be safe, respectful, and responsible online. .
11	Coding: Move It, Move It	- I know the meanings of the words algorithm, bug, and debugging. - I can follow a step-by-step set of instructions to complete a task with a teacher's guidance.	Code.org Course B Lesson 2: Move It, Move It	Student worksheet (included in the lesson plan online)
12-13	Coding: Sequencing with Angry Birds	I can create programs that include sequencing to accomplish a task independently, collaboratively, or with a teacher's guidance.	Code.org Course B Lesson 3: Sequencing with Angry Birds	Online puzzles

14-15	Coding: Programming with Angry Birds	• I know the meanings of the words algorithm, program, and programming. • I can create programs that include sequencing to accomplish a task independently, collaboratively, or with a teacher's guidance.	• Code.org Course B Lesson 4: Programming with Angry Birds	Online puzzles
16-17	Coding: Programming with Harvester	• I know the meanings of the following words: algorithm, program, programming, sequencing, bug, and debug. • I can create programs that include sequencing to accomplish a task independently, collaboratively, or with a teacher's guidance. • Students can debug programs that include sequencing to accomplish a task independently, collaboratively, or with a teacher's guidance.	• Code.org Course B Lesson 5: Programming with Harvester	Online puzzles
18	Coding: Getting Loopy	• I know the meaning of the word loop. • I can follow a step-by-step set of instructions to complete a task with my teacher's guidance. • With support from my teacher, I can model daily processes and follow algorithms to complete tasks.	• Code.org Course B Lesson 6: Getting Loopy	Online puzzles
19-20	Coding: Loops with Harvester	• I know the meanings of the words bug, debugging, and loop. • I can create programs that include sequencing and repetition to accomplish a task independently, collaboratively, or with a teacher's guidance. • I can debug programs that include sequencing and simple loops to accomplish a task independently, collaboratively, or with a teacher's guidance.	• Code.org Course B Lesson 7: Loops with Harvester	Online puzzles
21	Digital Publication: Hardware & Software	• I know the meanings of the words hardware and software.	• Lesson slides • Which Is It? Digital game	Exit ticket

22-23	Digital Publication: Keyboard Awareness	• I know the meanings of the words hardware, software, and troubleshooting. • I can recognize when there is a simple problem with my hardware or software that is preventing them from working as expected. • I can use accurate terminology to identify simple hardware or software problems, such as no sound coming from the device, or the caps lock key being turned on.	• Typing games from ABCYa.com	Observe students as they play online games and note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys.
24-25	Digital Publication: Word Processing Apps	• I know the meanings of the words hardware, software, and troubleshooting. • I can use accurate terminology to identify simple hardware or software problems, such as no sound coming from the device, or the caps lock key being turned on. • I can use a word processing app to publish a story.	• Google Docs	Observe students as they use a word processing app to digitally publish a provided story. Note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys. Also note students that need additional support or reteaching as they describe simple hardware and software problems using the correct terminology.
26-28	Digital Publication: Presentations	• I know the meanings of the words hardware, software, and troubleshooting. • I can use accurate terminology to identify simple hardware or software problems, such as no sound coming from the device, or the caps lock key being turned on. • I can create a simple presentation using an app.	• Google Slides	Observe students as they create a simple presentation. Note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys. Also note students that need additional support or reteaching as they describe simple hardware and software problems using the correct terminology.
29-30	Digital Publication: Choosing the Appropriate Software	• With little support, I can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or using a simple presentation tool.	• Google Docs • Google Slides	Observe students as they select the appropriate app to publish a story or create a presentation.

Proficiency Scales for Computer Science- First Grade

Grade Level/Course: Computer Science-First Grade	Standard(s): - MO Computer Science Standards - K-2 Networks & the Internet: - Cybersecurity 01 - K-2 Impacts of Computing: - Social Interactions 01 - Safety, Law & Ethics 01 Domain: Digital Citizenship
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Partner turn and talks - Lesson practice tasks - Internet safety quiz
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students know what makes passwords strong and why they should not share their passwords with others. - Students are safe, respectful, and responsible while working with others online. - Students know how to report concerns of cyberbullying.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - With guidance, students can identify appropriate and inappropriate online behavior. - Students know the meanings of the following words/phrases: password, device, online, cyberbully, personal information, private information, and digital footprint.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students can access digital devices, websites, and apps independently.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science- First Grade	Standard(s): - MO Computer Science Standards - K-2 Algorithms & Programming: - Algorithms 01 - Control 01 - Program Development 03 Domain: Coding
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Lesson worksheets - Code.org Course B online puzzles
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - With support from a teacher, students can model daily processes and follow algorithms to complete tasks. - Students can create programs that include sequencing and repetition to accomplish a task independently, collaboratively, or with a teacher's guidance. - Students can debug programs that include sequencing and simple loops to accomplish a task independently, collaboratively, or with a teacher's guidance.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. Students know the meanings of the following words: algorithm, program, programming, sequencing, loop, bug, and debug.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students can follow a step-by-step set of instructions to complete a task with a teacher's guidance.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-First Grade	Standard(s): ● MO Computer Science Standards ○ K-2 Computing Systems: ■ Hardware & Software 02 ■ Troubleshooting 01 ○ Domain: Digital Publication
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	● Exit tickets ● Students will type a decodable story that has been provided to them using a word processing app. ● Students will create a simple presentation with text and images using an app, such as Google Slides. ● Observe students as they identify when a computing system is not working and describe simple hardware and software problems using accurate terminology.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. ● With little support, students can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or using a simple presentation tool. ● Students can use accurate terminology to identify simple hardware or software problems, such as an app or program not working as expected, no sound coming from the device, or the caps lock key being turned on.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. ● With guidance, students can use a given software to complete a task, such as publishing a digital story or using a simple presentation tool. ● Students can recognize when there is a simple problem with their hardware or software that is preventing them from working as expected.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. ● Students demonstrate understanding of the meanings of the following words: software, hardware, and troubleshooting.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Second Grade Computer Science

Course Overview

Grade level(s): Second Grade	Credits earned: N/A (taught 30 minutes per week at elementary)
Course Description	
The Elementary Computer Science course introduces students to essential concepts in digital citizenship, algorithms and programming, and digital publication. Students will learn how to navigate the digital world safely and responsibly. They will understand how to communicate respectfully online and protect their private information. Through engaging activities, students will develop problem-solving and critical thinking skills by learning basic programming concepts. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> • Navigate the digital world safely and responsibly. • Collaborate with peers and communicate respectfully online. • Protect their private information in online communities. • Break down complex problems into smaller, manageable parts and use problem-solving skills to develop innovative solutions. • Present their ideas clearly and effectively using appropriate digital tools and applications.	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards • K-2 Computing Systems: ○ Hardware & Software 02 ○ Troubleshooting 01 • K-2 Networks & the Internet: ○ Cybersecurity 01 • K-2 Data & Analysis: ○ Storage 01 • K-2 Algorithms & Programming: ○ Algorithms 01 ○ Control 01 ○ Program Development 03 • K-2 Impacts of Computing: ○ Social Interactions 01 ○ Safety, Law & Ethics 01	

Unit 1: Digital Citizenship

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards ● K-2 Networks & the Internet: ○ Cybersecurity 01 ● K-2 Impacts of Computing: ○ Social Interactions 01 ○ Safety, Law & Ethics 01	<i>Students will understand that...</i> ● Navigating the digital world safely involves recognizing and avoiding potential online risks. ● Protecting their private information online is crucial for maintaining personal security and privacy. ● Responsible digital behavior includes respecting others' viewpoints and digital property, as well as practicing empathy and kindness in online interactions.	<i>Students will consider...</i> ● Why do we use passwords? ● What makes a password strong? ● How can I be a good digital citizen? ● What is cyberbullying? ● How can I report cyberbullies? ● What is safe to share online? What is unsafe to share online? ● What is a digital footprint?
Learning Targets		
<i>Students will...</i> ● Identify appropriate and inappropriate online behavior. ● Identify safe and unsafe examples of online communication. ● Explain why we use strong passwords to protect devices and information from unauthorized access. ● Develop a code of conduct and follow it while participating in an online community. ● Report inappropriate online behaviors, including cyberbullying. ● Know that the information they put online becomes part of their digital footprint.		
Unit Duration:		
1st trimester		

Unit 2: Coding

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards • K-2 Algorithms & Programming: ○ Algorithms 01 ○ Control 01 ○ Program Development 03	<i>Students will understand that...</i> • Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. • Effective collaboration and respectful communication are essential for successfully creating programs. • Breaking down complex problems into smaller, manageable parts is crucial for developing innovative solutions. • Debugging algorithms using a programming language involves a systematic approach to identifying, analyzing, and resolving errors to improve program functionality.	<i>Students will consider...</i> • How can I create and follow an algorithm to complete a task? • How can I use sequencing and looping in my program to solve a problem? • How can I debug a program that includes sequencing and looping?
Learning Targets		
<i>Students will...</i> • Create and follow algorithms to complete tasks with guidance. • Create programs that utilize sequencing and looping to solve problems with guidance. • Independently or collaboratively debug programs that include sequencing and looping with guidance.		
Unit Duration:		
2nd trimester		

Unit 3: Digital Publication

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards ● K-2 Computing Systems: ○ Hardware & Software 02 ○ Troubleshooting 01 ● K-2 Data & Analysis: ○ Storage 01	<i>Students will understand that...</i> ● Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. ● Different software applications serve various purposes and can help them accomplish specific tasks effectively. ● Troubleshooting involves following a series of steps to identify and resolve issues, requiring logical thinking and persistence.	<i>Students will consider...</i> ● How can I select the appropriate digital tool or application to complete a task? ● How can I clearly present my ideas using a variety of digital tools? ● How can I identify simple hardware or software problems? ● How can I use accurate terminology to discuss simple hardware and software problems with my classmates and teachers? ● How can I create, copy, locate, modify and delete a file in Google Drive? ● How can I use naming conventions to organize the files and folders in my Google Drive?
Learning Targets		
<i>Students will...</i> ● Independently choose appropriate software to accomplish a variety of tasks, such as publishing, writing, or presenting ideas visually. ● Use various digital tools to publish their ideas, including text and images. ● Use accurate terminology while identifying and describing simple problems hardware or software problems to peers and teachers. ● Follow a series of steps to resolve simple hardware or software issues. ● Create, copy, locate, modify, and delete an existing file. ● Use appropriate file naming conventions and recognize that files exist within an organizational structure, such as a drive, folder, and file.		
Unit Duration:		
3rd trimester		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
1	Digital Citizenship: Computer Lab Expectations	I can be safe, respectful, and responsible in the computer lab.	Computer Lab Expectations slide deck	Observe students as they practice the expectations and recognize students demonstrating the desired behaviors.
2	Digital Citizenship: We, the Digital Citizens	I can identify appropriate and inappropriate online behavior.	Common Sense Media: We, the Digital Citizens	Observe students as they participate in turn and talk discussions. Students should demonstrate that they can recognize appropriate and inappropriate behaviors online.
3	Digital Citizenship: That's Private!	- I know the meanings of the terms personal information and private information. - I can identify safe and unsafe examples of online communication.	Common Sense Media: That's Private!	Student handout (included in the lesson plan online)
4	Digital Citizenship: Digital Trails	- I know the meaning of the terms private information, personal information, and digital footprint. - I know that the information I put online becomes part of my digital footprint.	Common Sense Media: Digital Trails	Student handout (included in the lesson plan online)
5	Digital Citizenship: Who Is In Your Online Community?	- I know the meaning of the term online community. - I can respectfully participate in an online community. - I can identify safe and unsafe examples of online communication.	Common Sense Media: Who Is In Your Online Community?	Student handout (included in the lesson plan online)
6	Digital Citizenship: Putting a STOP to Online Meanness	- I know the meaning of the word cyberbullying - I can identify appropriate and inappropriate online behavior. - I can report inappropriate online behaviors, including cyberbullying.	Common Sense Media: Putting a STOP to Online Meanness	Student handout (included in the lesson plan online)
7-8	Digital Citizenship: Password Power-Up	I know the meanings of the words password and device.	Common Sense Media: Password Power-Up	Student handout (included in the lesson plan online)

		• can explain why we use strong passwords to protect devices and information from unauthorized access.		
9	Digital Citizenship: Our Device Charter	• I know the meaning of the term online community. • I can work with my class to develop a code of conduct and follow it while participating in an online community.	• Common Sense Media: Our Device Charter	Student handout (included in the lesson plan online)
10	Digital Citizenship: Join Our Online Community	• I can respectfully participate in an online community. • I can follow my class's code of conduct while participating in an online community.	• Google Classroom	Observe students' interactions in a discussion board post in Google Classroom.
11	Coding: My Robotic Friends Jr.	• I can model daily processes by creating and following a set of step-by-step instructions in an unplugged activity, with guidance. • I know the meanings of the following words: algorithm, bug, debug, and program.	• Code.org Course C Lesson 2: My Robotic Friends Jr.	Cup stacking partner activity observation
12-13	Coding: Programming with Angry Birds	• I know the meanings of the following words: algorithm, bug, debugging, and sequencing. • With guidance, I can create and follow algorithms to complete tasks.	• Code.org Course C Lesson 3: Programming with Angry Birds	Online puzzles
14-15	Coding: Debugging in Maze	• I know the meanings of the words bug and debugging. • With guidance, I can independently or collaboratively debug programs that include sequencing.	• Code.org Course C Lesson 4: Debugging in Maze	Online puzzles
16	Coding: Collecting Treasure with Laurel	• I know the meanings of the following words: algorithm, program, and programming. • With guidance, I can create programs that utilize sequencing to solve problems.	• Code.org Course C Lesson 5: Collecting Treasure with Laurel	Online puzzles

17	Coding: My Loopy Robotic Friends Jr.	- I can model daily processes by creating and following a set of step-by-step instructions in an unplugged activity, with guidance. - I know the meaning of the word loop.	Code.org Course C Lesson 7: My Loopy Robotic Friends Jr.	Observe students as they complete the unplugged cup stacking partner activity.
18-19	Coding: Loops with Rey and BB-8	- I know the meaning of the word loop. - With guidance, I can create programs that utilize sequencing and looping to solve problems.	Code.org Course C Lesson 8: Loops with Rey and BB-8	Online puzzles
20	Coding: Harvesting Crops with Loops	- I know the meaning of the words loop, bug, and debugging. - With guidance, I can create programs that utilize sequencing and looping to solve problems. - With guidance, students can independently or collaboratively debug programs that include sequencing and looping.	Code.org Course C Lesson 9: Harvesting Crops with Loops	Online puzzles
21	Digital Publication: Hardware & Software	I know the meanings of the words hardware and software.	BrainPOP Jr.: Parts of a Computer	Observe students as they sort pictures of hardware and software during a class relay race.
22	Digital Publication: Keyboard Awareness	I know the meanings of the words hardware and software.	Typing.com	Observe students as they complete the Learn the Keys lessons on Typing.com.
23	Digital Publication: Navigating Google Drive	I know the meanings of the following words: data, file, and folder.	Lesson slides	View students' Google Drive and note students that need support in naming and organizing their files and folders.
24-25	Digital Publication: Word Processing Apps	- I know the meanings of the following words: hardware, software, troubleshooting, file, and folder. - With guidance, I can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or creating a simple presentation. - I can use accurate terminology while identifying and discussing simple hardware or software problems, such as an app or program not working as	Google Docs	Observe students as they use a word processing app to digitally publish a provided story. Note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys. Also note students that need additional support or reteaching as they describe simple hardware and software problems using the correct terminology.

		expected, no sound coming from the device, or the caps lock key being turned on, with my peers and teachers.		
26-28	Digital Publication: Presentations	• I know the meanings of the following words: hardware, software, troubleshooting, file, and folder. • With guidance, I can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or creating a simple presentation. • I can use accurate terminology while identifying and discussing simple hardware or software problems, such as an app or program not working as expected, no sound coming from the device, or the caps lock key being turned on, with my peers and teachers.	• Google Slides	Observe students as they create a simple presentation. Note students that need help finding the letter keys, number keys, shift keys, spacebar, and punctuation keys. Also note students that need additional support or reteaching as they describe simple hardware and software problems using the correct terminology.
29-30	Digital Publication: Choosing the Appropriate Software	• I can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or creating a simple presentation.	• Google Docs • Google Slides • Book Creator	Observe students as they select the appropriate app to publish a story or create a presentation.

Proficiency Scales for Computers Second Grade

Grade Level/Course: Computer Science-Second Grade	Standard(s): ● MO Computer Science Standards ○ K-2 Networks & the Internet: ■ Cybersecurity 01 ○ K-2 Impacts of Computing: ■ Social Interactions 01 ■ Safety, Law & Ethics 01 Domain: Digital Citizenship
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	● Partner turn and talks ● Lesson handouts ● Discussion board participation
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. ● Students can explain why we use strong passwords to protect devices and information from unauthorized access. ● Students will develop a code of conduct and follow it while participating in an online community. ● Students can report inappropriate online behaviors, including cyberbullying. ● Students know that the information they put online becomes part of their digital footprint.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. ● Students can identify appropriate and inappropriate online behavior. ● Students can identify safe and unsafe examples of online communication. ● Students know the meanings of the following words/phrases: password, device, online community, cyberbully, personal information, private information, and digital footprint.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. ● Students can respectfully participate in an online community.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Second Grade	Standard(s): - MO Computer Science Standards - K-2 Algorithms & Programming: - Algorithms 01 - Control 01 - Program Development 03 Domain: Coding
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Unplugged cup stacking activity - Code.org Course C online puzzles
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - With guidance, students can create and follow algorithms to complete tasks. - With guidance, students can create programs that utilize sequencing and looping to solve problems. - With guidance, students can independently or collaboratively debug programs that include sequencing and looping.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. Students demonstrate understanding of the meanings of the following words: algorithm, bug, debug, program, programming, sequencing, and loop.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students can model daily processes by creating and following a set of step-by-step instructions in an unplugged activity, with guidance.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Second Grade	Standard(s): - MO Computer Science Standards - K-2 Computing Systems: - Hardware & Software 02 - Troubleshooting 01 - K-2 Data & Analysis: - Storage 01 Domain: Digital Publication
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Hardware & Software relay race - Google Drive organization checklist - Students will choose the appropriate software to publish a story. - Students will choose the appropriate software to create a simple presentation. - Students will recognize basic hardware and software issues and apply the correct terminology to effectively discuss and troubleshoot them.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or creating a simple presentation. - Students can use accurate terminology while identifying and discussing simple hardware or software problems, such as an app or program not working as expected, no sound coming from the device, or the caps lock key being turned on, with their peers and teachers.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - With guidance, students can choose the appropriate software to perform a variety of tasks, such as publishing a digital story or creating a simple presentation. - Students can recognize when there is a simple problem with their hardware or software that is preventing them from working as expected.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students demonstrate understanding of the meanings of the following words: software, hardware, data, file, folder, and troubleshooting.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Third Grade Computer Science Course Overview	
Grade level(s): Third Grade	Credits earned: N/A (taught 30 minutes per week at elementary)
Course Description	
The Elementary Computer Science course introduces students to essential concepts in digital citizenship, algorithms and programming, and digital publication. Students will learn how to navigate the digital world safely and responsibly. They will understand how to communicate respectfully online and protect their private information. Through engaging activities, students will develop problem-solving and critical thinking skills by learning basic programming concepts. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> • Navigate the digital world safely and responsibly. • Collaborate with peers and communicate respectfully online. • Protect their private information in online communities. • Break down complex problems into smaller, manageable parts and use problem-solving skills to develop innovative solutions. • Present their ideas clearly and effectively using appropriate digital tools and applications.	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards • 3-5 Computing Systems: ◦ Troubleshooting 01 • 3-5 Networks & the Internet: ◦ Cybersecurity 01 • 3-5 Algorithms & Programming: ◦ Algorithms 01 ◦ Control 01 ◦ Modularity 01 ◦ Program Development 03 • 3-5 Impacts of Computing: ◦ Social Interactions 01 ◦ Safety, Law & Ethics 01 and 02	International Society of Technology in Education Standards for Students (ISTE) • Creative Communicator ◦ 1.6a: Choose Platforms or Tools

Unit 1: Digital Citizenship***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 3-5 Networks & the Internet: - Cybersecurity 01 3-5 Impacts of Computing: - Social Interactions 01 - Safety, Law & Ethics 01 and 02	<i>Students will understand that...</i> - Navigating the digital world safely involves recognizing and avoiding potential online risks. - Protecting their private information online is crucial for maintaining personal security and privacy. - Responsible digital behavior includes respecting others' viewpoints and digital property, as well as practicing empathy and kindness in online interactions.	<i>Students will consider...</i> - How can I safely and respectfully participate in an online community? - How can using a device or network inappropriately cause problems? - How can I identify and report inappropriate online behaviors? - How can I respect the intellectual property rights of others?
Learning Targets		
<i>Students will...</i> - Identify some of the problems that may occur when someone uses a device or network inappropriately. - Develop a code of conduct and follow it while participating in an online community. - Identify and report inappropriate online behavior, including cyberbullying. - Identify types of digital data that may have intellectual property rights that prevent copying or require attribution.		
Unit Duration:		
1st trimester		

Unit 2: Coding

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standard 3-5 Algorithms & Programming: - Algorithms 01 - Control 01 - Modularity 01 - Program Development 03	<i>Students will understand that...</i> - Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. - Effective collaboration and respectful communication are essential for successfully creating programs. - Breaking down complex problems into smaller, manageable parts is crucial for developing innovative solutions. - Debugging algorithms using a programming language involves a systematic approach to identifying, analyzing, and resolving errors to improve program functionality.	<i>Students will consider...</i> - Which algorithm is most efficient in accomplishing a task? - How can I use control structures to make decisions within a program? - How can I decompose, or break down, a problem into the steps needed to solve it? - How can I find and fix bugs in a program?
Learning Targets		
<i>Students will...</i> - Compare multiple algorithms for accomplishing the same task. - Collaboratively create a program using control structures, such as sequence, conditionals, and interactive looping, to make decisions within their program. - Decompose a problem into the precise sequence of instructions needed to solve a problem. - Analyze and debug a program that includes sequencing, repetition, and variables in a programming language.		
Unit Duration:		
2nd Trimester		

Unit 3: Digital Publication

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 3-5 Computing Systems: - Troubleshooting 01 International Society of Technology in Education Standards for Students (ISTE) - Creative Communicator 1.6a: Choose Platforms or Tools	<i>Students will understand that...</i> - Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. - Different software applications serve various purposes and can help them accomplish specific tasks effectively. - Troubleshooting involves following a series of steps to identify and resolve issues, requiring logical thinking and persistence. - Different types of information are stored in different formats and have varied storage requirements.	<i>Students will consider...</i> - How can I select the appropriate application to complete a task? - How can I clearly present my ideas while digitally publishing a story, graphic, or audio/video recording? - How can I troubleshoot simple hardware or software problems? - Which programs can be used to open different file types, such as documents or PDFs?
Learning Targets		
<i>Students will...</i> - Recognize that different types of information are saved in different formats and can be opened by their associated programs. - Choose appropriate apps to digitally publish their work. - Demonstrate an understanding of basic word processing skills including highlighting, cutting, copying, and pasting text within a document. - Select fonts, select font size, select font color, insert and resize images, and utilize built-in spell check and grammar check tools while using a word processing document. - Create presentations that demonstrate their ability to insert text boxes, insert and resize images, and use drawing features.		
Unit Duration:		
3rd trimester		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
1	Digital Citizenship: Computer Lab Expectations	I can be safe, respectful, and responsible in the computer lab.	Computer Lab Expectations slide deck	Observe students as they practice the expectations and recognize students demonstrating the desired behaviors.
2	Digital Citizenship: Your Rings of Responsibility	- I know the meaning of the term digital citizen. - I can identify and report inappropriate online behavior, including cyberbullying.	Common Sense Media: Your Rings of Responsibility	Student handout (included in the lesson plan online)
3	Digital Citizenship: Our Digital Citizenship Pledge	- I know the meanings of the terms digital citizen and online community. - I can respectfully participate in an online community. - I can work with my class to develop a code of conduct and follow it while participating in an online community.	Common Sense Media: Our Digital Citizenship Pledge	Student handout (included in the lesson plan online)
4	Digital Citizenship: The Power of Words	- I know the meaning of the word cyberbully. - I can identify and report inappropriate online behavior, including cyberbullying.	Common Sense Media: The Power of Words	Student handout (included in the lesson plan online)
5	Digital Citizenship: From Bystanders to Upstanders	- I know the meaning of the word cyberbully. - I can identify and report inappropriate online behavior, including cyberbullying.	Be Internet Awesome: It's Cool to Be Kind Activity 1-From Bystanders to Upstanders	- Student handout (included in the lesson plan online) - Scenario discussions
6-7	Digital Citizenship: Sharing Fairly	I know the meanings of the terms intellectual property rights and credit.	Copyright & Creativity: Sharing Fairly	Observe students during their turn and talk partner discussions.
8	Digital Citizenship: Access Denied	- I know the meanings of the words device, network, and data. - I can identify some of the problems that may occur when someone uses a device or network inappropriately.	Cyber.org: Access Denied	Observe students during their small group discussions.
9-10	Digital Citizenship: Threats with Online Gaming	- I know the meanings of the words device, network, and data. - I can identify some of the problems that may occur when someone uses a device or network inappropriately.	Cyber.org: Threats with Online Gaming	Poster project

11	Coding: Graph Paper Programming	• With guidance, I can create a program using sequencing to complete a task. • I know the meaning of the word algorithm.	• Code.org Course D Lesson 2: Graph Paper Programming	Unplugged graph paper programming activity
12-13	Coding: Introduction to Online Puzzles	• I know the meanings of the following words: program, programming, loop, bug, and debugging. • I can decompose a problem into the precise sequence of instructions needed to solve a problem. • I can analyze and debug a program that includes sequencing and repetition.	• Code.org Course D Lesson 3: Introduction to Online Puzzles	Online puzzles
14-15	Coding: Debugging with Laurel	• I know the meanings of the words bug, debugging, and decompose. • Students can analyze and debug a program that includes sequencing and repetition.	• Code.org Course D Lesson 4: Debugging with Laurel	Online puzzles
16-17	Coding: Dance Party	• I know the meanings of the words program and variable. • I can collaboratively create a program using control structures, such as sequence and conditionals.	• Code.org Course D Lesson 8: Dance Party	Dance party coding project
18	Coding: Loops in Ice Age	• I know the meanings of the words algorithm, decompose, sequence, loop and repetition. • I can compare multiple algorithms for accomplishing the same task. • I can decompose a problem into the precise sequence of instructions needed to solve a problem.	• Code.org Course D Lesson 10: Loops in Ice Age	Online puzzles
19-20	Coding: Looking Ahead with Minecraft	• I know the meanings of the following words: algorithm, program, programming, and conditional. • I can collaboratively create a program using control structures, such as sequence, conditionals, and interactive looping, to make decisions within our program.	• Code.org Course D Lesson 14: Looking Ahead with Minecraft	Online puzzles
21	Digital Publication: Hardware & Software	• I know the meanings of the words software and hardware.	• Typing.com • Blooket	Online game, such as Booklet

22	Digital Publication: Keyboarding	I know the meanings of the words software and hardware.	Typing.com	Observe students as they begin working on the Learn the Keys lessons.
23	Digital Publication: Troubleshooting	- I know the meanings of the following words: software, hardware, and troubleshooting. - I can use accurate terminology while identifying and discussing simple hardware or software problems with their peers and teachers. - I can apply troubleshooting strategies, such as refreshing their screen, closing and reopening a file or application, and unmuting or adjusting the volume on headphones.	Google Docs Google Slides	Observe students as they discuss troubleshooting scenarios in both Docs and Slides. Students should describe basic problems with their hardware or software and suggest possible simple solutions to resolve them.
24-25	Digital Publication: Google Docs	- I know the meanings of the following words: font, text box, software, hardware, and troubleshooting. - I can use basic word processing skills including highlighting, cutting, copying, and pasting text within a document. - I can select fonts, select font size, select font color, insert and resize images, and utilize built-in spell check and grammar check tools while using a word processing app.	Google Docs	Observe students as they publish a story using a word processor, such as Google Docs.
26-28	Digital Publication: Google Slides	- I know the meanings of the following words: font, text box, software, hardware, and troubleshooting. - I can create presentations that demonstrate their ability to insert text boxes, insert and resize images, and use drawing features.	Google Slides	Observe students as they create a simple presentation using an appropriate app, such as Google Slides.
29-30	Digital Publication: Choosing the Appropriate Software	- I can choose the appropriate apps to perform a variety of tasks, such as publishing a digital story, creating an audio or video recording, and preparing a digital presentation. - I recognize that different types of information are saved in different formats and can be opened by their associated programs.	Google Docs Google Slides Book Creator	Monitor students as they select the appropriate app for the assigned task. For example, they might select Google Docs to create a journal entry.

Proficiency Scales for Computers Third Grade

Grade Level/Course: Computer Science-Third Grade	Standard(s): ● MO Computer Science Standards ○ 3-5 Networks & the Internet: ■ Cybersecurity 01 ○ 3-5 Impacts of Computing: ■ Social Interactions 01 ■ Safety, Law & Ethics 01 and 02 Domain: Digital Citizenship
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	● Partner turn and talks ● Lesson handouts ● Poster project
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. ● Students can identify some of the problems that may occur when someone uses a device or network inappropriately. ● Students will develop a code of conduct and follow it while participating in an online community. ● Students can identify and report inappropriate online behavior, including cyberbullying. ● Students can identify types of digital data that may have intellectual property rights that prevent copying or require attribution.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. ● Students demonstrate understanding of the meanings of the following words/phrases: device, network, online community, digital citizen, report, cyberbully, data, intellectual property rights, credit, and digital footprint.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. ● Students can respectfully participate in an online community.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Third Grade	Standard(s): - MO Computer Science Standards 3-5 Algorithms & Programming: - Algorithms 01 - Control 01 - Modularity 01 - Program Development 03 Domain: Coding
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Unplugged graph paper programming activity - Code.org Course D online puzzles
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students can compare multiple algorithms for accomplishing the same task. - Students can collaboratively create a program using control structures, such as sequence, conditionals, and interactive looping, to make decisions within their program. - Students can decompose a problem into the precise sequence of instructions needed to solve a problem. - Students can analyze and debug a program that includes sequencing, repetition, and variables in a programming language.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. Students demonstrate understanding of the meanings of the following words: algorithm, program, programming, sequence, conditional, looping, decompose, repetition, variable, bug, and debug.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. With guidance, students can create a program using sequencing to complete a task.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Third Grade	Standard(s): - MO Computer Science Standards 3-5 Computing Systems: - Troubleshooting 01 - International Society of Technology in Education Standards for Students (ISTE) - Creative Communicator 1.6a Domain: Digital Publication
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Students will choose the appropriate software to publish a digital story while demonstrating an understanding of basic word processing skills. - Students will choose the appropriate software to create and/or edit a presentation. - Students will open different file types from their Google Drive. - Students will recognize basic hardware and software issues and use basic troubleshooting strategies to resolve them.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students demonstrate an understanding of basic word processing skills including highlighting, cutting, copying, and pasting text within a document. - Students select fonts, select font size, select font color, insert and resize images, and utilize built-in spell check and grammar check tools while using a word processing app. - Students create presentations that demonstrate their ability to insert text boxes, insert and resize images, and use drawing features. - Students recognize that different types of information are saved in different formats and can be opened by their associated programs. - Students use accurate terminology while identifying and discussing simple hardware or software problems with their peers and teachers. - Students apply troubleshooting strategies, such as refreshing their screen, closing and reopening a file or application, and unmuting or adjusting the volume on headphones.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Students can choose the appropriate apps to perform a variety of tasks, such as publishing a digital story, creating an audio or video recording, and preparing a digital presentation. - Students can recognize when there is a simple problem with their hardware or software that is preventing them from working as expected and explain the issue to a classmate or teacher.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students demonstrate understanding of the meanings of the following words: font, text box, software, hardware, and troubleshooting.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Fourth Grade Computer Science

Course Overview

Grade level(s): Fourth Grade	Credits earned: N/A (taught 30 minutes per week at elementary)
Course Description	
The Elementary Computer Science course introduces students to essential concepts in digital citizenship, algorithms and programming, and digital publication. Students will learn how to navigate the digital world safely and responsibly. They will understand how to communicate respectfully online and protect their private information. Through engaging activities, students will develop problem-solving and critical thinking skills by learning basic programming concepts. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> • Navigate the digital world safely and responsibly. • Collaborate with peers and communicate respectfully online. • Protect their private information in online communities. • Break down complex problems into smaller, manageable parts and use problem-solving skills to develop innovative solutions. • Present their ideas clearly and effectively using appropriate digital tools and applications.	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards • 3-5 Computing Systems: ◦ Troubleshooting 01 • 3-5 Networks & the Internet: ◦ Cybersecurity 01 • 3-5 Algorithms & Programming: ◦ Algorithms 01 ◦ Control 01 ◦ Modularity 01 ◦ Program Development 03 • 3-5 Impacts of Computing: ◦ Social Interactions 01 ◦ Safety, Law & Ethics 01 and 02	International Society of Technology in Education Standards for Students (ISTE) • Creative Communicator ◦ 1.6a: Choose Platforms or Tools

Unit 1: Digital Citizenship

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 3-5 Networks & the Internet: - Cybersecurity 01 3-5 Impacts of Computing: - Social Interactions 01 - Safety, Law & Ethics 01 and 02	<i>Students will understand that...</i> - Navigating the digital world safely involves recognizing and avoiding potential online risks. - Protecting their private information online is crucial for maintaining personal security and privacy. - Responsible digital behavior includes respecting others' viewpoints and digital property, as well as practicing empathy and kindness in online interactions.	<i>Students will consider...</i> - What are some of the real-world cybersecurity issues that I may face? - How can I safely, respectfully, and responsibly participate in an online community? - How can I protect my personal and private information? - How can I identify and report inappropriate online behaviors? - How can violating someone's intellectual property rights impact me socially? - How can a negative digital footprint impact me?
Learning Targets		
<i>Students will...</i> - Discuss real-world cybersecurity issues. - Develop a code of conduct and follow it while participating in online communities. - Identify strategies that can be used to protect their personal and private information. - Identify and report inappropriate online behavior, including cyberbullying. - Understand the potential social impact of violating intellectual property rights. - Understand the implications of a negative digital footprint.		
Unit Duration:		
1st trimester		

Unit 2: Coding

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standard 3-5 Algorithms & Programming: - Algorithms 01 - Control 01 - Modularity 01 - Program Development 03	<i>Students will understand that...</i> - Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. - Effective collaboration and respectful communication are essential for successfully creating programs. - Breaking down complex problems into smaller, manageable parts is crucial for developing innovative solutions. - Debugging algorithms using a programming language involves a systematic approach to identifying, analyzing, and resolving errors to improve program functionality.	<i>Students will consider...</i> - How can I compare and simplify multiple algorithms for accomplishing the same task? - How can I create a program using control structures to solve a problem? - How can I decompose large problems into smaller, manageable sub problems during the program development process? - How can I analyze, create, and debug a program?
Learning Targets		
<i>Students will...</i> - Students can compare and simplify multiple algorithms for accomplishing the same task. - Students can create a program using control structures, such as sequences, conditionals, or interactive looping, to solve a problem. - Students can decompose large problems into smaller, manageable sub problems during the program development process. - Students can analyze, create, and debug a program that includes sequencing, repetition, conditionals, and variables in a programming language.		
Unit Duration:		
2nd trimester		

Unit 3: Digital Publication

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 3-5 Computing Systems: - Troubleshooting 01 International Society of Technology in Education Standards for Students (ISTE) - Creative Communicator 1.6a: Choose Platforms or Tools	<i>Students will understand that...</i> - Navigating the digital world safely involves using trusted websites and apps while maintaining their privacy. - Different software applications serve various purposes and can help them accomplish specific tasks effectively. - Troubleshooting involves following a series of steps to identify and resolve issues, requiring logical thinking and persistence. - Files can be stored in different locations based on the file type, storage requirements, and sharing requirements.	<i>Students will consider...</i> - How can I select the appropriate application to complete a task? - How can I clearly present my ideas while digitally publishing a story, graphic, or audio/video recording? - How can I troubleshoot simple hardware or software problems? - Which storage location is appropriate based on the type of file I am saving?
Learning Targets		
<i>Students will...</i> - Choose different storage locations for files based on the type of file, storage requirements, and sharing requirements. - Choose appropriate apps to digitally publish their work. - Demonstrate an understanding of basic word processing skills including highlighting, cutting, copying, and pasting text within a document. - Format and align text, insert and resize images, and utilize built-in spell check and grammar check tools while using a word processing document. - Create presentations that demonstrate their ability to choose a theme, insert text boxes, create bulleted and numbered lists, insert and resize images, and use drawing features.		
Unit Duration:		
3rd trimester		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
1	Digital Citizenship: Computer Lab Expectations	I can be safe, respectful, and responsible in the computer lab.	Computer Lab Expectations slide deck	Observe students as they practice the expectations and recognize students demonstrating the desired behaviors.
2	Digital Citizenship: Private and Personal Information	- I know the meanings of the terms private information and personal information. - I can discuss real-world cybersecurity problems. - I can identify strategies that can be used to protect my personal and private information.	Common Sense Media: Private and Personal Information	Exit ticket (included in lesson plan online)
3	Digital Citizenship: Our Online Tracks	- I know the meaning of the term digital footprint. - I understand the implications of a negative digital footprint.	Common Sense Media: Our Online Tracks	Handout (included in lesson plan online)
4-5	Digital Citizenship: Keeping Games Fun and Friendly	- I can respectfully participate in online communities. - I know the meaning of the term online community. - I can develop a code of conduct and follow it while participating in online communities.	Common Sense Media: Keeping Games Fun and Friendly	Video game cover project
6-7	Digital Citizenship: Be a Super Digital Citizen	- I can respectfully participate in online communities. - I know the meaning of the word cyberbullying. - I can identify and report inappropriate online behavior, including cyberbullying.	Common Sense Media: Be a Super Digital Citizen	Comic strip project
8-9	Digital Citizenship: A Creator's Rights and Responsibilities	- I know the meanings of the terms intellectual property rights and citing online sources. - I understand the potential social impact of violating intellectual property rights.	Common Sense Media: A Creator's Rights and Responsibilities	Handout (included in lesson plan online)
10	Digital Citizenship: Don't Fall for Fake	- I know the meanings of the terms private information, spam, and cybersecurity. - I can discuss real-world cybersecurity problems.	Be Internet Awesome: Don't Fall for Fake Activity 1	Handout (included in lesson plan online)

11-12	Coding: Swimming Fish with Sprite Lab	- I can create a program using a programming language to accomplish a simple task, such as making a new sprite that displays a speech bubble when the program is run. - I know the meanings of the words program and decompose.	Code.org Course E Lesson 3: Swimming Fish with Sprite Lab	Online puzzles/free play project
13-14	Coding: Hello World	- I can create a program using a programming language to accomplish a simple task, such as making a new sprite that displays a speech bubble when the program is run. - I know the meanings of the words algorithm and loop. - I can decompose large problems into smaller, manageable sub problems during the program development process.	Code.org Course E Lesson 4: Hello World	Online puzzles/free play project
15-16	Coding: Game Design	- I know the meanings of the word sequence and event. - I can create a program using control structures, such as sequences, conditionals, or interactive looping, to solve a problem.	Code.org Course E Lesson 6: Game Design	Online puzzles
17-18	Coding: Variables as Score	- I know the meanings of the words variable, conditional and debug. - I can analyze, create, and debug a program that includes sequencing, repetition, conditionals, and variables in a programming language.	Code.org Course E Lesson 7: Variables as Score	Online puzzles and free play coding challenge
19-20	Coding: Environment and Players	- I know the meanings of the words variable, conditional, simplify, and debug. - I can compare and simplify multiple algorithms for accomplishing the same task. - I can decompose large problems into smaller, manageable sub problems during the program development process.	Code.org Course E Lesson 8: Environment and Players	Online puzzles
21	Digital Publication: Hardware & Software	I know the meanings of the words, software and hardware.	Typing.com	Online hardware and software activities
22	Digital Publication: Keyboarding	I know the meanings of the words, software and hardware.	Typing.com	Observe students as they begin working on the Learn the Keys lessons.

23	Digital Publication: Troubleshooting	• I know the meanings of the following words: software, hardware, and troubleshooting. • I can use accurate terminology while identifying and discussing simple hardware or software problems with my peers and teachers. • I can apply troubleshooting strategies, such as rebooting the device, checking the power, and forcing a shutdown of an application.	• Google Docs • Google Slides	Observe students as they discuss troubleshooting scenarios in both Docs and Slides. Students should describe basic problems with their hardware or software and suggest possible simple solutions to resolve them.
24-25	Digital Publication: Google Docs	• I know the meanings of the following words: font, format, align, and troubleshooting. • I can use basic word processing skills including highlighting, cutting, copying, and pasting text within a document. • I can format and align text, insert and resize images, and utilize built-in spell check and grammar check tools while using a word processing app.	• Google Docs	Observe students as they publish a story using a word processor, such as Google Docs.
26-28	Digital Publication: Google Slides	• I know the meanings of the following words: font, text box, align, and troubleshooting. • I can create presentations that demonstrate my ability to choose a theme, insert text boxes, create bulleted and numbered lists, insert and resize images, and use drawing features.	• Google Slides	Observe students as they create a simple presentation using an appropriate app, such as Google Slides.
29-30	Digital Publication: Choosing the Appropriate Software	• I can choose the appropriate apps to perform a variety of tasks, such as publishing a digital story, creating an audio or video recording, and preparing a digital presentation. • I can choose different storage locations (e.g., physical, shared, cloud) for files based on the type of file, storage requirements, and sharing requirements.	• Google Docs • Google Slides • Book Creator	Monitor students as they select the appropriate app for the assigned task. For example, they might select Google Docs to create a journal entry.

Proficiency Scales for Computers Fourth Grade

Grade Level/Course: Computer Science-Fourth Grade	Standard(s): • MO Computer Science Standards ○ 3-5 Networks & the Internet: ■ Cybersecurity 01 ○ 3-5 Impacts of Computing: ■ Social Interactions 01 ■ Safety, Law & Ethics 01 and 02 Domain: Digital Citizenship
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	• Lesson handouts • Exit tickets • Partner/small group discussion observations • Video game cover project • Comic strip project
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. • Students will develop a code of conduct and follow it while participating in online communities. • Students can identify strategies that can be used to protect their personal and private information. • Students can identify and report inappropriate online behavior, including cyberbullying. • Students understand the potential social impact of violating intellectual property rights. • Students understand the implications of a negative digital footprint.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. • Students can discuss real-world cybersecurity problems. • Students demonstrate understanding of the meanings of the following words/phrases: cybersecurity, personal information, private information, online community, spam, cyberbullying, intellectual property rights, citing online sources, and digital footprint.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. • Students can respectfully participate in online communities.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Fourth Grade	Standard(s): - MO Computer Science Standards 3-5 Algorithms & Programming: - Algorithms 01 - Control 01 - Modularity 01 - Program Development 03 Domain: Coding
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Online puzzles - Free play coding challenges
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students can compare and simplify multiple algorithms for accomplishing the same task. - Students can create a program using control structures, such as sequences, conditionals, or interactive looping, to solve a problem. - Students can decompose large problems into smaller, manageable sub problems during the program development process. - Students can analyze, create, and debug a program that includes sequencing, repetition, conditionals, and variables in a programming language.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. Students know the meanings of the following words: algorithm, event, simplify, program, sequence, conditional, looping, decompose, debug, and variable.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students can create a program using a programming language to accomplish a simple task, such as making a new sprite that displays a speech bubble when the program is run.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Fourth Grade	Standard(s): - MO Computer Science Standards 3-5 Computing Systems: T 01 - International Society of Technology in Education Standards for Students (ISTE) - Creative Communicator 1.6a Domain: Digital Publication
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Students will choose the appropriate software to publish a digital story while demonstrating an understanding of basic word processing skills. - Students will use the spell check and grammar check tools in a word processing app to edit a document. - Students will recognize basic hardware and software issues and use basic troubleshooting strategies to resolve them. - Students will choose the appropriate storage location for a file based on given requirements (scenarios). - Students will choose the appropriate software to create a presentation that is visually appealing.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Students can choose different storage locations (e.g., physical, shared, cloud) for files based on the type of file, storage requirements, and sharing requirements. - Students demonstrate an understanding of basic word processing skills including highlighting, cutting, copying, and pasting text within a document. - Students can format and align text, insert and resize images, and utilize built-in spell check and grammar check tools while using a word processing app. - Students can create presentations that demonstrate their ability to choose a theme, insert text boxes, create bulleted and numbered lists, insert and resize images, and use drawing features. - Students can use accurate terminology while identifying and discussing simple hardware or software problems with their peers and teachers. - Students can apply troubleshooting strategies, such as rebooting the device, checking the power, and forcing a shutdown of an application.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Students can choose the appropriate apps to perform a variety of tasks, such as publishing a digital story, creating an audio or video recording, and preparing a digital presentation. - Students can recognize when there is a simple problem with their hardware or software that is preventing them from working as expected and explain the issue to a classmate or teacher.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Students demonstrate understanding the meanings of the following words: font, text box, format, align, software, hardware, and troubleshooting.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Fifth Grade Computer Science

Course Overview

Grade level(s): Fifth Grade	Credits earned: N/A (taught 40 minutes on alternating days -- A/B -- for one semester)
Course Description	
In 5th grade, students will expand on the skills they already learned in grades K-4. They will focus on digital citizenship, coding, and digital publication. They will understand how to communicate respectfully online and protect their private information. Students will understand the implications of the digital footprint and that it will never go away. They will also expand their coding skills and include sequencing, repetition, conditionals, and variables in their coding programs. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> • Develop systematic routines to solve problems • Follow grade-level appropriate behaviors while participating in an online community • Use the engineering design process to plan the development of a program • Use algorithms to simplify code. • Create a document or presentation with advanced formatting operations	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards • 3-5 Computing Systems: ◦ Troubleshooting 01 • 3-5 Networks & the Internet: ◦ Cybersecurity 02 • 3-5 Data & Analysis: ◦ Collection, Visualization & Transformation 01 • 3-5 Algorithms & Programming: ◦ Algorithms 01 ◦ Control 01 ◦ Program Development 03, 04 • 3-5 Impacts of Computing: ◦ Social Interactions 01 ◦ Safety, Law, & Ethics 02	International Society of Technology in Education (ISTE) • Creative Communicator ◦ 1.6a: Choose Platforms or Tools ◦ 1.6c: Communicate Complex Ideas

Unit 1: Digital Citizenship***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 3-5 Impacts of Computing: - Social Interactions 01 - Safety, Law, & Ethics 02	<i>Students will understand that...</i> - There are positive and negative impacts about what they share online - There are systematic routines to solve problems that deal with technology	<i>Students will consider...</i> - What is a hardware and software problem? - Why do certain strategies for solving hardware and software problems work? - What are different cybersecurity problems that could be caused by information that is published? - What is a digital footprint?
Learning Targets		
<i>Students will...</i> - Identify complex hardware and software problems that occur in everyday use - Discuss hardware and software problems with others - Apply strategies for solving hardware and software problems and explain why the strategy should work - Discuss cybersecurity problems caused by information that is published for different reasons - Develop a code of conduct that explains grade-level appropriate behaviors while participating in an online community - Adhere to the code of conduct - Identify and report cyberbullying - Understand the implications of positive and negative digital footprint		
Unit Duration:		
5 days		

Unit 2: Coding

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 3-5 Algorithms & Programming: - Algorithms 01 - Control 01 - Program Development 03, 04	<i>Students will understand that...</i> - There are algorithms to use while coding that are more efficient than others. - The engineering design process is used to plan the development of a program	<i>Students will consider...</i> - How can I simplify multiple algorithms with a programming language? - Which algorithm is most efficient? - What is the engineering design process? - How can I create a program in a programming language?
Learning Targets		
<i>Students will...</i> - Compare and simplify multiple algorithms with programming language - Determine which algorithm is most efficient - Create a program using control structures, event handlers, and variables to solve a problem or express ideas independently and collaboratively - Use the engineering design process to plan the development of a program that includes buttons, backgrounds, transitions, settings, etc - Analyze, examine, create, and debug a program that includes sequencing, repetition, conditionals, and variables in a programming language.		
Unit Duration:		
19 days		

Unit 3: Digital Publication

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 3-5 Computing Systems: - Troubleshooting 01 3-5 Networks & the Internet: - Cybersecurity 02 3-5 Data & Analysis: - Collection, Visualization & Transformation 01 International Society of Technology in Education (ISTE) - Creative Communicator 1.6a: Choose Platforms or Tools	<i>Students will understand that...</i> - Certain presentations are better suited to be a slideshow or a document. - Information can be collected to support a claim	<i>Students will consider...</i> - What process would you use to insert and manipulate graphics? - What process would you use to create and format a table using word processing application software?
Learning Targets		
<i>Students will...</i> - Create a document that demonstrates an understanding of advanced formatting operations - Create a slideshow that demonstrates an understanding of advanced formatting operations - Organize and present collected data visually to highlight comparisons and support a claim		
Unit Duration:		
12-15 days		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
Ongoing	Keyboarding	Students will demonstrate improvement in speed and accuracy of keyboarding	Typing.com	- Monthly typing assessments to determine growth. - Keyboarding goal tracking sheet
.5	Expectations	Students will be able to identify and adhere to the expectations set for the computers classroom	Presentation for expectations, school matrix, classroom matrix	Observe students adhering to the expectations.
1	Digital Citizenship: Hardware/software	- Students will be able to collaborate with a group to creatively dramatize a skit outline - Students will be able to define various internal and external computer parts - Students will be able to understand how internal and external computer parts function as a system - Students will be able to identify complex hardware and software problems that occur in everyday use - Students will be able to discuss hardware and software problems with others - Students will be able to apply strategies for solving hardware and software problems and explain why the strategy should work	Code.org Unplugged activity	- Assessment provided by lesson plan - Observation of skit performed
1	Digital Citizenship: Cyberbullying	- Students will be able to recognize similarities and differences between in-person bullying, and being mean - Students will be able to empathize with the targets of cyberbullying - Students will be able to identify strategies for dealing with cyberbullying and ways they can be an upstander for those being bullied - Students will be able to develop a code of conduct that explains grade-level appropriate behaviors while participating in an online community - Students will be able to adhere to the code of conduct	Common Sense Media	- Cyberbullying quiz- provided by lesson plan online - Create a code of conduct

.5	Digital Citizenship: Relationships and Communication	- Students will be able to compare and contrast different kinds of online-only friendships - Students will be able to describe the benefits and risks of online-only friendships - Describe how to respond to an online-only friend if the friend asks something that makes them uncomfortable	Common Sense Media	Quiz/handout- provided by lesson plan online
.5	Digital Citizenship: Digital Footprint	- Students will be able to define “gender stereotype” and describe how they can be present online - Students will be able to describe how gender stereotypes can lead to unfairness or bias - Students will be able to create an avatar and a poem that show how gender stereotypes impact who they are. - Students will be able to understand the implications of positive and negative digital footprints	Common Sense Media	Quiz/handout- provided by lesson plan online
.5	Digital Citizenship: Privacy and Security	- Students will be able to define “the curiosity gap” - Students will be able to explain how clickbait uses the curiosity gap to get your attention - Students will be able to use strategies for avoiding clickbait - Students will be able to discuss cybersecurity problems caused by information that is published for different reasons	Common Sense Media	Quiz/handout- provided by lesson plan online
.5	Digital Publication: Qualities for Success	Students will be able to reflect on what qualities make them successful in school, their community, home, and a future career	Career Trek	Qualities for success worksheet provided by lesson
.5	Digital Publication: Employability Skills	- Students will be able to reflect on their “good at” skills that make for a good job candidate - Students will be able to reflect on their “need to improve” skills that need to be worked on to make them a better job candidate	Career Trek	Employability skills worksheet
.5	Digital Publication Student Resume	Students will be able to identify the different parts of a resume and its importance	Career Trek	Student resume worksheet
1	Digital Publication: Docs	Students will be able to create a document that demonstrates an understanding of advanced formatting operations.	- B.E. Publishing - The Google Docs Specialist-Unit 1	Google Doc about their work place/ business

			Docs- Project 1.6 Menu Item List	
1	Digital Publication: Slides/Presentations	Students will be able to create a slideshow presentation that demonstrates an understanding of advanced formatting operations	- B.E. Publishing - The Google Docs Specialist-Unit 3 Slides- Project 3.3 Digital Menu Sign	Google Slides about their work place/ business
1	Digital Publication: Forms	Students will be able to organize and present collected data visually to highlight comparisons and support a claim	- B.E. Publishing - The Google Docs Specialist-Unit 4 Forms- Project 4.3 Smoothie Contest Voting Ballot	Google Form about their work place/ business
1-2	Digital Publication: Project	Students will be able to create a portfolio of all the document types for their business	Google Docs, Google Forms, Google Slides, Google Sheets,	Portfolio of all things for their job
.5	Coding: Sprite Lab	- Students will be able to describe the importance of the user in the design process - Students will be able to identify sprites in a running computer program	Code.org Course F Lesson 2	Observation of coding on their project
.5	Coding: Making Sprites	- Students will be able to create an animation using sprites and assign them costumes and behaviors - Students will be able to create new sprites and assign them costumes and behaviors - Students will be able to identify ways to refine an algorithm to include more advanced features	Code.org Course F Lesson 3	Observation of coding on their project
.5	Coding: Sprites in Action	- Students will be able to create an interactive animation using events - Students will be able to develop programs that respond to timed events - Students will be able to develop programs that respond to user input	Code.org Course F Lesson 4	Observation of coding on their project
1	Coding: Mini-Project: Virtual Pet	- Students will be able to create an interactive virtual pet using events, behaviors, and custom art - Students will be able to program solutions to problems that arise when designing a virtual pet,	Code.org Course F Lesson 5	Virtual Pet project

		like feed it		
.5	Coding: Blank Space Stories	- Students will be able to assign a variable a value - Students will be able to call variable multiple times in a program - Students will be able to declare a variable - Students will be able to determine the relationship between how a variable is defined, stored, and retrieved when we press “run” on a program	Code.org Course F Lesson 6	Observation of coding on their project
.5	Coding: Text and Prompts	- Students will be able to use variables in conjunction with prompts - Students will be able to use variables to hold words and phrases	Code.org Course F Lesson 7	Observation of coding on their project
1	Coding: Mini-Project User Input Programs	- Students will be able to Create an interactive computer program that responds to user input - Students will be able to use variables in conjunction with prompts - Students will be able to use variables to hold words and phrases	Code.org Course F Lesson 8	Program created
.5	Coding: Lots of Sprites	Students will be able to create sprite lab programs where sprites are created in groups, and controlled individually using events	Code.org Course F Lesson 9	Observation of coding on their project
.5	Coding: Counting with Variables	- Students will be able to create a clicker game in Sprite Lab where sprites can be removed to score points - Students will be able to create a variable that stores information and changes over time	Code.org Course F Lesson 10	Observation of coding on their project
1	Coding: Mini-project Collector Game	Students will be able to create their own collector game in Sprite Lab using multiple sprites which are controlled individually and variables to track the player’s score	Code.org Course F Lesson 11	Collector Game
.5	Coding: Simulating Experiments	- Students will be able to create double line graph to compare data about two different sources - Students will be able to make and test a prediction by modifying simulation variables - Students will be able to use a computer simulation to collect data about a model	Code.org Course F Lesson 12	Observation of coding on their project

.5	Coding: Outbreak	- Students will be able to identify viruses and how they spread - Students will be able to help slow the spread of a virus	Code.org Course F Lesson 13	Observation of coding on their project
.5	Coding: AI for Oceans	- Students will be able to discuss the role artificial intelligence plays in their lives - Students will be able to reason about how humans bias plays a role in machine learning - Students will be able to train and test a machine learning model	Code.org Course F Lesson 14	“How could artificial Intelligence be used to solve a problem in the world?” exit ticket
.5	Coding: The internet	- Students will be able to learn about the complexity of sending messages over the internet - Students will be able to Translate URLs into IP addresses	Code.org Course F Lesson 15	Assessment Worksheet provided by lesson plan
1	Coding: End of Course Project	- Students will be able to articulate the design process and how it helped shape the finished culminating project - Students will be able to explain how system limitations can affect project design - Students will be able to describe how compromise can help keep a project on track and inspire creativity - Students will be able to draft and implement plans to resolve any issues in their code - Students will be able to learn to plan in advance for an ongoing assignment	Code.org Course F Lesson 16	End of course project

Proficiency Scales for Computers Fifth Grade

Grade Level/Course: Computer Science-Fifth Grade	Standard(s): ● MO Computer Science Standards ○ 3-5 Impacts of Computing: SI 01, SLE 02 ○ 3-5 Computing Systems: T 01 ○ 3-5 Networks & the Internet: C 02 ● International Society of Technology in Education (ISTE) ○ Creative Collaborator 1.6 c Domain: Digital Citizenship
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	● Create a flyer that promotes digital citizenship in an online community. ● Create a slideshow and teach others about digital citizenship. ● Create and dramatize in a skit about proper digital citizenship usage.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. ● Identify, using accurate terminology, more complex hardware and software problems that may occur during everyday use. Discuss problems with peers and adults, apply strategies for solving these problems and <u>explain why the strategy should work</u>. ● Discuss cybersecurity problems caused by information that is published for different reasons. ● Develop a code of conduct that explains grade-level appropriate behaviors while participating in an online community (that includes talking safely online, promoting good digital citizens, privacy settings, and cyberbullying) ● Adhering to the code of conduct that was previously developed ● Identifying and reporting inappropriate behavior and know how to report cyberbullying ● Understand the implications of positive and negative digital footprints and that they never go away	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. ● Identify, using accurate terminology, more complex hardware and software problems that may occur during everyday use. Discuss problems with peers and	

	adults and apply strategies for solving these problems. (such as rebooting the computing device, checking the power, force shut down of an application) • Identify vocabulary terms associated with cybersecurity problems caused by information that is published for different reasons. • Understand what a code of conduct is and how to adhere to it • Understands the difference between appropriate and inappropriate behaviors online. • Understand the implications of positive and negative digital footprints	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. • 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. (such as refreshing the screen, closing and reopening an application or file, unmuting or adjusting the volume) • Identify cybersecurity problems caused by information that is published for different reasons. • Demonstrates an understanding of basic vocabulary such as: digital citizen, privacy settings, cyberbullying, digital footprint, • Understand what a code of conduct is and what it means to adhere to it • Identify appropriate and inappropriate behaviors online • Understand the implications of positive digital footprints	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Fifth Grade	Standard(s): - MO Computer Science Standards 3-5 Algorithms & Programming: A 01, C 01, PD 01, 03 Domain: Coding
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Coding Cards for creating a program and debugging their own codes. - Create a code and debug issues that arise. - Create a plan using a graphic organizer for an If/Then coding story. - Create an If/Then story with coding that includes sequencing, repetition, conditionals, and variables.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Compares AND simplifies multiple algorithms with a programming language AND determine which is the most efficient (ex: adding a button with your own code) - Create a program using control structures, event handlers and variables to solve a problem or express ideas both independently AND collaboratively. - Use the engineering design process to plan the development of a program that includes other perspectives and user preferences (buttons, backgrounds, transitions, settings, etc.) while solving simple problems. - Analyze, examine, create, and debug a program that includes sequencing, repetition, conditionals, and variables in a programming language.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Understands basic terminology from unit- algorithms, control structures, event handlers, variables, programs, debug, sequencing, repetition, conditionals, programming language - Compares AND simplifies multiple algorithms for accomplishing the same task with a programming language - Create a program using control structures to solve a problem or express ideas both independently AND collaboratively. - Use the engineering design process to plan the development of a program that includes user preferences (buttons, backgrounds, transitions, settings, etc.) while solving simple problems. - Analyze, create, and debug a program that includes sequencing, repetition, conditionals, and variables in a programming language.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal.	

	• Compare multiple algorithms for accomplishing the same task with a programming language • Collaboratively create a program using control structures to make decisions within a program. • Identifies the steps of the engineering design process: Ask, Explore, Model, Evaluate, Explain • Analyze and debug a program that includes sequencing, repetition and variables in a programming language.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computer Science-Fifth Grade	Standard(s): - MO Computer Science Standards 3-5 Data and Analysis: CVT 01 - International Society of Technology in Education (ISTE) - Creative Collaborator 1.6 a, c Domain: Digital Publication
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Create a flyer that promotes digital citizenship in an online community. - Create a slideshow that teaches others about digital citizenship.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Create a document and slideshow that communicate a message that demonstrates an understanding of advanced formatting operations including enhancement features such as: alignment, highlight text, bold, italicize, underline, bulleted/numbered list, tables, slide transitions and animations, layout - Organize and present collected data visually to highlight comparisons and support a claim - Demonstrate Growth in keyboarding	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Create a document and slideshow that communicate a message that demonstrates an understanding of basic formatting operations such as: font size, font color, font style, insert images/graphics, - Organize and present collected data visually to highlight comparisons.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Identify where the menu bar is for basic formatting operations. - Understands basic formatting vocabulary such as font size, font color, font style, alignment, insert images/graphics, tables - Collect and organize data in various formats	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Sixth Grade Computer Science <i>Course Overview</i>	
Grade level(s): Sixth Grade	Credits earned: N/A (taught 40 minutes on alternating days -- A/B -- for one semester)
Course Description	
Students in 6th grade will expand on the skills they already learned in grades K-5. They will focus on digital citizenship, coding, and digital publication. While learning to be digital citizens, they will focus on cybersecurity and troubleshooting. They will create variables, loops, conditional statements, and control structures while creating their own coding programs. Additionally, students will gain hands-on experience using various tools and applications to create and publish their digital work, enhancing their ability to present ideas clearly and effectively.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> • Make decisions to troubleshoot problems • Compare tradeoffs associated with security and computing technologies • Make decisions about their coding options • Collect and display data	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards • 6-8 Computing Systems: ◦ Troubleshooting 01 • 6-8 Networks & the Internet: ◦ Cybersecurity 01 • 6-8 Data & Analysis: ◦ Visualization & Transformation 01 • 6-8 Algorithms & Programming: ◦ Variables 01 ◦ Control 01 ◦ Program Development 05 • 6-8 Impacts of Computing: ◦ Culture 01 ◦ Safety, Law, & Ethics 01	

Unit 1: Digital Citizenship***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Impacts of Computing: - Culture 01 - Safety, Law, & Ethics 01	<i>Students will understand that...</i> - There is a routine for troubleshooting problems - There are certain characteristics to look for to determine computer threats - There are tradeoffs to having information public and private - There are tradeoffs with technology that have impacts on people's lives.	<i>Students will consider...</i> - What is troubleshooting? - How can you troubleshoot? - What is a computer threat? - How can I protect my information online? - What happens if my information is public? - How can computing technologies impact people's activities, careers, and lives?
Learning Targets		
<i>Students will...</i> - Develop a systematic troubleshooting routine to identify the problem - Research solutions to troubleshooting problems - Fix problems with computing devices, components, and software - Recognize and determine computer threats - Identify programs and methods to protect electronic information - Compare tradeoffs associated with computing technologies that have impacted people's activities, careers and lives - Describe tradeoffs between allowing information to be public and keeping information private and secure		
Unit Duration:		
12 days		

Unit 2: Coding

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Algorithms & Programming - Variables 01 - Control 01 - Program Development 05	<i>Students will understand that...</i> - Code blocks and loops can be used to shorten codes - Compound conditionals should be used to reach a desired result within a code	<i>Students will consider...</i> - When would you need to make your own code blocks? - What is a loop and a compound conditional? - How can I create a program in a programming language? - How can I break down larger tasks within a partnership and assign roles?
Learning Targets		
<i>Students will...</i> - Create clearly named variables by making their own code blocks to store and manipulate information. - Design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements) - Manage project tasks and timelines when collaboratively developing computational artifacts.		
Unit Duration:		
23 Days		

Unit 3: Digital Publication***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Computing Systems - Troubleshooting 01 6-8 Networks & the Internet - Cybersecurity 01 6-8 Data & Analysis - Visualization & Transformation 01	<i>Students will understand that...</i> - Spreadsheets and forms can be used to organize and collect data - Information can be collected to support a claim - Basic formulas can be used in cells on a spreadsheet - Basic formatting operations should be used on spreadsheets to make data collected easier to display and understand	<i>Students will consider...</i> - What is a spreadsheet? - What are the fundamentals of creating an equation? - What process would you use to create charts and graphs using spreadsheet application software? - What process would you use to create and format a table using word processing application software?
Learning Targets		
<i>Students will...</i> - Collect data using computational tools - Display data in an easy to understand way. - Create a spreadsheets/forms to organize and present collected data to highlight comparisons and support a claim - Demonstrates adding a basic formula to cells in a spreadsheet (addition, subtraction, multiplication, division) - Demonstrates basic formatting operations in a spreadsheet such as merge cells, add borders, & change the cell size.		
Unit Duration:		
12 days		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
ongoing	Keyboarding	Students will demonstrate improvement in speed and accuracy of keyboarding	- Nitrotype Typing.com	- Monthly typing assessments to determine growth. - Keyboarding goal tracking sheet
.5	Expectations	Students will be able to identify and adhere to the expectations set for the computers classroom	Presentation for expectations, school matrix, classroom matrix	Observe students adhering to the expectations.
.5	Digital Citizenship: Troubleshooting	- Students will be able to develop a systematic troubleshooting routine to identify the problem - Students will be able to research solutions to troubleshooting problems - Students will be able to fix problems with computing devices, components, and software	DART Lesson Plan 19: Troubleshooting Basics	Scavenger hunt provided by lesson plan
1	Digital Citizenship:	Students will be able to compare tradeoffs associated with computing technologies that have impacted people's activities, careers, and lives	- B.E. Publishing - Cyber Literacy for the Digital Age- Ch. 10- Online Shopping E-Commerce Unveiled	- Online Purchase Poster - E Commerce Site - Shopping Safely Quiz or Collage
1	Digital Citizenship: Cyber Careers and the future of technology	Students will be able to compare tradeoffs associated with computing technologies that have impacted people's activities, careers, and lives	- B.E. Publishing: - Cyber Literacy for the Digital Age- Chapter 16-Cyber Careers and the future of technology	- Research Virtual Careers- pg 239 in teacher edition, 225 in student pages - Letter to your future self
1	Digital Citizenship: Cyberbullying	- Students will be able to reflect on how easily drama can escalate online - Students will be able to Identify de-escalation strategies when dealing with digital drama - Students will be able to reflect on how digital drama can affect not only oneself but also those around us	Common Sense Media	Cyberbullying quiz- provided by the online lesson

.5	Digital Citizenship: Balance in Digital World	- Students will be able to reflect on their common online and offline activities - Students will be able to identify ways to “unplug” to maintain balance between online and offline activities - Students will be able to use the digital habits checkup routine to create a personal challenge to achieve more media balance	Common Sense Media	Quiz/handout- provided by lesson plan online
1	Digital Citizenship: Privacy and Security	- Students will be able to compare and contrast identity theft with other kinds of theft - Students will be able to describe different ways that identity can occur online - Students will be able to use message clues to identify examples of phishing. - Students will be able to recognize and determine computer threats - Students will be able to identify programs and methods to protect electronic information	Common Sense Media	- Quiz/handout- provided by lesson plan online - Have students create their own phishing email
.5	Digital Citizenship: Digital Footprint and Identity	- Students will be able to reflect on reasons why people might create fake social media accounts - Students will be able to identify the possible results of posting from a fake social media account - Students will be able to debate the benefits and drawbacks of posting from multiple accounts - Students will be able to describe tradeoffs between allowing information to be public and keeping information private and secure	Common Sense Media	Quiz/handout- provided by lesson plan online
.5	Digital Citizenship: Relationships and Communication	- Students will be able to analyze how well they know the people they interact with online - Students will be able to reflect on what information is safe to share with different types of online friends - Students will be able to recognize red flag feelings	Common Sense Media	Quiz/handout- provided by lesson plan online

1	Digital Publication:	Students will be able to display data in an easy to understand way	- B.E. Publishing - The Google Docs Specialist-Unit 1 Docs- Project 1.4 Price List - The Google Docs Specialist-Unit 1 Docs-Project 1.15 Sporting Goods Equipment List - Canva	Google Doc created from their lesson plan
2	Digital Publication	- Students will be able to create a spreadsheet/form to organize and present collected data to highlight comparisons and support a claim - Students will be able to demonstrate adding a basic formula to cells in a spreadsheet (addition, subtraction, multiplication, division) - Students will be able to demonstrate basic formatting operations in a spreadsheet such as merge cells, add borders, and change the cell size	- B.E. Publishing - The Google Docs Specialist-Unit 2 Sheets-Project 2.1- Employee Contact Information - The Google Docs Specialist-Unit 2 Sheets-Project 2.4- New Product Sales - The Google Docs Specialist-Unit 2 Sheets-Project 2.7 Grand Opening Budget	Google sheet created from their lesson plan
1	Digital Publication	- Students will be able to collect data using computational tools - Students will be able to display data in an easy to understand way - Students will be able to create a spreadsheet/form to organize and present collected data to highlight comparisons and support a claim	- B.E. Publishing - The Google Docs Specialist-Unit 4- Forms-Project 4.1 Customer Satisfaction Survey	Google form created from their lesson plan
2	Digital Publication	- Students will be able to collect data using computational tools - Students will be able to display data in an easy to understand way - Students will be able to create a spreadsheet/form to organize and present collected data to highlight comparisons and support a claim - Students will be able to demonstrate adding a basic formula to cells in a spreadsheet (addition, subtraction, multiplication, division)	- B.E. Publishing - Burger Shack	Burger Shack Portfolio

		Students will be able to demonstrate basic formatting operations in a spreadsheet such as merge cells, add borders, and change the cell size		
.5	Coding: Drawing in Game Lab	Students will be able to draw and place shapes, fill in shapes and change colors of shapes, and debug coordinate errors	Code.org- Self Paced Introduction to Game Lab- Lesson 1	Debugging assessment
.5	Coding: Shapes and Parameters	Students will be able to debug code with specific parameters	Code.org- Self Paced Introduction to Game Lab- Lesson 2	Debugging sky
1	Coding: Variables	Students will be able to create clearly named variables by making their own code blocks to store and manipulate information	Code.org- Self Paced Introduction to Game Lab- Lesson 3	Smiley Face Variables
.5	Coding: Random Numbers	- Students will be able to create clearly named variables by making their own code blocks to store and manipulate information - Students will be able to change variable numbers	Code.org- Self Paced Introduction to Game Lab- Lesson 4	Rainbow Snake
.5	Coding: Sprites	Students will be able to create a code to create your own sprite	Code.org- Self Paced Introduction to Game Lab- Lesson 5	Create your own sprites assessment
.5	Coding: Sprite Properties	Students will be able to edit the code of sprites to make objects bigger or smaller	Code.org- Self Paced Introduction to Game Lab- Lesson 6	Make it Fit Assessment
.5	Coding: Text	Students will be able to add text to a scene	Code.org- Self Paced Introduction to Game Lab- Lesson 7	Add Text Assessment
.5	Coding: The Draw Loop	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 8	Salt Shaker Loops assessment

.5	Coding: Sprite Movement	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 9	Fish Animation
.5	Coding: Conditionals	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 10	Magic Dinosaur
.5	Coding: Keyboard Input	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 11	Observation of Code.org where they have a sprite moving around using arrow keys
.5	Coding: Mouse Input	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 12	Shaking sprite
.5	Coding: Velocity	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 13	Swimming Left and Right
.5	Coding: Collision Detection	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 14	Rainbow Horse
.5	Coding: Complex Sprite Movement	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 15	Gravity

.5	Coding: Collisions	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 16	Sprite Interactions
1	Coding: Functions	Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements)	Code.org- Self Paced Introduction to Game Lab- Lesson 17	Collector Game
1-2	Coding: Project- Design a Game	- Students will be able to create clearly named variables by making their own code blocks to store and manipulate information - Students will be able to design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements) - Students will be able to manage project tasks and timelines when collaboratively developing computational artifacts	- Code.org- Self Paced Introduction to Game Lab- Lesson 18 - Scratch	Final Game

Proficiency Scales for Computers Sixth Grade

Grade Level/Course: Computers Sixth Grade	Standard(s): • MO Computer Science Standards ○ 6-8 Impacts of Computing: C 01, SLE 01 ○ 6-8 Computing Systems: T 01 ○ 6-8 Networks & the Internet: C 01 Domain: Digital Citizenship
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	• Have students identify steps they would take to troubleshoot different issues • Have students create Scam emails that have various phishing features • Create slides that discuss what could happen if information is public vs. private and how that has impacted people's lives , careers and activities
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. • Develop a systematic troubleshooting routine to identify the problem, research solutions and fix problems with computing devices, components and software • Recognize and determine computer threats and be able to identify programs and methods to protect electronic information. • Compare tradeoffs associated with computing technologies (security, efficiency, convenience) that have impacted people's activities, careers and lives when solving global problems using the power of computing • Describe tradeoffs between allowing information to be public and keeping information private and secure	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. • Troubleshoots common strategies to identify the problem, research solutions and fix problems with computing devices, components, and software (such as rebooting the computing device, checking the power, force shut down of an application, checking the wifi connection, reloading the website, etc) • Identify common computer threats and be able to identify common programs and methods to protect electronic information • Understands vocabulary terms- tradeoff, online security, private information, scam, identity theft, phishing, • Identify and describe some tradeoffs between allowing information to be public and keeping information private and secure (by using secure sites- with the lock)	

1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. • Troubleshoots to try to identify the problem, research solutions and fix problems with computing devices, components, and software • Identify common computer threats OR be able to identify common programs and methods to protect electronic information • Identify some computing technologies (such as the internet, online communities, online shopping, digital publications, digital devices, etc.) and how they have had an impact on other people's lives • Identify what information is public and private	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computers Sixth Grade	Standard(s): - MO Computer Science Standards 6-8 Algorithms & Programming: V 01, C 01, PD 01 Domain: Coding
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Create a game on code.org or scratch with coding that includes sequencing, repetition, conditionals, and variables. - Create a code and debug issues that arise. - Create a plan using a graphic organizer for creating a game - Create code blocks to use in coding projects
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Create clearly named variables by making their own code blocks to store and manipulate information. Design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements) - Manage project tasks and timelines when collaboratively developing computational artifacts.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Create variables by making their own code blocks - Use combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements) - Identify when tasks need to be completed when collaboratively developing computational artifacts	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Identify where variables exist on their coding programs - Identify when to use combinations of control structures and where those blocks are located, such as multiple loops (repeat blocks) and compound conditionals (if then statements) - Takes direction from group members when collaboratively developing computational artifacts	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Computers Sixth Grade	Standard(s): - MO Computer Science Standards 6-8 Data & Analysis: VT 01 - International Society of Technology in Education (ISTE) - Creative Collaborator 1.6 a, c Domain: Digital Publication
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Create a Google form survey to collect information - Use this information to create a spreadsheet to display this data - Develop a claim about this data - Use various sheets features on a google sheets assignment
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Collect data using computational tools and display it for the end user in an easy to understand way. - Create a spreadsheets/forms to organize and present collected data to highlight comparisons and support a claim (i.e. Choose correct charts/graphs to display data) - Demonstrates adding a basic formula to cells in a spreadsheet (addition, subtraction, multiplication, division) - Demonstrates basic formatting operations in a spreadsheet such as merge cells, add borders, & change the cell size.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Identify the correct name of spreadsheet cells (i.e. A1, B12, C4, etc.) - Collect data using computational tools and display it	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Identify where the menu bar is for basic formatting operations. - Collect data and display it	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

PLTW Design and Modeling <i>Course Overview</i>	
Grade level(s): 7th Grade or 8th Grade	Credits earned: N/A (taught for one semester at middle school)
Course Description	
This semester-long Project Lead the Way course is a beginning course for engineering. Students discover the design process and develop an understanding of the influence of creativity and innovation in their lives. Students learn a variety of collaboration, critical thinking, creativity and communication skills. Students also learn to use tools to create projects. They are then challenged and empowered to use and apply what they've learned throughout the unit to design a therapeutic device for a child who has a medical issue. This course allows students to explore all three PLTW pathways---Engineering, Computer Science, and Biomedical---in a fun and hands-on experience.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to...</i> - Analyze issues and develop effective solutions. - Foster critical thinking and creativity by designing and innovating new ideas. - Apply technical skills through hands-on projects, utilizing tools and understanding how systems work. - Collaborate effectively by enhancing teamwork and communication in engineering challenges. - Develop project management skills by planning, organizing, and managing time efficiently to complete complex tasks in both daily life and future careers.	
Priority Missouri Learning Standards/National Standards	
Missouri Science Standards 6-8 Engineering, Technology, and Application of Science - Defining and Delimiting Engineering Problems 1.A.1 - Developing Possible Solutions 1.B.1, 2, 3 MO Computer Science Standards 6-8 Data and Analysis - Collection, Visualization, and Transformation 01	

Unit 1: Defining and Delimiting Engineering Problems

Desired Results

Standards	Enduring Understandings	Essential Questions
Missouri Science Standards 6-8 Engineering, Technology, and Application of Science - Defining and Delimiting Engineering Problems 1.A.1	<i>Students will understand that...</i> - Effective design solutions require a clear definition of the problem, including specific criteria and constraints that guide the development process. - Scientific principles are foundational to engineering practices, and that they must apply relevant scientific knowledge to create viable solutions - Decisions can have significant effects on people and the natural environment, and that responsible design considers these impacts to promote sustainability and well-being. - Design process is iterative, meaning that solutions can be improved through ongoing evaluation and refinement based on scientific understanding and feedback. - Value of teamwork and collaboration, recognizing that diverse perspectives and skills enhance the design process and lead to more innovative and effective solutions.	<i>Students will consider...</i> - What are criteria and constraints, and why are they important in the design process? - How can we define a design problem clearly to ensure a successful solution? - What scientific principles should we consider when designing a solution to a problem? - How do constraints such as materials, time, cost, and technology impact the design process? - In what ways can a design solution affect people and the environment, both positively and negatively? - How can engineers balance innovation with ethical and environmental considerations in their designs? - What strategies can we use to evaluate whether a design meets all criteria and constraints effectively? - How do trade-offs influence the decision-making process in engineering design?
Learning Targets		
<i>Students will...</i> - Define the engineering problem with key constraints identified. - Describe the impact of their design solution on people and the natural environment.		
Unit Duration:		
Ongoing		

Unit 2: Developing Possible Solutions - Defining Criteria and Constraints

Desired Results

Standards	Enduring Understandings	Essential Questions
Missouri Science Standards 6-8 Engineering, Technology, and Application of Science - Developing Possible Solutions 1.B.1 MO Computer Science Standards 6-8 Data and Analysis - Collection, Visualization, and Transformation 01	<i>Students will understand that...</i> - Not all solutions are equally effective. - A systematic process ensures fair evaluation. - Trade-offs are part of decision-making. - Testing and data drive design decisions. - Continuous improvement leads to better designs	<i>Students will consider...</i> - How can we compare different design solutions objectively? - What criteria should be used to determine the best solution for a given problem? - How do constraints (such as cost, time, and materials) impact the evaluation of design solutions? - Why is testing and iteration important when selecting the best design? - What role does evidence play in justifying the selection of one design over another?
Learning Targets		
Students will... Evaluate competing design solutions using a decision matrix		
Unit Duration:		
Ongoing		

Unit 3: Developing Solutions - Analyzing Test Data

Desired Results

Standards	Enduring Understandings	Essential Questions
Missouri Science Standards 6-8 Engineering, Technology, and Application of Science - Developing Possible Solutions 1.B.2 MO Computer Science Standards 6-8 Data and Analysis - Collection, Visualization, and Transformation 01	<i>Students will understand that...</i> - Prototypes and models help test solutions. - Failure is a step toward success. - Testing must be systematic and repeatable. - Data informs better design choices. - Design is an iterative process - Effective data visualization supports informed decision-making by providing clear insights and highlights from raw data, emphasizing the importance of visual literacy in various fields.	<i>Students will consider...</i> - Why is testing a design important before finalizing a solution? - How can failure in testing improve a design? - What methods can be used to systematically test and evaluate a design? - How does data from testing influence design decisions? - Why is the design process iterative, and how does redesign improve solutions? - What role do prototypes and models play in engineering and problem-solving? - How do criteria and constraints affect the way a design is tested and refined? - How does the process of creating a data visualization contribute to our overall understanding of the data itself?
Learning Targets		
<i>Students will...</i> - Develops multiple potential solutions; uses basic comparative analysis to select the best approach. - Creates a prototype that meets most design requirements with refinement. - Shows clear evidence of design iteration and improvement. - Can create a Box and Whiskers graph based on testing data.		
Unit Duration:		
Ongoing		

Unit 4: Developing Solutions - Modeling Development

Desired Results

Standards	Enduring Understandings	Essential Questions
Missouri Science Standards 6-8 Engineering, Technology, and Application of Science - Developing Possible Solutions 1.B.3	<i>Students will understand that</i> - Optimization improves design solutions. - Iteration leads to innovation. - Trade-offs are necessary for improvement. - Data -driven decisions enhance designs. - Technology and tools support optimization	<i>Students will consider...</i> - Why is it important to improve and refine a design? - How do engineers determine if a design needs improvement? - What role does testing and data analysis play in optimizing a solution? - Why is the design process iterative rather than a one-time effort? - How do constraints and trade-offs influence design improvements? - What strategies can be used to make a design more effective or efficient? - How does technology help engineers optimize and refine their designs?
Learning Targets		
<i>Students will...</i> - Develop a model/prototype for testing and modification based on the requirement of a client. - Create a presentation that demonstrates how their solution meets the needs of the client.		
Unit Duration:		
Ongoing		

<i>Learning Plan</i>				
Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
Week 1 Week 2	Introduction to Design	- Define the engineering problem with key constraints identified. - Describe the impact of their design solution on people and the natural environment. - Evaluate competing design solutions using a decision matrix - Develops multiple potential solutions; uses basic comparative analysis to select the best approach. - Creates a prototype that meets design requirements with refinement. - Shows clear evidence of design iteration and improvement. - Can create a Box and Whiskers graph based on testing data. - Develop a model/prototype for testing and modification based on the requirement of a client. - Create a presentation that demonstrates how their solution meets the needs of the client.	- PLTW.org Unit 1 1.1 Foot Orthosis Design Challenge <u>Discover Engineering</u> <u>Science Journal for Kids and Teens</u> <u>PBS NewsHour Daily News Lessons</u> - Youtube - Crash Course - Crash Course Kids - National Science Foundation - Mark Rober <u>PBS Learning Media</u> - Khan Academy	Foot Orthosis Model (Formative Assessment)
Week 3 Week 4 Week 5	Introduction to Design	- Develops multiple potential solutions; uses basic comparative analysis to select the best approach. - Creates a prototype that meets most design requirements with refinement. - Shows clear evidence of design iteration and improvement.	- PLTW.org Unit 1 1.2 Picture is worth a Thousand Words (Design Sketching) 1.3 Measuring Matters 1.4 Skimmer Statistics - Google Sheets	Skimmer Model (Formative Assessment)
Week 6 Week 7	Introduction to Design	- Develops multiple potential solutions; uses basic comparative analysis to select the best approach. - Creates a prototype that meets most design requirements with refinement. - Shows clear evidence of design iteration and improvement.	- PLTW.org Unit 1 1.5 Dialed in (Dial Caliper) 1.6 Investigate the inside (Design Brief)	Carlo's Puzzle (Formative Assessment) PLTW Formative Assessment #1
Week 8 Week 9 Week 10	Solid Modeling	- Define the engineering problem with key constraints identified. - Describe the impact of their design solution on people and the natural environment.	- PLTW.org Unit 2 2.1 Taking Modeling to another Dimension (3D design with Tinkercad)	Bird House (Formative Assessment)

PLTW Design and Modeling (7th or 8th grade)

		• Evaluate competing design solutions using a decision matrix • Define the engineering problem with key constraints identified. • Develops multiple potential solutions; uses basic comparative analysis to select the best approach. • Creates a prototype that meets most design requirements with refinement. • Shows clear evidence of design iteration and improvement. • Can create a Box and Whiskers graph based on testing data. • Develop a model/prototype for testing and modification based on the requirement of a client. • Create a presentation that demonstrates how their solution meets the needs of the client.	○ 2.2 For Good Measure (surface and volume of 3D rectangular prism shapes) ○ 2.3 It's for the Birds (designing 3D birdhouse) • TinkerCad	
Week 11 Week 12 Week 13	Solid Modeling	• Define the engineering problem with key constraints identified. • Describe the impact of their design solution on people and the natural environment. • Evaluate competing design solutions using a decision matrix • Develops multiple potential solutions; uses basic comparative analysis to select the best approach. • Creates a prototype that meets most design requirements with refinement. • Shows clear evidence of design iteration and improvement. • Can create a Box and Whiskers graph based on testing data. • Develop a model/prototype for testing and modification based on the requirement of a client. • Create a presentation that demonstrates how their solution meets the needs of the client.	• PLTW.org Unit 2 ○ 2.4 Puzzle Cube Design Challenge	Puzzle Cube PLTW Formative Assessment #2
Week 14 Week 15	Real -world project challenges	• Define the engineering problem with key constraints identified. • Describe the impact of their design solution on people and the natural environment. • Evaluate competing design solutions using a decision matrix	• Discover Engineering • Science Journal for Kids and Teens • PBS NewsHour Daily News Lessons • Youtube	Articles, videos, and mini challenges to review and reinforce skills

		• Develops multiple potential solutions; uses basic comparative analysis to select the best approach. • Creates a prototype that meets most design requirements with refinement. • Shows clear evidence of design iteration and improvement. • Can create a Box and Whiskers graph based on testing data. • Develop a model/prototype for testing and modification based on the requirement of a client. • Create a presentation that demonstrates how their solution meets the needs of the client.	Crash Course Crash Course Kids National Science Foundation • PBS Learning Media	
Week 16-18	Design Challenge	• Define the engineering problem with key constraints identified. • Describe the impact of their design solution on people and the natural environment. • Define the engineering problem with key constraints identified. • Evaluate competing design solutions using a decision matrix • Develops multiple potential solutions; uses basic comparative analysis to select the best approach. • Creates a prototype that meets most design requirements with refinement. • Shows clear evidence of design iteration and improvement. • Can create a Box and Whiskers graph based on testing data. • Develop a model/prototype for testing and modification based on the requirement of a client. • Create a presentation that demonstrates how their solution meets the needs of the client.	• PLTW.org Unit 3 ○ Therapeutic Toy Design	PLTW Summative Assessment - Therapeutic Toy Design

Proficiency Scales for PLTW Design and Modeling

Grade Level/Course: PLTW Design and Modeling	Standard(s): - MO Science Standards 6.8.ETS1.A.1 Domain: Defining and Delimiting Engineering Problem
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Rubric based project - Design Brief analysis - Peer Review and self assessment - Exit Tickets/Quick Check
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Define the engineering problem with key constraints identified. - Describe the impact of their design solution on people and the natural environment.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Identifies a general problem . - Demonstrated understanding that all solutions have an impact on people and the environment.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Demonstrate an understanding of the engineering terms such as: problem,constraints, requirements, time, budget, resources, etc.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Design and Modeling	Standard(s): - MO Science Standards 6.8.ETS1.B.1 Domain: Developing Possible Solutions - Defining Criteria and Constraints
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Using the design process to create Foot Orthosis for a child with Cerebral Palsy - Researching and brainstorming - Using a decision matrix to find the best solution - Creating a digital/paper journal of the process - Creating a prototype
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. Evaluate competing design solutions using a decision matrix.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Identify criteria and constraints - Compare different solutions	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Demonstrates understanding of criteria and constraints and evaluate them on their own project	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Design and Modeling	Standard(s): - MO Science Standards 6.8.ETS1.B.2 - MO Computer Science Standards 6-8 DA.VT.01 Domain: Developing Possible Solutions - Analyzing test data
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Working with a design brief to identify criteria and constraints. - Brainstorming ideas and using a decision matrix to decide on the best solutions. - digital/Paper journal that shows the building and testing of the prototype. - Creating thumbnail, isometric, and multiview sketches that contain dimensions and annotations.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Develops multiple potential solutions; uses basic comparative analysis to select the best approach. - Creates a prototype that meets design requirements with refinement. - Shows clear evidence of design iteration and improvement. - Can create a Box and Whiskers graph based on testing data.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Demonstrate an understanding of data analysis. - Demonstrate an understanding testing their prototype and evaluating the results. - Demonstrate an understanding of the 5 number summary to create the graph.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Demonstrate an understanding of the steps in the design process. - Define a problem, generate ideas, design solution, build and test, redesign, evaluate, and present solution. - Can define minimum, maximum, quartile 1, lower quartile 2 (center) and upper quartile 3	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Design and Modeling	Standard(s): - MO Science Standards 6.8.ETS1.B.3 Domain: Developing Possible Solutions - Model Development
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Solution Presentation- Google slides, product advertisement, etc. - Solution Sketches - Prototype - Digital/Paper Journal of product testing.
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Develop a model/prototype for testing and modification based on the requirement of a client. - Create a presentation that demonstrates how their solution meets the needs of the client.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can ask questions to clarify the problem. - Can sketch possible solutions to a problem. - Students can present their solutions to the client.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Demonstrate an understanding of the terms client, prototype, problem, criteria, requirements, etc.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

PLTW Computer Science for Innovators and Makers

Course Overview

Grade level(s): 7th Grade or 8th Grade
Credits earned: N/A (taught for one semester at middle school)

Course Description

Computer Science for Innovators and Makers teaches students that programming goes beyond the virtual world into the physical world. Students are challenged to creatively use sensors and actuators to develop systems that interact with their environment. Designing algorithms and using computational thinking practices, they code and upload programs to microcontrollers that perform a variety of authentic tasks. The unit broadens students' understanding of computer science concepts through meaningful applications. Teams select and solve a personally relevant problem related to wearable technology, interactive art, or mechanical devices. Throughout the unit, students learn about programming for the physical world by blending hardware design and software development, allowing students to discover computer science concepts and skills by creating personally relevant, tangible, and shareable projects.

Transfer Goals/Big Ideas

Students will be able to independently use their learning to...

- Follow a design process that can help them effectively develop a physical computing solution
- Ask questions and define problems
- Develop and use models
- Use mathematics and computational thinking
- Construct explanations and design solutions
- Obtain, evaluate, and communicate information
- Use iterative design

Priority Missouri Learning Standards/National Standards

[MO Computer Science Standards](#)

- 6-8 Computing Systems
 - Devices 01
 - Hardware & Software 01
 - Troubleshooting 01
- 6-8 Data & Analysis
 - Collection, Visualization & Transformation 01
- 6-8 Algorithms & Programming
 - Variables 01
 - Modularity 01, 02
 - Program Development 01, 02, 04, 05

Program Development (Planning)***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Algorithms & Programming - Program Development 01	<i>Students will understand that...</i> - Following steps make it possible to develop an effective solution to a problem: Define the problem, Generate concepts, Design a solution, Build and Test, Evaluate your solution, Present your solution. - Computer scientists might look at how one part of a computer interacts with another part, or they might study how people interact with a computer or connected computing systems, like a network or the internet. - Algorithmic thinking skills can help you solve complex problems. - Programmers break down problems into easy-to-follow steps for a computer.	<i>Students will consider...</i> - What is a computer scientist? - How can algorithmic thinking skills be used across multiple disciplines? - How can computer programs solve problems?
Learning Targets		
<i>Students will...</i> - Design and program solution ideas that are generated and planned using pseudocode, sketches, flowcharts, LED planning grids, and wiring diagrams. - Use algorithmic thinking to write specific, clear, and complete directions. - Reflect on the role of a computer scientist. - Break a complex task into a sequence of small steps. - Design and program a computing solution. - Engage in problem-solving and be creative when developing solutions. - Follow a design process to effectively develop a physical computing solution.		
Unit Duration:		
2 weeks; Unit 1: Blink!		

Algorithms and Programming

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Algorithms & Programming - Modularity 01, 02 - Variables 01 6-8 Data & Analysis - Collection, Visualization & Transformation 01	<i>Students will understand that...</i> - The following steps make it possible to develop an effective solution to a problem: Define the problem, Generate concepts, Design a solution, Build and Test, Evaluate your solution, Present your solution. - Programs should be organized, concise, and easy for others to read and understand. - You can express yourself by creating various solutions to problems.	<i>Students will consider...</i> - What do programming best practices look like? - How do you express yourself and your creativity through computer science? - How is a design process used to develop physical computing systems?
Learning Targets		
<i>Students will...</i> - Create programs using inputs and outputs that receive and respond to human and/or environmental interactions. Algorithms in the code progress in a logical sequence for executing an event-driven action or accomplishing a task. Code is written in an efficient and readable fashion. - Explore a coding environment. - Plan a program sequence. - Design and program a computing solution. - Work with others to create and modify programs that use input and output devices. - Develop and modify algorithms that use variables. - Follow a design process to effectively develop a physical computing solution. - Simplify code to make it easier to read.		
Unit Duration:		
Ongoing; Unit 1: Blink!, Unit 2: The Ins and Outs, Unit 3: Program the Physical World		

Troubleshooting (Debugging)*Desired Results*

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Computing Systems - Troubleshooting 01 6-8 Algorithms & Programming - Program Development 04	<i>Students will understand that...</i> - The following steps make it possible to develop an effective solution to a problem: Define the problem, Generate concepts, Design a solution, Build and Test, Evaluate your solution, Present your solution. - Troubleshooting, problem-solving, and debugging are all ways to locate a problem and systematically find a fix or solution.	<i>Students will consider...</i> - How is a design process used to develop physical computing systems? - How can you locate and fix a problem in your program?
Learning Targets		
<i>Students will...</i> - Use code debugging techniques, such as using code tracing charts and adding helpful program comments, to correct program anomalies. - Analyze, revise, and test a program or device hardware to fix errors. - Use code tracing to break down a program into logical parts to look for errors. - Follow a design process to effectively develop a physical computing solution.		
Unit Duration:		
1 week; Unit 1: Blink!		

Computing Systems (Devices, Hardware, and Software)

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Computing Systems - Devices 01 - Hardware & Software 01	<i>Students will understand that...</i> - The following steps make it possible to develop an effective solution to a problem: Define the problem, Generate concepts, Design a solution, Build and Test, Evaluate your solution, Present your solution. - You can express yourself by creating various solutions to problems. - The hardware includes inputs and outputs that send and receive data from the software built into the computer.	<i>Students will consider...</i> - How is a design process used to develop physical computing systems? - How do you express yourself and your creativity through computer science? - How do software and hardware work together to perform a variety of tasks?

Learning Targets

Students will...

- Select hardware that thoroughly meets the needs of the user or environment interactions in order to solve a physical computing problem. The project's program and hardware design are thoroughly demonstrated and explained clearly with visual aids, documentation (planning materials), and code.
- Explore how computers receive, process, and send information.
- Examine the parts of a microcontroller.
- Build a physical computing device.
- Explore wireless technology.
- Follow a design process to effectively develop a physical computing solution.

Unit Duration:

Ongoing; Unit 1: Blink!, Unit 2: The Ins and Outs, Unit 3: Program the Physical World

Program Development (Collaboration)***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8.Algorithms & Programming - Program Development 02, 05	<i>Students will understand that...</i> - The following steps make it possible to develop an effective solution to a problem: Define the problem, Generate concepts, Design a solution, Build and Test, Evaluate your solution, Present your solution. - To work effectively as a team member, you need to use effective communication skills, constructive feedback, team norms, and time management.	<i>Students will consider...</i> - How is a design process used to develop physical computing systems? - What skills are necessary for students to work together effectively with a common purpose to achieve desired results?
Learning Targets		
<i>Students will...</i> - Work effectively as a team and incorporate the strengths of each member. They will use feedback from their team members and manage project tasks and timelines in order to complete their task on time. - Demonstrate effective communication skills. - Work with others to create and modify programs that use input and output devices. - Use pair programming to collaborate. - Collaborate within a team.		
Unit Duration:		
Ongoing; Unit 1: Blink!, Unit 2: The Ins and Outs, Unit 3: Program the Physical World		

Learning Plan

Week(s)	Topic	Resources/Texts	Learning Targets	Assessment
6 weeks	Unit 1: Blink! - Activity 1.1: The Brain - Activity 1.2: What To Do - Activity 1.3: How To Do It - Activity 1.4: Crush the Bug - Project 1.5: The Blinking Message	my.pltw.org app.diagrams.net makecode.microbit.org blooket.com flocabulary.com applieddigitalskills.withgoogle.com https://learn.sparkfun.com/tutorials/tags/micro-bit https://blockly.games/ https://code.org/ https://scratch.mit.edu/	The students will... - Explore how computers receive, process, and send information. - Examine the parts of a microcontroller. - Reflect on the role of a computer scientist. - Break a complex task into a sequence of small steps. - Use algorithmic thinking to write specific, clear, and complete directions. - Demonstrate effective communication skills. - Explore a coding environment. - Plan a program sequence. - Use code tracing to break down a program into logical parts to look for errors. - Analyze, revise, and test a program or device hardware to fix errors. - Design and program a computing solution. - Engage in problem-solving and be creative when developing solutions.	Conclusion Questions: - After learning more about computer science and physical computing, write your new definition for computer science. - How is the processor on the microcontroller and inside a computer like a brain? - When creating physical computing projects, why is it important to have good communication skills? - Why is it important to create an algorithm or flowchart prior to coding when solving a problem? - Other than people who write code, what careers do you think may also use algorithms or flowcharts to help organize their ideas? - Why is it necessary to explore and understand the coding environment before you create your first code? - Why do computer scientists create flowcharts before they code? - Troubleshooting, problem solving, and debugging are all ways to locate a problem and systematically find a fix or solution. Describe a scenario

				in your life where you've had to troubleshoot, debug or problem solve. Describe the problem and your method for solving it. • Why is code tracing important when debugging? • What devices do you or people you know use that may have an accelerometer or compass to perform certain tasks or functions? • What are other ways you could use the shake or compass code blocks in physical computing projects?
8 weeks	Unit 2: The Ins and Outs • Activity 2.1: Need Input • Activity 2.2: Responding Output • Activity 2.3: Get Connected • Project 2.4: Secrets and Safes	• my.pltw.org • app.diagrams.net • makecode.microbit.org • blooket.com • quizizz.com • quizlet.com • edpuzzle.com • tinkercad.com • https://learn.sparkfun.com/tutorials/tags/micro-bit • https://www.101computing.net/category/bbc-microbit/ • https://tinkercademy.com/microbit/ • https://makered.org/resources/cyber-arcade-programming-and-making-with-microbit/ • arcade.makecode.com	The students will... • Work with others to create and modify programs that use input and output devices. • Use pair programming to collaborate. • Build a physical computing device. • Explore wireless technology. • Develop and modify algorithms that use variables. • Follow a design process to effectively develop a physical computing solution. • Collaborate within a team. • Demonstrate effective communication skills.	Conclusion Questions: • Are the copper tape and aluminum foil switches analog or digital devices? How do you know? • Where and how could you use a switch made out of aluminum foil or copper tape? • Describe a situation where it would be advantageous to use the copper tape or aluminum foil instead of the A and B push buttons on the micro:bit. • Think about the three different physical computing problem options presented to you in The Brain activity (activity 1.1). How might learning to program input and output devices help you find a solution to one of the three problem choices?

		• https://code.org/ • https://makecode.adafruit.com/ • https://scratch.mit.edu/		• Name something you or your parents have that accepts at least two inputs and will provide at least two outputs. Specifically, describe what the input is and what the output is. • What might be a benefit of wirelessly connecting smart devices in a physical computing project? • How might you connect more than two micro:bits to activate multiple outputs triggered by one micro:bit? • How do batteries and other types of power sources make physical computing systems more mobile? Why is that helpful? • Test the distance your two micro:bits are able to send data packets without interference. How far apart were your micro:bits able to go before your micro:bit antennas stopped sending data to each other? • Why are failures important when creating computer programs and physical computing devices? • How do you express yourself and your creativity through computer science? • Explain how your team collaborated and communicated on your design solution.
--	--	---	--	--

4 weeks	Unit 3: Program the Physical World - Activity 3.1: Clean Up Your Code - Problem 3.2: Interactions	my.pltw.org app.diagrams.net makecode.microbit.org blooket.com tinkercad.com https://learn.sparkfun.com/tutorials/tags/micro-bit https://www.101computing.net/category/bbc-microbit/ https://tinkercademy.com/microbit/ https://makered.org/resources/cyber-arcade-programming-and-making-with-microbit/ arcade.makecode.com https://code.org/ https://makecode.adafruit.com/ https://scratch.mit.edu/ blockscad3d.com https://www.birdbraintechologies.com/products/finch-robot-2-0/	The students will... - Simplify code to make it easier to read. - Follow a design process to effectively develop a physical computing solution. - Collaborate within a team. - Demonstrate effective communication skills.	Conclusion Questions: - Describe the roles of a navigator and driver in pair programming. - Describe the advantages of creating elegant code. - Compare writing elegant code to writing an essay for your ELA class. How are they similar and different? - How can you continue learning about computer science and improve your coding abilities after completing this unit? - How do physical computing systems impact you, people you know, and society? - In this unit you used the skills of collaboration, problem solving, communication, and creative thinking. How might you use these skills in your daily life? - Look back on all of the Career Connections presented in the unit. What aspects of the different careers interest you and why?
---------	---	--	--	---

Proficiency Scales for Computers PLTW Computer Science for Innovators and Makers

Grade Level/Course: PLTW Computer Science for Innovators and Makers	Standard(s): • MO Computer Science Standards ○ 6-8 Algorithms & Programming: PD01 Domain: Program Development (Planning)
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	• Create pseudocode, sketches, flowcharts, LED planning grids, and wiring diagrams • Summative Assessment - Project 1.5: The Blinking Message • Formative Assessment - Parking Garage Gate Challenge • Summative Assessment - Project 2.4: Secrets and Safes • Summative Assessment - Problem 3.2: Interactions
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. • Design and program solution ideas that are generated and planned using pseudocode, sketches, flowcharts, LED planning grids, and wiring diagrams. • Use algorithmic thinking to write specific, clear, and complete directions.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. • Create algorithms (lists of steps) to complete a task, identify flowchart symbols, understand the difference between linear and conditional flowcharts, and sketch designs.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. • Create algorithms (lists of steps) and sketch designs.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science for Innovators and Makers	Standard(s): - MO Computer Science Standards 6-8 Algorithms & Programming: M 01, 02, V 01 6-8 Data & Analysis: VT 01 Domain: Algorithms and Programming
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Program a blinking, animated message (scrolling text, images, animations) and download it to a micro:bit - Program an interactive security device - Program a physical computing system for wearable technology, interactive digital art, or mechanical devices - Summative Assessment - Project 1.5: The Blinking Message - Lesson 2 Formative Assessment (Quizizz) - Formative Assessment - Parking Garage Gate Challenge - Summative Assessment - Project 2.4: Secrets and Safes - Summative Assessment - Problem 3.2: Interactions
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Create programs using inputs and outputs that receive and respond to human and/or environmental interactions. - Algorithms in the code progress in a logical sequence for executing an event-driven action or accomplishing a task. - Code is written in an efficient and readable fashion.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. Program scrolling messages, images, inputs, and outputs following tutorials and direct instruction	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Program scrolling messages and images, following tutorials and direct instruction	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science for Innovators and Makers	Standard(s): - MO Computer Science Standards 6-8 Computing Systems: T 01 6-8 Algorithms & Programming: PD 04 Domain: Troubleshooting (Debugging)
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Use code tracing charts and add comments to their programs in order to debug their programs 1.4 FORMATIVE ASSESSMENT: BUGGY Code Tracing Chart - Summative Assessment - Project 1.5: The Blinking Message - Summative Assessment - Project 2.4: Secrets and Safes - Summative Assessment - Problem 3.2: Interactions
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Use code debugging techniques, such as using code tracing charts and adding helpful program comments, to correct program anomalies. - Analyze, revise, and test a program or device hardware to fix errors.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. Identify bugs in their programs and develop solutions to debug programs.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Identify bugs in their programs.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science for Innovators and Makers	Standard(s): - MO Computer Science Standards 6-8 Computing Systems: HS 01, D 01 Domain: Computing Systems (Devices, Hardware, and Software)
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Design physical computing solutions to real world problems, such as: - After dark, increased outdoor safety is needed. Design and program a device that provides for a safer condition after dark, is wearable, and can be activated by the wearer or an environmental condition. - A community center wants to develop an interactive art installation that will entertain a variety of people. Your team will design and program an interactive art display. The installation must include lights and possibly movement. It should also be accessible and engaging. - Every day, people and pets need immediate access to items. Your team will design and program a device that dispenses objects when an environmental interaction occurs. The device must have lights to indicate it has been activated and must use a motor to dispense objects. - Summative Assessment - Problem 3.2: Interactions
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Select hardware that thoroughly meets the needs of the user or environment interactions in order to solve a physical computing problem. - The project’s program and hardware design are thoroughly demonstrated and explained clearly with visual aids, documentation (planning materials), and code.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. Connect and program hardware (servos, LEDs, photocell resistors, pressure sensors, flex sensors, buzzers, etc.) individually.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Utilize wiring diagrams to connect hardware (servos, LEDs, photocell resistors, pressure sensors, flex sensors, buzzers, etc.).	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science for Innovators and Makers	Standard(s): - MO Computer Science Standards 6-8 Algorithms & Programming: PD 02, 05 Domain: Program Development (Collaboration)
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Work effectively as a team and equally contribute to solve a problem. - Formative Assessment - Parking Garage Gate Challenge - Summative Assessment - Project 2.4: Secrets and Safes - Summative Assessment - Problem 3.2: Interactions
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Work effectively as a team and incorporate the strengths of each member. - They will use feedback from their team members and manage project tasks and timelines in order to complete their task on time.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Work effectively as a team and incorporate the strengths of each member. - They will ask for assistance when there is a conflict.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Identify when teammates are not working effectively and attempt to find a solution.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Middle School Robotics

Course Overview

Grade level(s): 7th Grade or 8th Grade	Credits earned: N/A (taught for one semester in middle school)
Course Description	
This semester-long class is a beginning course in robotics. We will be utilizing various robotics kits and materials. The objective of this course is to introduce the student to basic programming as well as problem-solving strategies. This course will involve students in robot development, building, and programming. Students will work hands-on in teams to design, build, program, and document their progress. Topics may include motor control, gear ratios, torque, friction, sensors, program loops, decision-making, and timing sequences.	
Transfer Goals/Big Ideas	
<i>Students will be able to independently use their learning to:</i> • Apply problem-solving strategies, critical thinking, and engineering design process to overcome real-world challenges. • Work effectively in teams, utilizing clear communication and collaborative skills to solve problems, distribute tasks, and refine designs. • Analyze problems and break them down into logical steps applying basic programming concepts to automate tasks or develop creative solutions. • Troubleshoot issues, refine solutions, and iterate on designs independently, demonstrating resilience and creativity when faced with unexpected challenges. • Students will recognize how robotics and automation impact industries and begin to explore career pathways to personal interests in STEM fields. • Effectively represent, encode, and visually organize data using various methods so that they can analyze, communicate, and apply information in different contexts	
Priority Missouri Learning Standards/National Standards	
<u>MO Computer Science Standards</u> • 6-8 Computing Systems ○ Devices 01 ○ Hardware & Software 01 ○ Troubleshooting 01 • 6-8.Algorithms & Programming ○ Algorithms 01 ○ Variables 01 ○ Control 01 ○ Modularity 01, 02 ○ Program Development 01, 02, 03, 04, 05 • 6-8 Impacts of Computing ○ Culture 02 ○ Safety, Law, & Ethics 01	<u>Missouri Science Standards</u> • 6-8 Engineering, Technology, and Application of Science ○ Defining and Delimiting Engineering Problems 1.A.1 ○ Developing Possible Solutions 1.B.1, 2, 3

Unit 1: Algorithms & Programming

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Algorithms & Programming - Algorithms 01 - Variables 01 - Control 01 - Modularity 01, 02 - Program Development 01, 02, 03, 04, 05	<i>Students will understand that...</i> - Algorithms provide a step-by-step approach to solving problems and can be represented using flowcharts and pseudocode. - Well-designed algorithms help break down complex problems into manageable steps. - Control structures, including loops and conditionals, determine the flow of a program and allow for dynamic decision-making. - Nested loops and compound conditionals enable programmers to handle complex logic efficiently.	<i>Students will consider...</i> - How do algorithms help us solve complex problems? - Why are flowcharts and pseudocode useful in designing algorithms? - How can an algorithm be optimized for efficiency? - How do control structures impact the flow of a program? - conditionals, and how can they be avoided? - How do flowcharts and pseudocode aid in problem-solving and program development?
Learning Targets		
<i>Students will...</i> - Demonstrate understanding of what a robot is and how it works to complete a task/solve a problem. - Use block coding to code robotic sensors and motors. - Describe the difference between the sensors and understand what each is used for. - Demonstrate the components of Pair Programming. - Break big problems into smaller steps and figure out how to give clear instructions (like a recipe or a set of directions). - Find mistakes in a set of instructions (an algorithm) and fix them so everything works as intended. - Use smart coding like repeating actions (loops) and "if-then" rules (conditionals) to make your programs complete an intended task. - Understand that using conditionals allows for more complex task completion/solutions.		
Unit Duration:		
Ongoing throughout the semester		

Unit 2: Computing Systems

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Computing Systems - Devices 01 - Hardware & Software 01 - Troubleshooting 01 6-8 Networks & the Internet - Cybersecurity 01, 02 6-8 Impacts of Computing - Culture 02 - Safety, Law & Ethics 01	<i>Students will understand that...</i> - Good design considers user needs, ergonomics, and technological constraints - Designing projects that integrate hardware and software enhances problem-solving and innovation. - Systematic troubleshooting is essential for identifying, diagnosing, and resolving problems in computing devices, components, and software. - Problem-solving skills and persistence are key to resolving technical issues efficiently.	<i>Students will consider...</i> - How can computing device designs be improved to better meet user needs? - How do hardware and software interact to process and exchange data. - What are the key steps in an effective troubleshooting routine?
Learning Targets		
<i>Students will...</i> - Find mistakes in a set of instructions (an algorithm) and fix them so everything works as intended. - Take pieces of code that already exist and change them to make something new—while making sure to give credit to the original creator. - Develop ability to recognize patterns and create effective programs.		
Unit Duration:		
Ongoing throughout the semester		

Unit 3: Troubleshooting

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Algorithms and Programming - Modularity 01 - Program Development 04 6-8 Computing Systems - Troubleshooting 01 6-8 Networks & the Internet - Cybersecurity 01, 02	<i>Students will understand that...</i> - Effective troubleshooting strategies help identify, analyze, and resolve computing problems efficiently. - Protecting digital information requires an understanding of threats, safeguards, and responsible online behavior. - Breaking down complex problems into smaller, reusable modules makes programming more efficient and maintainable. - The iterative development process, including testing and debugging, leads to more effective and reliable software solutions.	<i>Students will consider...</i> - How can systematic troubleshooting techniques help diagnose and fix problems in computing systems? - What are common cybersecurity threats, and how can individuals and organizations protect against them? - How does using modular components improve the efficiency, reusability, and readability of programs? - Why is it important to test and refine a program throughout the development process?
Learning Targets		
<i>Students will...</i> - Identify and fix errors in code using debugging techniques. - Apply problem-solving strategies to troubleshoot hardware and software malfunctions. - Describe and implement basic cybersecurity practices, such as strong passwords and multi-factor authentication. - Break down a complex problem into smaller, reusable code modules or functions. - Follow an iterative development process that includes planning, coding, testing, and refining a program.		
Unit Duration:		
Ongoing throughout the semester		

Unit 4: Program Development

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 6-8 Algorithms & Programming - Algorithms 01 - Variables 01 - Control 01 - Modularity 01, 02 - Program Development 01, 02, 03, 04, 05 6-8 Impacts of Computing - Culture 02 - Safety, Law, & Ethics 01 Missouri Science Standards 6-8 Engineering, Technology, and Application of Science - Defining and Delimiting Engineering Problems 1.A.1 - Developing Possible Solutions 1.B.1, 2, 3	<i>Students will understand that...</i> - Computing technologies have transformed the way people work, communicate, and solve global problems, offering both benefits and challenges. - The ability to work collaboratively in a digital environment is essential for innovation and success in many fields.	<i>Students will consider...</i> - How has computing technology changed the way people live, work, and solve global problems? - How does collaboration improve the development of computational solutions? - What challenges arise when collaborating digitally, and how can they be addressed? - How will you design and program a robot for a specific task?
Learning Targets		
<i>Students will...</i> - Create robots for a specific purpose. - Code robots to complete a specified task. - Apply problem-solving strategies to troubleshoot hardware and software malfunctions. - Use smart coding like repeating actions (loops) and "if-then" rules (conditionals) to make your programs complete an intended task. - Understand that using conditionals allows for more complex task completion/solutions. - Identify and fix errors in code using debugging techniques. - Follow an iterative development process that includes planning, coding, testing, and refining a program. - Evaluate design solutions to determine how well they meet the criteria and constraints of the problem.		
Unit Duration:		
Weeks 9-18		

Learning Plan

Week(s)	Topic	Learning Targets	Resources/Texts	Assessment
Weeks 1-2	Understanding the Sensors & Intro to Block Coding (Algorithms and Programming; Computing Systems; Troubleshooting)	- Students will demonstrate understanding of what a robot is and how it works to complete a task/solve a problem. - Students will use block coding to code a robotic sensor. - Students will describe the difference between the sensors and understand what each is used for. - Students will demonstrate the components of Pair Programming.	- Lego Spike Prime Robotics Kits Pair Programming video Lego Spike Prime Lessons - Lessons 1-7 (Sensors/Motors - light matrix, motor, color sensor, force sensor, gyro sensor) Tutorial videos for Lego sensors and motor	Vocabulary Quiz/Demonstration over sensors
Week 3	Pseudocode and Computational Thinking (Algorithms and Programming; Troubleshooting; Computing Systems)	- Break big problems into smaller steps and figure out how to give clear instructions (like a recipe or a set of directions). - Find mistakes in a set of instructions (an algorithm) and fix them so everything works as intended.	- Lego Spike Prime Robotics Kits Lego Spike Prime Online Coding website (unit -- Kickstart a Business) Pair Programming video Lego Spike Prime Lessons - Lessons 8 & 10	- Flowcharts/Pseudocode graphic organizers Place Your Order project Out of Order project
Week 4	Modifying Code (Algorithm and Programming; Troubleshooting; Computing Systems)	- Take pieces of code that already exist and change them to make something new—while making sure to give credit to the original creator. - Develop ability to recognize patterns and create effective programs.	- Lego Spike Prime Robotics Kits Lego Spike Prime Online Coding website (unit -- Kickstart a Business) Pair Programming video Lego Spike Prime Lessons - Lesson 12	Track Your Packages project
Weeks 5-8	Coding & Troubleshooting (Algorithms and Programming;	Use smart coding like repeating actions (loops) and "if-then" rules (conditionals) to make your programs complete an intended task.	Lego Spike Prime Robotics Kits	- Safe Deposit Box Design and Redesign projects - Lego Keep it Safe

	Troubleshooting; Program Development; Computing Systems)	• Understand that using conditionals allows for more complex task completion/solutions. • Identify and fix errors in code using debugging techniques. • Apply problem-solving strategies to troubleshoot hardware and software malfunctions. • Describe and implement basic cybersecurity practices, such as strong passwords and multi-factor authentication. • Break down a complex problem into smaller, reusable code modules or functions. • Follow an iterative development process that includes planning, coding, testing, and refining a program.	• Lego Spike Prime Online Coding website (unit -- Kickstart a Business) • Pair Programming video • Lego Spike Prime Lessons - Lessons 14, 16, 18	○ Keep it Really Safe • Automated Helper Design project - Lego
Weeks 9 - 18	Coding & Troubleshooting & Robot Design (Algorithms and Programming; Troubleshooting; Program Development; Computing Systems)	• Create robots for a specific purpose. • Code robots to complete a specified task. • Apply problem-solving strategies to troubleshoot hardware and software malfunctions. • Use smart coding like repeating actions (loops) and "if-then" rules (conditionals) to make your programs complete an intended task. • Understand that using conditionals allows for more complex task completion/solutions. • Identify and fix errors in code using debugging techniques. • Follow an iterative development process that includes planning, coding, testing, and refining a program. • Evaluate design solutions to determine how well they meet the criteria and constraints of the problem.	• Lego Spike Prime Robotics Kits • Lego Spike Prime Online Coding website (unit -- Life Hacks) • Pair Programming video • Lego Spike Prime Lessons - Lessons 22, 23, 26, 27, 30, 40 • Code Your Moves - unplugged activity • Break Dance video • Rebuild the World with Dance Technology video • Repeat 5 Times video • Rain or Shine video • Brain Game video • Optional Lego Spike Prime builds • Robot table top and mats	• Code Your Moves - unplugged activity • Break Dance • Rebuild the World with Dance Technology - design a high-tech stage for an unforgettable performance • Repeat 5 times • Rain or Shine • Brain Game • "Competition Builds" based on Lego & student choice builds

Proficiency Scales for Computers Middle School Robotics

Grade Level/Course: Middle School Robotics	Standard(s): • MO Computer Science Standards ○ 6-8 Algorithms & Programming: PD 01, 02, 03, 04, 05, A 01, V 01, C 01, M 01, 02 Domain: Program Development (Planning & Collaboration)
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	• Flowcharts for coding (graphic organizer) • Vocabulary quizzes • Robot builds • Evaluate peers' flowcharts/pseudocode for clarity and correctness • Use feedback to refine/revise code • Group rubrics for projects
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. • Design and program solution ideas that are generated and planned using pseudocode and flowcharts. • Use algorithmic thinking to write specific, clear, and complete directions. • Work effectively as a team and incorporate the strengths of each team member • Use feedback from team members and manage all tasks requirements and timelines.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. • Create algorithms (lists of steps) to complete a task, identify flowchart symbols, and understand the difference between linear and conditional flowchart. • Work effectively as a team member and incorporate the strengths of each member and ask for assistance when there is a conflict.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. • Create algorithms (lists of steps). • Understand coding vocabulary such as: flowchart symbols, pseudocode, syntax, and algorithm concepts. • Identify when teammates are not working effectively and attempt to find a solution.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Middle School Robotics	Standard(s): - MO Computer Science Standards 6-8 Algorithms & Programming: M 01, 02, V 01, A 01, C 01 Domain: Algorithms and Programming
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	Create code for robots using predetermined criteria.
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Create programs. - Create clearly named variables by making their own code blocks to store and manipulate information. - Design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements) - Manage projects tasks and timelines when collaboratively developing computational artifacts.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Create clearly named variables by making their own code blocks to store and manipulate information. - Design and develop combinations of control structures, such as multiple loops (repeat blocks) and compound conditionals (if then statements) - Manage projects tasks and timelines when collaboratively developing computational artifacts. - Understand the meaning of programming vocabulary such as: boolean, conditions, etc.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. - Program scrolling messages and images, using tutorials - Understand cybersecurity vocabulary such as: encryption, case-sensitive, etc.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Middle School Robotics	Standard(s): - MO Computer Science Standards 6-8 Algorithms and Programming: M01, PD 04 6-8 Computing Systems: T01 6-8 Networks & the Internet: C01, 02 Domain: Troubleshooting (Debugging)
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Test a given coded program to determine if the robot completes the intended task. - If the robot does not complete the task correctly, use flowchart/pseudocode to determine where coding went wrong. - Test out robots' coding created by students and troubleshoot as necessary to complete the intended task. - Evaluate peers' flowcharts/pseudocode for clarity and correctness - Use a predetermined code and alter it to make something new
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Use code debugging techniques, such as using code tracing charts and adding helpful program comments, to correct program anomalies. - Analyze, revise, and test a program or device hardware to fix errors.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Identify bugs in their programs and develop solutions to debug programs. - Recognizes patterns in code to troubleshoot issues and create effective programs.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Identify bugs in their programs. - Can follow a troubleshooting checklist	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: Middle School Robotics	Standard(s): - MO Computer Science Standards 6-8 Computing Systems: D01, HS01, T01 6-8 Networks & the Internet: C01, 02 6-8 Impacts of Computing: C02, SLE 01 Domain: Computing Systems (Devices, Hardware, and Software)
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Robot Build and coding for Predetermined task - Example: build a robot to detect a color or avoid obstacles -- choose the best sensors and explain reasoning behind that choice
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Analyzes different hardware components and chooses the best ones to improve the robot's functionality - Builds robot and creates code that includes use of color sensors, pressure and distance sensors, gyroscope, etc. that completes a task. - Sends code to robot via Bluetooth connection from chromebook/computer	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Builds a robot using the step-by-step instructions - Codes the task using step-by-step instructions - Demonstrates understanding of how the robot software and hardware work together to control the robot's movements and actions	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Compares different types of robotics components (including sensors and motors) and describes how they impact a robot's functionality. - Demonstrates understanding of block coding interface	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

PLTW Computer Science Essentials

Course Overview

Grade level(s): 9-12**Credits earned:** 1 Credit**Course Description**

Students will learn the fundamentals of computer programming and build computational-thinking skills, then apply what they know to design solutions such as crowdsourcing apps for mobile devices using MIT App Inventor®. Students will also transfer the understanding of programming gained in App Inventor® to text-based programming in Python® and apply their knowledge to create algorithms for games of chance and strategy. This is the introductory Project Lead the Way (PLTW) course in the Computer Science pathway.

Transfer Goals

Students will be able to independently use their learning to understand...

- How to create apps that rely on the use of variables, data structures, operators, iteration, and abstraction.
- The impact of computer science on society.
- Problem decomposition and iterative design are enhanced using scrum programming.
- The importance of user-centered design and professional conduct.

Priority Missouri Learning Standards/National Standards[MO Computer Science Standards](#)

- 9-10 Algorithms and Programming
 - Algorithms 01
 - Variables 01, 02
 - Control 01
 - Modularity 01
 - Program Development 01, 02, 04, 05
- 9-10 Data Analysis
 - Inference & Models 01
- 9-10 Impacts of Computing
 - Culture 01

- 9-10 Computing Systems
 - Devices 01
 - Troubleshooting 01
- 11-12 Algorithms and Programming
 - Algorithms 02, 03
 - Variables 01
 - Control 01
 - Modularity 02
 - Program Development 01, 03

Unit 1: Electronic Data <i>Desired Results</i>		
Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Algorithms and Programming - Variables 01, 02 11-12 Algorithms and Programming - Variables 01 9-10 Data Analysis - Inference & Models 01	<i>Students will understand...</i> - Data is a tool to inform decision making within a program. - Variables are essential in programming for storing, manipulating, and using data effectively. - Clear, well-structured code helps programmers understand how data flows through a program.	<i>Students will consider...</i> - How is data stored in computer programs? - What kinds of data can be stored? - How are data compared and manipulated in a computer program?
Learning Targets		
<i>Students will...</i> - Create, assign, and retrieve variables. - Appropriately use string, integer, boolean, and float data types. - Create and manipulate single dimension data structures. - Arithmetic and logical operators to compare data. - Use arithmetic operators to perform simple calculations with data.		
Unit Duration:		
9 weeks		

Unit 2: Controlling Program Flow <i>Desired Results</i>		
Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Algorithms and Programming - Algorithms 01 - Control 01 11-12 Algorithms and Programming - Control 01	Students will understand.... - Algorithms are created to solve real-world problems. - Conditionals enhance the control structure and efficiency of a program. - Iteration in programming increases efficiency and readability.	Students will consider... - How are complex instructions converted to simple steps for a computer to complete? - How does a computer program make decisions? - How does a program repeat steps or move through multiple items in a group?
Learning Targets		
Students will. . . - Devise a set of steps to perform a task. - Use if/else statements to make decisions about data. - Execute a repetitive task using for or while loops.		
Unit Duration:		
9 weeks		

Unit 3: Creative Problem Solving

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Computing Systems - Devices 01 9-10 Algorithms and Programming - Modularity 01 - Program Development 01, 04 11-12 Algorithms and Programming - Modularity 02 - Program Development 01, 03	Students will understand... - Abstraction reduces the unnecessary details involved in program implementation. - A design process is necessary to create and evaluate an user-centered application	Students will consider... - How are complicated programs and languages made simple enough for users? - How can a large problem be broken down into manageable steps? - How are complex programs built with repeated design cycles?
Learning Targets		
Students will. . . - Convert a complex computational problem into a series of coherent steps. - Demonstrate understanding of multiple layers of language abstractions between the code they write and the code that is run by the processor. - Write a function to modularize code that will be used repeatedly, and call that function. - Use the design cycle to create, and subsequently improve a program.		
Unit Duration:		
6 weeks		

Unit 4: Collaborative Programming

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Algorithms and Programming - Program Development 02, 03 9-10 Computing Systems - Troubleshooting 01 11-12 Algorithms and Programming - Algorithms 02, 03	Students will understand.... - User-centered design relies on developing and documenting of a solution, testing, and debugging. - Scrum programmings iterative process promotes efficient collaboration, adaptability, and efficiency.	Students will consider... - How should user input be used in the production of software? - How/why would a pair of programmers work on the same program? - How do a group of three or more people work together on the same program?
Learning Targets		
Students will. . . - Build a program based on user needs, and refine the program based on user feedback. - Work in a pair as the driver or navigator. - Fulfill their assigned role in a development scrum and collaborate with the other members to design and write a program.		
Unit Duration:		
6 weeks		

Unit 5: Social Computing “Soft Skills”***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11-12 Algorithms and Programming - Algorithms 03 9-10 Impacts of Computing - Culture 01	Students will understand... - The historical and future impact computer science has on society. - The importance of ethical decision making, responsible use of technology, and respecting intellectual property.	Students will consider... - How does technology shape society and vice versa? - How should computer science professionals conduct themselves in the workplace? - How can computer scientists effectively present their ideas and work to others?
Learning Targets		
Students will. . . - Make predictions about potential effects of new technology on society using multiple lines of reasoning - Understand and exhibit the expectations of modern professional conduct - Present a project exhibiting understanding of the application		
Unit Duration:		
6 weeks		

Learning Plan

Week(s)	Topic	Resources/Texts	Learning Targets	Assessment
Weeks 1-9	Electronic Data	PLTW Website Unit 1 Student Journals MIT App Inventor	- Students will be able to create, assign, and retrieve variables - Students will be able to appropriately use string, integer, boolean, and float data types - Students will be able to create and manipulate single dimension data structures - Students will be able to use arithmetic and logical operators to compare data - Students will be able to use arithmetic operators to perform simple calculations with data	- Create a program that creates variables and performs simple math with them, assigning solutions to new variables. - Add and replace elements in data structures.
Weeks 10-18	Controlling Program Flow	PLTW Website Unit 2 Student Journals Vex Robots Vex V5 Vex VR	- Students will be able to devise a set of steps to perform a task - Students will be able to use if/else statements to make decisions about data - Students will be able to execute a repetitive task using for or while loops	- Students list possible steps to solve a problem - Student follows program flow to give correct print statement
Weeks 19-24	Creative Problem-Solving	PLTW Website Unit 3 Visual Studio Code	- Students will be able to convert a complex computational problem into a series of coherent steps - Students will understand that there are multiple layers of language abstractions between the code they write and the code that is run by the processor	- Given an example user story the student can convert a complex problem into a series of coherent steps, understands the abstraction, write a function that will be used repeatedly - Student gives ideas on what to improve and how to go about it

			- Students will be able to write a function to modularize code that will be used repeatedly, and call that function - Students will be able to use the design cycle to create, and subsequently improve a program	
Weeks 25-30	Collaborative Programming	PLTW Website Unit 4 Visual Studio Code application (program on the desktops in the computer lab)	- Students will be able to build a program based on user needs, and refine the program based on user feedback - Students will be able to work in a pair as the driver or navigator - Students will be able to fulfill their assigned role in a development scrum and collaborate with the other members to design and write a program - Students will be able to make predictions about potential effects of new technology on society using multiple lines of reasoning	- Given a user brief the student can build a program based on user needs - Student can fulfill their assigned role in a scrum
Weeks 31-36	Social Computing “Soft Skills”	PLTW Website units 1-4	- Students will be able to understand and exhibit the expectations of modern professional conduct - Students will be able to present a project exhibiting understanding of the application	- Given a user brief the student can predict potential effects of new technology - Student exhibits professional conduct - Application project and presentation

Proficiency Scales for PLTW Computer Science Essentials

Grade Level/Course: PLTW Computer Science Essentials	Standard(s): • MO Computer Science Standards ○ 9-10 Algorithms and Programming: V01, 02 ○ 11-12 Algorithms and Programming: V01 ○ 9-10 Data Analysis: IM01 Domain: Electronic Data
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	• Create a program that creates variables and performs simple math with them, assigning solutions to new variables. • Add and replace elements in data structures.
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. • Create, assign, and retrieve variables. • Use boolean statements to complete an algorithm. • Manipulate single dimension data structures. • Use arithmetic and logical operators to compare all data types.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. • Create single dimension data structures. • Use string, integer, and float data types. • Use arithmetic and logical operators to compare a data type.. • Use arithmetic operators to perform simple calculations.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. • Create a variable. • Use a string data type. • Identify arithmetic operators.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Essentials	Standard(s): - MO Computer Science Standards 9-10 Algorithms and Programming: A01, C01 11-12 Algorithms and Programming: C01 Domain: Controlling Program Flow
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Students list possible steps to solve a problem - Student follows program flow to give correct print statement
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Devise a set of steps to perform a task. - Use if/else statements to make decisions about data with at least three possible choices. - Execute a repetitive task using "for" and "while" loops.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Devise a set of tasks. - Use if/else statements to make decisions about data with at least two possible choices. - Execute a repetitive task with either a for or while loop.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Describe the purpose of a task - Use an if/then statement.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Essentials	Standard(s): - MO Computer Science Standards 9-10 Computing Systems: D01 9-10 Algorithms and Programming: M01; PD 01,04 11-12 Algorithms and Programming: M02; PD 01, 03 Domain: Creative Problem Solving
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Given an example user story the student can convert a complex problem into a series of coherent steps, understands the abstraction, write a function that will be used repeatedly - Student gives ideas on what to improve and how to go about it
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Convert a complex computational problem into a series of coherent steps. - Demonstrates understanding that there are multiple layers of language abstraction between the code they write and the code that is run by the processor. - Write a function to modularize code that will be used repeatedly, and call that function. - Use the design cycle to create, and subsequently improve a program.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Convert parts of a problem into coherent steps. - Understands that the processor does not run the exact code written. - Understands that code should be modularized rather than rewritten. - Starts any program with a simple form that can be upgraded.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. Understands the benefit of decomposing a problem.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Essentials	Standard(s): - MO Computer Science Standards 9-10 Algorithms and Programming: PD 02, 05 9-10 Computing Systems: T01 11-12 Algorithms and Programming: A02, 03 Domain: Collaborative Programming
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Given a user brief the student can build a program based on user needs - Student can fulfill their assigned role in a scrum
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Build a program based on user needs, and refine the program based on user feedback. - Complete a program in a pair as the driver or navigator. - Fulfill their assigned role in a development scrum and collaborate with the other members to design and write a program.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Build a program based on given user needs. - Begin building a program with a partner. - Complete an assigned role as a member of a scrum team.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Build a program based on defined criteria.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Essentials	Standard(s): - MO Computer Science Standards 9-10 Impacts of Computing: C01 11-12 Algorithms and Programming: A03 Domain: Social Computing “Soft Skills”
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Given a user brief the student can predict potential effects of new technology - Student exhibits professional conduct - Application project and presentation
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Make predictions about potential effects of new technology on society using multiple lines of reasoning. - Demonstrates the expectations of modern professional conduct. - Presents a project exhibiting understanding of the application and the appropriate terminology.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Describe the impact that technology has on society. - Identify the expectations of modern professional conduct. . - Present a project exhibiting understanding of the application.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. - Verbally describe the purpose of an application. - Identify how technology is used in society.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

PLTW Cybersecurity

Course Overview

Grade level(s): 9-12

Credits earned: 1

Course Description

Students in Cybersecurity establish an ethical code of conduct while learning to defend data in today's complex cyberworld. Cybersecurity introduces the tools and concepts of cybersecurity and encourages students to create solutions that allow people to share computing resources while protecting privacy. Nationally, computational resources are vulnerable and frequently attacked; in Cybersecurity, students solve problems by understanding and closing these vulnerabilities. This course raises students' knowledge of and commitment to ethical computing behavior. It also aims to develop students' skills as consumers, friends, citizens, and employees who can effectively contribute to communities with a dependable cyber-infrastructure that moves and processes information safely.

Transfer Goals

Students will be able to independently use their learning to

- Assess security needs of a system and protect a system from possible threats.
- Collaborate as a cyber team to analyze network traffic and implement protection measures to secure the network.
- Recognize patterns of a cybersecurity event, develop a plan for remediation, and document all evidence and procedures of remediation.

Priority Missouri Learning Standards/National Standards

MO Computer Science Standards

- 9-10 Networks and the Internet
 - Network Communication & Organization 01, 02
 - Cybersecurity 01, 02, 03, 04
- 9-10 Impacts of Computing
 - Safety, Laws & Ethics 02, 05
- 9-10 Algorithms and Programming
 - Modularity 02
- 9-10 Data Analysis
 - Storage 02

- 11-12 Networks and the Internet
 - Network, Communication & Organization 01, 02, 03
 - Cybersecurity 01, 02
- 11-12 Algorithms and Programming
 - Program Development 06

Unit 1: Personal Security***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Networks and the Internet - Cybersecurity 02, 03 9-10 Impacts of Computing - Safety, Laws, & Ethics 02 11-12 Algorithms and Programming - Program Development 06	<i>Students will understand...</i> - The importance of personal and digital security. - Internet structure and security. - Managing files and processes.	<i>Students will consider...</i> - Why does information need protection? - How do computers safely store information? - How can malware be stopped?
Learning Targets		
<i>Students will...</i> - Identify insecure data sharing practices and unsafe practices related to social media and email. - Protect against spam and phishing. - Evaluate suspicious emails or websites. - Configure a firewall to protect against malware. - Access, manipulate, and encrypt files - Manage, recognize and terminate suspicious processes. - Examine and manage browser configurations and site certificates.		
Unit Duration:		
8 weeks		

Unit 2: System Security***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Networks and the Internet - Network Communication & Organization 01, 02 9-10 Impacts of Computing - Safety, Law, & Ethics 05 11-12 Networks and the Internet - Network Communication & Organization 02, 03	<i>Students will understand...</i> - Security vulnerabilities of web services. - Host names relate to digital addresses. - How information can be safely and securely exchanged on a public network.	<i>Students will consider...</i> - How do computers safely store information? - What makes a network vulnerable? - How can we minimize network vulnerabilities?
Learning Targets		
<i>Students will...</i> - Identify website ownership and observe website traffic. - Identify ports and their services. - Create and describe a network topology. - Describe malware types and motivations for use. - Investigate a web server for directory browsing and log file location vulnerabilities. - Mitigate web server vulnerabilities including directory browsing, default log file location, and anonymous FTP. - Create a report based on a cybersecurity threat and mitigation. - Recognize and mitigate cross-site scripting exploits and SQL injection exploits. - Monitor and analyze data packets.		
Unit Duration:		
9 weeks		

Unit 3: Network Security***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Networks and the Internet - Cybersecurity 01, 04 9-10 Data Analysis - Storage 02 11-12 Networks and the Internet - Network Communication & Organization 01 11-12 Algorithms and Programming - Program Development 06	<i>Students will understand...</i> - How an operating system organizes information to manage and secure digital information. - System processes and how malware spreads around a network. - Patterns that represent exploits. - How to secure a network against attacks.	<i>Students will consider...</i> - How does past knowledge help with data analysis? - How can information be safely exchanged? - What makes a network vulnerable?
Learning Targets		
<i>Students will...</i> - Use commands to navigate the file system and to manage files and directories. - Use ownership and permissions settings to analyze files. - Use file hiding and encryption. - Manage and examine processes and services in a Linux based system. - Identify suspicious processes. - Create a plan to identify and remediate suspicious activity. - Create a baseline of network traffic. - Capture and analyze suspicious network traffic and configure firewall rules to improve security.		
Unit Duration:		
11 weeks		

Unit 4: Applied Cybersecurity

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 9-10 Networks and the Internet - Cybersecurity 03 9-10 Algorithms and Programming - Modularity 02 11-12 Networks and the Internet - Cybersecurity 01, 02	<i>Students will understand...</i> - Encryption, ciphers, and frequency predictors. - Data hiding using cryptography and steganography. - The process of gathering digital evidence and analysis. - The value of digital evidence and the authority to search.	<i>Students will consider...</i> - Why do people engage in risky behavior in cyberspace? - What are the consequences of inappropriate behavior in cyberspace? - Where and how are cybersecurity skills used?
Learning Targets		
<i>Students will...</i> - Identify types of encryption. - Validate who owns information. - Use paired key encryption. - Test the security of various cryptography methods. - Recognize different storage containers of digital evidence. - Collect digital evidence. - Validate and authenticate data using hashing. - Create a forensic image. - Identify artifacts in digital forensics. - Process a forensic image with a forensic tool suite. - Use forensic knowledge to investigate a computer. - Prepare a report on investigative findings.		
Unit Duration:		
8 weeks		

Learning Plan

Week(s)	Topic	Resources/Texts	Learning Targets	Assessment
1-8	Personal Security	PLTW Website Unit 1 PLTW Security Lab 1.1.2 PLTW Security Lab 1.1.3 PLTW Security Lab 1.2.1 PLTW Security Lab 1.2.2 PLTW Security Lab 1.2.3 PLTW Security Lab 1.2.5 PLTW Security Lab 1.3.1 Student journals	- Be able to identify insecure data sharing practices and unsafe practices related to social media and email. - Be able to protect against spam and phishing. - Be able to evaluate suspicious emails or websites. - Be able to configure a firewall to protect against malware. - Be able to access, manipulate, and encrypt files - Be able to manage, recognize and terminate suspicious processes. - Be able to examine and manage browser configurations and site certificates.	- Investigate a personal computer hack attack. - Create a plan of action to detect and identify suspicious activity on a computer, how the threat will be resolved, recover missing data, and how to protect from future events.
9-17	System Security	PLTW Website Unit 2 PLTW Security Lab 2.1.1 PLTW Security Lab 2.1.2 PLTW Security Lab 2.2.2 PLTW Security Lab 2.2.3 PLTW Security Lab 2.2.4 PLTW Security Lab 2.3.1 PLTW Security Lab 2.3.2 PLTW Security Lab 2.3.3 PLTW Security Lab 2.3.4 PLTW Security Lab 2.4.1 Student Journals	- Be able to Identify website ownership and observe website traffic. - Be able to identify ports and their services. - Be able to create and describe a network topology. - Be able to describe malware malware types and motivations for use. - Be able to investigate a web server for directory browsing and log file location vulnerabilities. - Be able to mitigate web server vulnerabilities including directory browsing, default log file location, and anonymous FTP. - Be able to create a report based on a cybersecurity threat and mitigation. - Be able to recognize and mitigate cross-site scripting exploits and SQL injection exploits - Be able to monitor and analyze data packets.	- Identify, Detect, Protect, and Respond to securing an e-commerce website. - Create and implement a plan of action to identify the status of server vulnerabilities, test exploits on web pages using pen test site, and give recommendations to secure vulnerabilities.

18-28	Network Security	PLTW Website Unit 3 PLTW Security Lab 3.1.1 PLTW Security Lab 3.1.2 PLTW Security Lab 3.1.3 PLTW Security Lab 3.1.4 PLTW Security Lab 3.2.2 PLTW Security Lab 3.2.3 PLTW Security Lab 3.2.4 PLTW Security Lab 3.2.5 PLTW Security Lab 3.2.6 PLTW Security Lab 3.3.1 PLTW Security Lab 3.3.2 PLTW Security Lab 3.3.5 PLTW Security Lab 3.4.1 Student Journals	• Be able to use commands to navigate the file system and to manage files and directories. • Be able to use ownership and permissions settings to analyze files. • Be able to use file hiding and encryption. • Be able to manage and examine processes and services in a Linux based system. • Be able to identify suspicious processes. • Be able to create a plan to identify and remediate suspicious activity. • Be able to create a baseline of network traffic. • Be able to capture and analyze suspicious network traffic and configure firewall rules to improve security.	• Respond to an attack by identifying an exploit, securing the system and making improvements to prohibit future attacks. • Create a security update report that compares normal and malicious network traffic, identifies and mitigates the attack, performs pen tests on the affected network, and thoroughly describes actions that were taken to eliminate the attack.
29-36	Applied Cybersecurity	PLTW Website Unit 4 PLTW Security Lab 4.1.2 PLTW Security Lab 4.1.3 PLTW Security Lab 4.1.4 PLTW Security Lab 4.1.5 PLTW Security Lab 4.2.2 PLTW Security Lab 4.2.4 PLTW Security Lab 4.2.5 PLTW Security Lab 4.2.6 PLTW Security Lab 4.3.1	• Be able to identify types of encryption. • Be able to validate who owns information. • Be able to use paired key encryption. • Be able to test the security of various cryptography methods. • Be able to recognize different storage containers of digital evidence. • Be able to collect digital evidence. • Be able to validate and authenticate data using hashing. • Be able to create a forensic image. • Be able to identify artifacts in digital forensics. • Be able to process a forensic image with a forensic tool suite. • Be able to use forensic knowledge to investigate a computer. • Be able to prepare a report on investigative findings.	• Investigate a crime and provide digital evidence to solve it. • Document and analyze each step of a forensic investigation to solve a crime and prepare a report of the findings.

Proficiency Scales for Cybersecurity

Grade Level/Course: PLTW Cybersecurity	Standard(s): ● Missouri Computer Science Standards ○ 9-10 Networks and the Internet: C02, 03 ○ 9-10 Impacts of Computing: SLE 02 ○ 11-12 Algorithms and Programming: PD 06 Domain: Personal Security
---	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	● Create a plan of action to detect and identify suspicious activity on a computer, how the threat will be resolved, recover missing data, and how to protect from future events. ● Implement the plan of action. ● Identify and explain if an email is suspicious, demonstrate steps to protect against spam and phishing.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. ● Create a presentation on how to identify suspicious emails or websites. ● Configure a firewall to protect against malware. ● Manage, recognize and terminate suspicious processes. ● Examine and manage browser configurations and site certificates.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. ● Evaluate suspicious emails or websites. ● Identify insecure data sharing practices and unsafe practices related to social media and email. ● Protect against spam and phishing.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. ● Understands that emails and websites may present vulnerability risks. ● Understands that firewalls and browser configurations can be used to protect their personal security.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Cybersecurity	Standard(s): - Missouri Computer Science Standards 9-10 Networks and the Internet: NCO 02, 03 9-10 Impacts of Computing: SLE 05 11-12 Networks and the Internet: NCO 02, 03 Domain: System Security
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Identify, Detect, Protect, and Respond to securing an e-commerce website. - Create and implement a plan of action to identify the status of server vulnerabilities, test exploits on web pages using pen test site, and give recommendations to secure vulnerabilities.
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Close ports and their services to prevent exploits. - Create network topology based on a given scenario. - Recognize and mitigate cross-site scripting exploits and SQL injection exploits - Monitor and analyze data packets.	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Create network topology when given a list of devices. - Determine what can prevent malware. - Identify website ownership and observe website traffic. - Identify ports and their services.	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Describe malware types and motivations for use. - Describe a network topology.	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Cybersecurity	Standard(s): - Missouri Computer Science Standards 9-10 Networks and the Internet: C 01, 04 9-10 Data Analysis: S02 11-12 Networks and the Internet: NCO 01 11-12 Algorithms and Programming: PD 06 Domain: Network Security
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Respond to an attack by identifying an exploit, securing the system and making improvements to prohibit future attacks. - Create a security update report that compares normal and malicious network traffic, identifies and mitigates the attack, performs pen tests on the affected network, and thoroughly describes actions that were taken to eliminate the attack.
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Use commands to navigate the file system and to manage files and directories. - Manage and examine processes and services in a Linux based system. - Create a plan to identify and remediate suspicious activity. - Capture and analyze suspicious network traffic and configure firewall rules to improve security.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Identify suspicious processes. - Create and compare baseline network traffic to suspicious traffic. - Analyze suspicious network traffic and configure firewall rules to improve security.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. - Create a baseline of network traffic. - Identifies that firewall rules can improve security. - Compare and contrast suspicious processes and normal processes.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Cybersecurity	Standard(s): - Missouri Computer Science Standards 9-10 Networks and the Internet: C 03 9-10 Algorithms and Programming: M 02 11-12 Networks and the Internet: C 01, 02 Domain: Applied Cybersecurity
---	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Document and analyze each step of a forensic investigation to solve a crime and prepare a report of the findings. - Identify where to look for digital evidence and explain suspicious activity from a given set of data.
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Identify types of encryption. - Use forensic knowledge to investigate a computer. - Prepare a report on investigative findings.	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Test the security of various cryptography methods. - Collect digital evidence and trace possible cause/effect relationships of the artifacts.	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. - Identify artifacts in digital forensics. - Understands why encryption is used.	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

PLTW AP Computer Science Principles (Advanced Placement)	
Grade level(s): 10-12	Credits earned: 1
Course Description	
Students explore and become inspired by career paths that utilize computing, discover tools that foster creativity and collaboration, and use what they've learned to tackle challenges. Students create apps, automate tasks, find patterns in data, and interpret simulations. Students collaborate to create and present solutions that can improve people's lives. Students will have the option to take the AP exam at the end of the course to gain college credit.	
Transfer Goals	
<i>Students will be able to independently use their learning to understand...</i> • Creatively design solutions to problems by deconstructing them into smaller, more manageable issues. • Extrapolate real world information into a data representation, then store and access that information in a computer system. • Provide computational instructions for a computer to complete a task with a variety of data input. • Illustrate how the internet is a group of systems that are all interacting by using a shared procedure to transfer data • Present how computational solutions can be massively impactful to our societies by allowing for previously impossible solutions	
Priority Missouri Learning Standards/National Standards	
MO Computer Science Standards • 9-10 Impacts of Computing ◦ Culture 01, 03 ◦ Social Interactions 01 Advanced Placement Standards • Creative Development 1, 2, 3, 4 • Data 1, 2, 3, 4 • Algorithms and Programming 1, 2, 3, 4, 5, 6, 7, 8, 9,10, 11, 12, 13, 14, 15, 16, 17, 18 • Computer Systems and Networks 1, 2, 3 • Impact of Computing 1, 2, 3, 4, 5, 6	MO Computer Science Standards • 11-12 Algorithms and Programming ◦ Algorithms 01, 04 ◦ Control 01 ◦ Modularity 01, 02, 03 ◦ Program Development 01, 02, 03, 05, 07 ◦ Variables 01 • 11-12 Data Analysis ◦ Collection, Visualization & Transformation 01 ◦ Inference & Models 01 ◦ Storage 01 • 11-12.Impacts of Computing ◦ Culture 01 ◦ Safety, Law, & Ethics 01 • 11-12.Networks and the Internet ◦ Cybersecurity 01 ◦ Network Communication & Organization 01, 02, 03

Unit 1: Creative Design <i>Desired Results</i>		
Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11-12 Algorithms and Programming - Modularity 03 11-12 Data Analysis - Collection, Visualization & Transformation 01 - Inference & Models 01 11-12. Impacts of Computing - Culture 01 AP Standards - Creative Development 1, 2, 3, 4 - Algorithms and Programming 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15	<i>Students will understand that...</i> - Creative development is essential in creating computational artifacts. - Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem. - Computing can extend traditional forms of human expression and experience.	<i>Students will consider...</i> - How can a creative development process affect the creation of computational artifacts? - How can computing and the use of computational tools foster creative expression? - How can computing extend traditional forms of human expression and experience?
Learning Targets		
<i>Students will...</i> - Creative development process to create computational artifacts. - Create a computational artifact for creative expression. - Create a computational artifact using computing tools and techniques to solve a problem. - Collaborate in the creation of a computational artifact. - Analyze the correctness, usability, functionality, and suitability of computational artifacts.		
Unit Duration:		
4 weeks		

Unit 2: Data

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11-12 Data Analysis - Collection, Visualization & Transformation 01 11-12.Impacts of Computing - Culture 01 - Safety, Law, & Ethics 01 AP Standards - Data 1, 2, 3, 4	<i>Students will understand that...</i> - People use computer programs to process information and gain insight and knowledge. - Computing facilitates exploration and the discovery of connections of information. - There are trade-offs when representing information as digital data.	<i>Students will consider...</i> - How can computation be employed to help people process data and information to gain insight and knowledge? - How can computation be employed to facilitate exploration and discovery when working with data? - What considerations and trade-offs arise in the computational manipulation of data? - What opportunities do large data sets provide for solving problems and creating knowledge?
Learning Targets		
<i>Students will...</i> - Explain how binary sequences are used to represent digital data. - Find patterns and test hypotheses about digitally processed information to gain insight and knowledge. - Collaborate when processing information to gain insight and knowledge. - Explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notations, and precise language. - Extract information from data to discover and explain connections or trends. - Determine how large data sets impact the use of computational processes to discover information and knowledge. - Analyze how data representation and storage of data involve computational manipulation of information.		
Unit Duration:		
4 weeks		

Unit 3: Programming and Algorithms

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11-12 Algorithms and Programming - Variables 01 - Algorithms 01, 04 - Control 01 - Modularity 01, 02, 03 - Program Development 01, 02, 03, 05, 07 11-12 Data Analysis - Storage 01 AP Standards - Algorithms and Programming 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 17	<i>Students will understand that...</i> - Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society). - People write programs to execute algorithms. - Programming is facilitated by appropriate abstractions. - Programs are developed, maintained, and used by people for different purposes. - Programming uses mathematical and logical concepts. - Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages. - Algorithms can solve many, but not all, computational problems. - Multiple levels of abstraction are used to write programs or create other computational artifacts. - Models and simulations use abstraction to generate new understandings and knowledge.	<i>Students will consider...</i> - How are programs developed to help people, organizations, or society solve problems? - How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge? - How do computer programs implement algorithms? - How does abstraction make the development of computer programs possible? - How do people develop and test computer programs? - Which mathematical and logical concepts are fundamental to computer programming? - How are algorithms implemented and executed on computers and computational devices? - Why are some languages better than others when used to implement algorithms? - What kind of problems are easy, what kinds are difficult, and what kinds are impossible to solve algorithmically? - How are algorithms evaluated? - How can vastly different kinds of data, physical phenomena, and mathematical concepts be represented on a computer? - How does abstraction help us in writing programs, creating computational artifacts, and solving problems? - How can computational models and simulations help generate new understanding and knowledge?
Learning Targets		
<i>Students will...</i> - Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge. - Develop a correct program to solve problems. - Collaborate to develop a program. - Explain how programs implement algorithms. - Use abstraction to manage complexity in programs. - Evaluate the correctness of a program. - Employ appropriate mathematical and logical concepts in programming. - Develop an algorithm for implementation in a program.		- Express an algorithm in a language. - Explain the difference between algorithms that run in a reasonable time and those that do not run in a reasonable time. - Explain the difference between solvable and unsolvable problems in computer science. - Evaluate algorithms analytically and empirically for efficiency, correctness, and clarity. - Describe the variety of abstractions used to represent data. - Develop an abstraction when writing a program or creating other computational artifacts. - Use multiple levels of abstraction to write programs. - Use models and simulations to formulate, refine, and test hypotheses.
Unit Duration:		
16 weeks		

Unit 4: The Internet

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11-12 Networks and the Internet - Cybersecurity 01 - Network Communication & Organization 01, 02, 03 AP Standards - Computer Systems and Networks 1, 2, 3 - Algorithms and Programming 11 - Impact of Computing 6	<i>Students will understand that...</i> - The Internet is a network of autonomous systems. - Characteristics of the Internet influence the systems built on it. - Cybersecurity is an important concern for the Internet and the systems built on it.	<i>Students will consider...</i> - What is the Internet? How is it built? How does it function? - What aspects of the Internet's design and development have helped it scale and flourish? - How is cybersecurity impacting the ever-increasing number of Internet users?
Learning Targets		
<i>Students will...</i> - Explain the abstractions in the Internet and how the Internet functions. - Explain the characteristics of the Internet and the systems built on it. - Explain how the characteristics of the Internet influence the systems built on it. - Identify existing cyber-security concerns and potential options to address these issues with the internet and the systems built on it.		
Unit Duration:		
3 weeks		

Unit 5: Impact of Computing

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11-12 Impacts of Computing - Culture 01 - Safety, Law & Ethics 01 AP Standards - Impact of Computing 1, 2, 3, 4, 5 - Algorithms and Programming 16, 17, 18	<i>Students will understand that...</i> - Computing enhances communication, interaction, and cognition. - Computing enables innovation in nearly every field. - Computing has global effects - both beneficial and harmful - on people and society. - Computing innovations influence and are influenced by the economic, social, and cultural contexts in which they are designed and used.	<i>Students will consider...</i> - How does computing enhance human communication, interaction, and cognition? - How does computing enable innovation? - What are some potential beneficial and harmful effects of computing? - How do economic, social, and cultural contexts influence innovation and the use of computing?
Learning Targets		
<i>Students will...</i> - Explain how computing innovations affect communication, interaction, and cognition. - Explain how people participate in a problem-solving process that scales. - Explain how computing has impacted innovations in other fields. - Analyze the beneficial and harmful effects of computing. - Explain the connections between computing and real-world contexts, including economic, social, and cultural contexts. - Access, manage, and attribute information using effective strategies. - Evaluate online and print sources for appropriateness and credibility.		
Unit Duration:		
3 weeks		

<i>Learning Plan</i>				
Week(s)	Topic	Resources/Texts	Learning Targets	Assessment
Weeks 1-4	Creativity	• PLTW Inking textbook • Visual Studio Code Integrated Development Environment • Python Programming Language and its documentation • Python Math Module and its documentation • Python Turtle Module and its documentation	• Students will apply a creative development process to create computational artifacts. • Students will create a computational artifact for creative expression. • Students will create a computational artifact using computing tools and techniques to solve a problem. • Students will collaborate in the creation of a computational artifact. • Students will analyze the correctness, usability, functionality, and suitability of computational artifacts.	• Create a program that incorporates user interaction • In the program, user input fundamentally affects what happens; program has an objective with degrees of progress; and program must include different states
Weeks 5-9	Abstraction	• PLTW Inking textbook • Visual Studio Code Integrated Development Environment • Python Programming Language and its documentation • Python Math Module and its documentation • Python Turtle Module and its documentation	• Students will describe the variety of abstractions used to represent data. • Students will explain how binary sequences are used to represent digital data. • Students will develop an abstraction when writing a program or creating other computational artifacts. • Students will use multiple levels of abstraction to write programs. • Students will use models and simulations to formulate, refine, and test hypotheses.	• Create a product for a “client” that is a simple app or program that will be developed into a product. • Use diagrams, lists, and flowcharts to illustrate levels of abstraction for the product. • Create a product backlog, user stories, and sprint task list and will consider the use of a compiler and/or interpreter to translate the program for other operating systems to meet client’s needs. • Coding Puzzles

Weeks 10-13	Data and Information	• PLTW Inking textbook • Visual Studio Code Integrated Development Environment • Python Programming Language and its documentation • Python Math Module and its documentation • Python Pandas Module and its documentation • Python Numpy Module and its documentation	• Students will find patterns and test hypotheses about digitally processed information to gain insight and knowledge. • Students will collaborate when processing information to gain insight and knowledge. • Students will explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notations, and precise language.	• Use a provided set of data to extract information from data to discover and explain connections and trends. • Determine how large data sets impact the use of computational processes to discover information and knowledge. • Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information. • Coding Puzzles • “Plotting data in Python” Project
Weeks 14-20	Algorithms	• PLTW Inking textbook • Visual Studio Code Integrated Development Environment • Python Programming Language and its documentation • Python Math Module and its documentation	• Students will develop an algorithm for implementation in a program. • Students will express an algorithm in a language. • Students will explain the difference between algorithms that run in a reasonable time and those that do not run in a reasonable time. • Students will explain the difference between solvable and unsolvable problems in computer science. • Students will evaluate algorithms analytically and empirically for efficiency, correctness, and clarity.	• Write computer programs after first creating an algorithm to handle the program logic and that include: general algorithm strategies; different data types, variables, and functions; conditionals and branching; nested branching; and strings, tuples, and lists; for and while loops • Compete against classmates in the Iterative Prisoner’s Dilemma • Evaluate key algorithms of a program • Coding Puzzles • Code Warm-ups
Weeks 21-28	Programming	• PLTW Inking textbook • Visual Studio Code Integrated Development Environment • Python Programming Language and its documentation	• Students will develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge. • Students will develop a correct program to solve problems. • Students will collaborate to develop a program. • Students will explain how programs implement algorithms. • Students will use abstraction to manage complexity in programs. • Students will evaluate the correctness of a program.	• Be provided with a set list of client scenarios, will select one client, and will: develop a program to satisfy the client’s needs and solve their problem; and evaluate the correctness of the program. • Employ mathematical and logical concepts in their programming. • Coding Puzzles • “Storytime” Project • “Binary Converter” Project

			Students will employ appropriate mathematical and logical concepts in programming.	
Weeks 29-31	The Internet	PLTW Inkling textbook	- Students will explain the abstractions in the Internet and how the Internet functions. - Students will explain the characteristics of the Internet and the systems built on it. - Students will explain how the characteristics of the Internet influence the systems built on it. - Students will identify existing cyber-security concerns and potential options to address these issues with the internet and the systems built on it.	- Complete activities including: explore and analyze the relationships between routers, switches, servers, topology and addressing; and describe key protocols and underlying processes of internet-based services (e.g., http/https and Simple Mail Transfer Protocol (SMTP)/internet Message Access Protocol (IMAP), routing protocols - Explain how the characteristics of the Internet influence the systems developed on it - Complete a cryptography and cipher project that will detail how secure protocols and key encryption impact cybersecurity "Password Analysis" Project - Coding Puzzles
Weeks 32-36	Global Impact	- PLTW Inkling textbook NetLogo Simulations	- Students will explain how computing innovations affect communication, interaction, and cognition. - Students will explain how people participate in a problem-solving process that scales. - Students will explain how computing has impacted innovations in other fields. - Students will analyze the beneficial and harmful effects of computing. - Students will explain the connections between computing and real-world contexts, including economic, social, and cultural contexts. - Students will access, manage, and attribute information using effective strategies. - Students will evaluate online and print sources for appropriateness and credibility.	- Read selections from <i>Blown to Bits</i>, and will complete a summary/report of the selection - After examining communications technology and discussing ways in which computing affects our ability to build community, students will successfully debate their assigned side of a topic about issues of free speech and how different countries have different standards for what ought to be censored. - Write and present a report about a field of their choice in which computing has impacted innovation - Coding Puzzles

Proficiency Scales for AP Computer Science Principles

Grade Level/Course: PLTW Computer Science Principles (AP)	Standard(s): • MO Computer Science Standards ○ 11-12 Data Analysis: CVT 01; IM 01 ○ 11-12 Algorithms and Programming: M 03 ○ 11-12 Impacts of Computing: C 01 • AP Standards ○ CRD 1, 2 Domain: Creative Development
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	• Written reflections on programming projects • Project check ins with the instructor • Group benchmark check ins with team members • Comparison between project requirements and actual produced project • Creation of an artifact of original design and component
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. • Respond to project requirements and produce a program for creative expression, satisfy a personal curiosity or to create new knowledge • Develop a correct program to solve a presented problem either • Implement a previously designed algorithm in code successfully • Streamline program complexity by implementing thoughtful abstraction tools • Adjust their program design to meet requirement changes and limitations through their development process • Evaluate a program and determine how closely a program meets requirements, as well as troubleshoot errors, so the program is functional	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. • Collaborate with others to develop a program, contributing solutions in a constructive manner • Articulate how an algorithm can be developed in words • Respond to project requirements and produce a program design in collaboration with a peer • Add or modify their code to reflect feedback and suggestions on how to align their program design to meet requirement changes and limitations • Evaluate a program and determine if design elements are missing,, incorrect or non functional with respect to the program design	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. • Describe project requirements • Correct issues with a program design that are identified with direct feedback • Evaluate a program and determine if design elements are missing	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Principles (AP)	Standard(s): - MO Computer Science Standards 11-12 Data Analysis: CVT 01 11-12 Impacts of Computing: C 01, SLE 01 - AP Standards - DAT 1, 2 Domain: Data
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Student can derive the relationship between light intensity and distance using a light detector - Student can analyze publicly sourced data to track its change over time or under given conditions
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Generate and process data from a data collection device to make an informed decision regarding an issue - Student can process data and present it in a graphical manner to support a claim - Student can determine if a situation calls for data compression and what the costs of it would be	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Process a data set programmatically and use the results to support a conclusion - Filter and clean data using computational tools - Describe how data compression works, including its benefits and drawbacks	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. - Describe how data can be collected and processed - Describe how the size of a data set can be difficult to process	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Principles (AP)	Standard(s): - MO Computer Science Standards 11-12 Algorithms and Programming: A 01, 04; V 01; C01; M 01, 02, 03; PD 01, 02, 03, 05, 07 - AP Standards - AAP 1, 2, 3, 4 Domain: Algorithms & Programming	
Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Student can generate an image using the turtles library - Student can solve coding puzzles/warm ups - Student can create their own "Make your own adventure" game
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Given project requirements, the student can generate a program that may include: - Logical decision making - Various Data Types - Storage data structures - Iteration - Components, such as functions or objects, that can be custom made or from a library - Evaluate the time complexity of an algorithm and then offer code suggestions to improve said time complexity - Evaluate the code of another programmer, offering constructive criticisms of their work	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Given project requirements, the student can generate a program that includes: - Variable assignments - Iteration - Randomization - Calling procedures - Understands that different types of algorithms have differing time complexities - Describe the benefits and drawbacks of simulations - Discuss and describe example undecided problems - Evaluate their code in a review with another programmer	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Code a sequence of steps that allows a program to utilize defined variables in mathematical operations - Describe how iteration and data structures can be used - Describe what time and space complexity are	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Principles (AP)	Standard(s): - MO Computer Science Standards 11-12 Networks and the Internet: NCO 01, 02, 03; C 01 - AP Standards - CSN 1, 2 Domain: The Internet
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Given a diagram, the student can show how fault-tolerance has or has not been achieved - The student can design a fictional system to better solve a previously unsolvable problem using a computer network
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Describe how multiple machines in communication can achieve a task that single machines cannot - Describe how data is transferred within a network	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Explain how a network can suffer a node or series of node failures and still be in operation - Evaluate the efficiency of a parallel model versus a sequential model to decide which one is more efficient	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Describe the common protocols used to transfer data within a network and how the data is transferred - Explain how redundancy can lead to fault tolerance - Describe the difference between sequential and parallel computing	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: PLTW Computer Science Principles (AP)	Standard(s): - MO Computer Science Standards 11-12 Impacts of Computing: C01, SLE01 - AP Standards - IPOC 1, 2, 3, 4, 5 - AP 16, 17, 18 Domain: Impact of Computing
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks ("How can I assess this standard?")
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the "on target" success criteria.	- Given a scenario, the student can provided a cohesive argument as to who owns a produced program - Given a scenario, the student can offer tangible solutions that would close the digital divide - The student can design a citizen science experiment, noting how computer systems can allow for its execution
3 On Target	To achieve a 3, the "on target" (proficient) student meets all success criteria for the priority standard listed. - Describe why it is important for a responsible programmer needs to take care in their design and create safe-guard limitations - Debate the ownership of a program given the facts surrounding its creation	
2 In Progress	To achieve a 2, the "in progress" (basic) student meets all the success criteria below. - Given a scenario, the student can explain how a program might benefit or harm a community depending on its purpose and design - Offer possible solutions to help narrow it - Describe how crowdsourcing can support citizen science and enhance problem solving	
1 Getting Started	To achieve a 1, the "getting started" (below basic) meets all the success criteria below to show they are "getting started" toward the target goal. - Student can describe the digital divide - Explain how computing bias can occur - Describe how crowdsourcing works	
LND	The student's current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

PLTW Computer Science A (Advanced Placement)

Grade level(s): 11, 12	Credits earned: 1.0
Course Description	
This course immerses students in object-oriented programming using Java, emphasizing algorithm development, data structures, and problem-solving. Students will learn to design, implement, and analyze solutions to problems, preparing them for the AP Computer Science A exam and potential college credit. Through hands-on projects and collaborative work, students will develop computational thinking skills applicable to various fields.	
Transfer Goals	
<i>Students will be able to independently use their learning to...</i> • Design and implement modular solutions using object-oriented principles. • Analyze and evaluate processes for efficiency and clarity. • Utilize data structures such as arrays and ArrayLists to manage and process data. • Apply logical thinking to simplify, and solve complex problems. • Collaborate effectively to develop, test, and debug programs.	
Priority Missouri Learning Standards/National Standards	
Advanced Placement Standards • Modularity 1, 2, 5, 9 • Variables 1, 2, 5, 6, 7, 8 • Control 1, 2, 3, 4, 6, 7, 8, 10 • Impact of Computing 5, 7	MO Computer Science Standards • 11-12 Algorithms and Programming ○ Algorithms 01, 04 ○ Control 01 ○ Modularity 01, 02 ○ Program Development 01, 02, 03, 05, 07 ○ Variables 01

Unit 1: Primitive Type <i>Desired Results</i>		
Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Variables 01 - Control 01 - Modularity 01 - Program Development 01, 03 AP Standards - Modularity: 1 - Variables: 1 - Control: 1	<i>Students will understand that...</i> - Variables store data values that programs use and manipulate. - The use of appropriate data types improves code readability, functionality, and efficiency. - A strong understanding of data representation and operations forms the foundation of all programming.	<i>Students will consider...</i> - What are variables and why are they fundamental to programming? - How do data types affect how we write and understand programs? - How can we use expressions and operators to process data?
Learning Targets		
<i>Students will...</i> - Declare and initialize variables using appropriate primitive data types (int, double, boolean). - Identify and use valid identifiers following Java naming conventions. - Perform arithmetic and logical operations using Java expressions. - Understand the concept of type casting and its implications. - Apply integer and floating-point division correctly in expressions. - Use basic debugging to identify and fix syntax and logic errors involving variables and expressions.		
Unit Duration:		
2-3 Weeks		

Unit 2: Using Objects***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Variables 01 - Control 01 - Modularity 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Modularity: 1 - Variables: 1 - Control: 1	<i>Students will understand that...</i> - Objects are instances of classes and allow for modular, reusable, and organized code. - Methods define the behaviors of objects and enable interactions within and between programs. - Parameters and return values allow for dynamic and flexible methods that operate on different data.	<i>Students will consider...</i> - What is an object, and how does it encapsulate data and behavior? - How do we use methods to interact with objects in Java? - What roles do parameters and return values play in method functionality?
Learning Targets		
<i>Students will...</i> - Create and use object instances of predefined classes. - Call methods using the dot notation and understand their purpose. - Pass arguments to methods and understand the role of parameters. - Use return values from methods and assign them to variables. - Recognize and use the <code>String</code> class and its common methods. - Understand and apply the concept of <code>null</code> references. - Use objects in expressions and control structures. - Debug issues related to incorrect method calls or object usage.		
Unit Duration:		
2–3 Weeks		

Unit 3: Boolean Expressions and If statements***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Variables 01 - Control 01 - Modularity 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Control 1 and Control 2	<i>Students will understand that...</i> - Boolean expressions evaluate conditions that control decision-making in programs. - Conditional statements allow programs to execute different actions based on logical evaluations. - Relational and logical operators help create flexible and powerful Boolean expressions.	<i>Students will consider...</i> - How do we represent and evaluate conditions using Boolean expressions? - What structures allow us to make decisions in code? - How can we combine conditions to form more complex logical expressions?
Learning Targets		
<i>Students will...</i> - Write and evaluate Boolean expressions using relational operators (==, !=, <, >, <=, >=). - Use logical operators (&&, , !) to combine or modify Boolean expressions. - Construct if and if-else statements to control program flow. - Understand short-circuit evaluation and how it affects Boolean expressions. - Apply Boolean logic to real-world problem solving. - Debug common errors in conditional logic and Boolean expressions.		
Unit Duration:		
2–3 Weeks		

Unit 4 : Iteration

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Control 01 - Modularity 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Control: 2	<i>Students will understand that...</i> - Iteration enables a program to execute instructions repeatedly under certain conditions. - Loops are essential for automating repetitive tasks and processing collections of data. - Choosing the correct loop structure is key to efficient and readable code.	<i>Students will consider...</i> - Why do we need repetition in programs? - What are the different types of loops and when should each be used? - How can we ensure our loops behave as expected and avoid infinite loops?
Learning Targets		
<i>Students will...</i> - Use while, for, and do-while loops to repeat code. - Design and trace loops to solve problems. - Control iteration using loop conditions and counters. - Recognize and correct off-by-one and infinite loop errors. - Apply nested loops to solve more complex problems. - Use loops to process data sets and simulate real-world processes.		
Unit Duration:		
2–3 Weeks		

Unit 5 : Writing a Class***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Variables 01 - Modularity 01, 02 - Program Development 01, 02, 03, 05, 07 AP Standards - Modularity: 1 and 2 - Variables: 1 - Impact of Computing: 1	<i>Students will understand that...</i> - Classes are blueprints for creating objects that encapsulate state and behavior. - Constructors initialize objects and set up their initial state. - Encapsulation promotes modular, maintainable, and secure code.	<i>Students will consider...</i> - What is the purpose of defining a class in Java? - How do constructors and instance variables work together to initialize objects? - Why is encapsulation important in object-oriented programming?
Learning Targets		
<i>Students will...</i> - Define classes with instance variables and methods. - Use constructors to initialize object state. - Apply proper access control using <code>private</code> and <code>public</code> modifiers. - Create and use accessor (<code>get</code>) and mutator (<code>set</code>) methods. - Instantiate objects and call methods using dot notation. - Understand the difference between class structure and object behavior.		
Unit Duration:		
2-3 Weeks		

Unit 6: Arrays

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Variables 01 - Control 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Variables: 2 - Control: 2	<i>Students will understand that...</i> - Arrays store collections of elements of the same type and allow indexed access to these elements. - Arrays can be traversed using iterative control structures such as loops. - Arrays are foundational structures used in algorithms and data processing.	<i>Students will consider...</i> - What are arrays and how are they structured in Java? - How can we access, modify, and iterate through elements in an array? - What are common mistakes when working with arrays, and how can they be avoided?
Learning Targets		
<i>Students will...</i> - Declare, instantiate, and initialize arrays. - Access and modify elements using index notation. - Use loops (e.g., for, enhanced for) to traverse arrays. - Determine and use the .length property of arrays. - Identify and prevent common array-related errors (e.g., IndexOutOfBoundsException). - Solve problems using arrays to store and manipulate data.		
Unit Duration:		
4 Weeks		

Unit 7: ArrayList

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Variables 01 - Control 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Variables: 2 - Control: 2 - Impact of Computing: 1	<i>Students will understand that...</i> - The ArrayList class provides a resizable array-like structure that supports dynamic data storage. - ArrayLists support built-in methods for adding, removing, and searching elements. - Choosing the right data structure improves program efficiency and flexibility.	<i>Students will consider...</i> - What is an ArrayList, and how does it differ from a regular array? - How do you add, access, and remove elements from an ArrayList? - When should an ArrayList be used over an array?
Learning Targets		
<i>Students will...</i> - Declare and instantiate ArrayList objects. - Use methods such as add(), get(), set(), remove(), and size(). - Traverse an ArrayList using traditional and enhanced for loops. - Understand how ArrayList manages memory and resizing behind the scenes. - Compare the use of ArrayList versus arrays in different programming scenarios. - Avoid common pitfalls such as IndexOutOfBoundsException or modifying lists during iteration.		
Unit Duration:		
4 Weeks		

Unit 8: 2D Arrays

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Variables 01 - Control 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Variables: 2 - Control: 2	<i>Students will understand that...</i> - Two-dimensional arrays model data in table-like structures with rows and columns. - Nested loops are essential for traversing 2D arrays and performing operations on each element. - Understanding the structure and access of 2D arrays enables efficient storage and manipulation of data.	<i>Students will consider...</i> - What is a 2D array, and how is it structured in Java? - How do you access, modify, and iterate through a 2D array? - What types of problems are best solved using 2D arrays?
Learning Targets		
<i>Students will...</i> - Declare, instantiate, and initialize 2D arrays. - Access and modify elements using [row][column] notation. - Use nested for loops to iterate through rows and columns. - Calculate values across rows, columns, and the entire array. - Understand memory layout and limitations of 2D arrays. - Debug common 2D array errors such as <code>ArrayIndexOutOfBoundsException</code>.		
Unit Duration:		
3 Weeks		

Unit 9: Inheritance

Desired Results

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Modularity 01, 02 - Control 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Modularity: 3	<i>Students will understand that...</i> - Inheritance allows for hierarchical class structures, enabling code reuse and extension. - Subclasses inherit attributes and behaviors from their parent (super) classes. - Overriding methods allows subclasses to provide specific behavior while maintaining a common interface.	<i>Students will consider...</i> - What is the purpose of inheritance in object-oriented programming? - How does Java implement inheritance using the extends keyword? - When and why should a method be overridden?
Learning Targets		
<i>Students will...</i> - Define superclass and subclass relationships using the extends keyword. - Call superclass constructors and use the super keyword appropriately. - Understand and apply method overriding rules. - Recognize polymorphism through inherited method calls. - Trace method calls and attribute access through inheritance hierarchies. - Explain how inheritance improves modularity, reusability, and organization in programs.		
Unit Duration:		
2 Weeks		

Unit 10: Recursion***Desired Results***

Standards	Enduring Understandings	Essential Questions
MO Computer Science Standards 11–12 Algorithms and Programming - Modularity 01 - Control 01 - Program Development 01, 02, 03, 05, 07 AP Standards - Control: 2	<i>Students will understand that...</i> - Recursion is a problem-solving technique where a method calls itself to solve a smaller instance of the same problem. - Every recursive method must have a base case to avoid infinite loops. - Recursive thinking is a powerful approach for breaking down complex problems into simpler, manageable steps.	<i>Students will consider...</i> - What is recursion, and how does it differ from iteration? - How do base and recursive cases work together in a recursive method? - When is recursion more appropriate than iteration?
Learning Targets		
<i>Students will...</i> - Define and identify recursive methods in Java. - Construct recursive solutions to problems such as factorial, Fibonacci, and string processing. - Differentiate between base cases and recursive cases. - Trace recursive calls using call stacks. - Compare recursion and iteration in terms of structure, readability, and performance. - Recognize stack overflow errors and understand how they occur.		
Unit Duration:		
2 Weeks		

Learning Plan

Week(s)	Topic	Resources/Texts	Learning Targets	Assessment
Weeks 1-3	Primitive Types	- PLTW Inkling textbook Visual Studio Code Integrated Development Environment Java Oracle documentation Java Math Module and its documentation	- Declare and initialize variables using appropriate primitive data types (int, double, boolean). - Identify and use valid identifiers following Java naming conventions. - Perform arithmetic and logical operations using Java expressions. - Understand the concept of type casting and its implications. - Apply integer and floating-point division correctly in expressions. - Use basic debugging to identify and fix syntax and logic errors involving variables and expressions.	- Code Puzzles “Numbers Riddle” Project
Weeks 3-6	Using Objects	- PLTW Inkling textbook Visual Studio Code Integrated Development Environment Java Oracle documentation Java Math Module and its documentation	- Create and use objects, and call methods on objects. - Access object fields and methods using dot notation. - Write simple methods that return values and accept parameters. - Understand the concept of object-oriented programming and how it promotes modularity and reusability. - Design classes that represent real-world concepts with fields and methods.	- Code Puzzles “Escape Room” Project “Custom Class” Project
Weeks 6-9	Boolean Expressions	- PLTW Inkling textbook Visual Studio Code Integrated Development Environment Java Oracle documentation Java Math Module and its documentation	- Evaluate boolean expressions using comparison and logical operators. - Write if statements to execute code conditionally. - Write if-else statements for binary decision-making. - Use nested if-else statements to handle multiple conditions. - Test and debug programs that involve boolean expressions.	- Code Puzzles - Code Warm-ups “Story Time” Project

Weeks 9-10	Iteration	• PLTW Inkling textbook • Visual Studio Code Integrated Development Environment • Java Oracle documentation • Java Math Module and its documentation	• Use for-each loops to iterate over arrays and collections. • Determine the proper use of iteration for solving problems based on problem requirements. • Recognize the impact of loop termination conditions on program flow. • Debug programs involving iteration and loops.	• Code Puzzles • Code Warm-ups • ‘Consumer Review’ Project
Weeks 10-12	Writing a Class	• PLTW Inkling textbook • Visual Studio Code Integrated Development Environment • Java Oracle documentation • Java Math Module and its documentation	• Define methods that accept parameters and return values. • Understand method overloading and the concept of different method signatures. • Call methods with parameters and use return values. • Use method decomposition to break down complex problems into simpler steps. • Write recursive methods and understand the role of base cases.	• Code Puzzles • Code Warm-ups • “Custom Class” Project • “Sell my Pet Food” Project
Weeks 12-16	Arrays	• PLTW Inkling textbook • Visual Studio Code Integrated Development Environment • Java Oracle documentation • Java Math Module and its documentation	• Declare and initialize arrays in Java. • Access and modify elements of an array using index notation. • Understand how to pass arrays to methods and return arrays from methods. • Perform common array operations (e.g., finding max/min, summing elements, reversing). • Understand the concept of array length and bounds checking.	• Code Puzzles • Code Warm-ups • “Elevens” Project • “Memory Game” Project
Weeks 16-20	ArrayList	• PLTW Inkling textbook • Visual Studio Code Integrated Development Environment • Java Oracle documentation • Java Math Module and its documentation	• Create and manipulate ArrayList objects using add, get, remove, and set methods. • Understand the difference between arrays and ArrayLists in terms of size and flexibility. • Iterate through an ArrayList using for-each loops. • Implement methods to find and modify elements in ArrayLists. • Use ArrayLists to store and manipulate collections of objects.	• Code Puzzles • Code Warm-ups • “Data Lab” Project

Weeks 20-24	2D Array	• PLTW Inkling textbook • Visual Studio Code Integrated Development Environment • Java Oracle documentation • Java Math Module and its documentation	• Declare and initialize 2D arrays. • Access and modify elements in 2D arrays using nested loops. • Pass 2D arrays as parameters to methods. • Implement algorithms to traverse and manipulate 2D arrays. • Understand the concept of rows and columns in 2D arrays.	• Code Puzzles • Code Warm-ups • “Real Picture?” Image • “Data Lab 2” Project
Weeks 24-32	Inheritance	• PLTW Inkling textbook • Visual Studio Code Integrated Development Environment • Java Oracle documentation • Java Math Module and its documentation	• Understand the concept of inheritance and how it allows code reuse. • Use the extends keyword to create subclasses that inherit properties and behaviors from parent classes. • Override methods in subclasses to customize behavior. • Use polymorphism to refer to objects of different subclasses through parent class references. • Design and implement class hierarchies using inheritance.	• Code Puzzles • Code Warm-ups • “Vehicles” Project • “Inventory System” Project • “Celebrity” Project
Weeks 32-36	Recursion	• PLTW Inkling textbook • Visual Studio Code Integrated Development Environment • Java Oracle documentation • Java Math Module and its documentation	• Define and identify recursive methods in Java. • Construct recursive solutions to problems such as factorial, Fibonacci, and string processing. • Differentiate between base cases and recursive cases. • Trace recursive calls using call stacks. • Compare recursion and iteration in terms of structure, readability, and performance. • Recognize stack overflow errors and understand how they occur.	• Code Puzzles • Code Warm-ups • “Recursive File Finder” Project

Proficiency Scales for AP Computer Science A

Grade Level/Course: AP Computer Science A	Standard(s): • MO Computer Science Standards ○ 11–12 Algorithms and Programming: M01, C01, M01, PG01, PG03 • AP Standards ○ MOD-1 ○ VAR-1 ○ CON-1 Domain: Primitive Types
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	• Write code using variables to execute a requested simple mathematical operation or series of operations • Write code to update variables using complex operations • Write code to change variables from one type to another • Predict the output of a given code segment that involves variable overwrites or type changes
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. • Given a scenario, the student can: ○ identify the appropriate primitive type to represent data and justify their selection ○ Modify variables using complex assignment operators • Given a data set, the student can write a series of statements to execute mathematical operations then print the results • Can cast a variable from one type to another, justifying why the new type would be appropriate for the situation	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. • Describe what primitive types exist in Java and what kind of data they contain • Initialize and create variables, as well as assign elements to the previously mentioned variables • Create System Print statements to display data in the terminal • Facilitate mathematical operations using variables and operators	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. • Can define what a primitive type is in Java • Can identify variables and their values in Java code • Can describe what a System Print is	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, M01, PD01, PD02, PD03, PD05, PD07 - AP Standards - MOD-2 - VAR-2 - CON-2 Domain: Using Objects
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write a String variable that is the sum of several other strings - Write code to determine the length of a String, then create one or more substrings of varying lengths - Write code to compare String components - Write code to evaluate the highest or lowest value from an executed Math operation - Explain the usefulness of APIs - Write code for a method that returns data and code for a method that returns no data - Describe in writing what an Object is and how they relate to classes - Write code to create an Object and assign it data - Describe in writing the uses of the Math, Integer, and Double Classes
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Store data within object(s) - Can justify why a method should be a Void method - Can use String methods to produce a requested printed output - Can illustrate how an Application Program Interface(API) is documented and its importance in programming - Can determine the highest or lowest value in a list using a Math method or a method of their own creation - Can predict the outcome of math operations that use different data types	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can describe what an Object is, its relationship to classes and create objects - Can create methods as well as describe what Void methods are - Can describe the different aspects of String objects and methods - The student can provide an explanation of the Math, Integer and Double Classes	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. - Describe how to call a function from an object or class - Describe what a class is supposed to do	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, M01, PD01, PD02, PD03, PD05, PD07 - AP Standards - CON-1, CON-2 Domain: Logic Flow and Operation Sequencing
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write code that will provide a specific response to only certain inputs - Create an equivalent Boolean expression using De Morgan’s Laws - Write code where given an input, a certain operation or series of operations is executed - Create an illustration how different flow operators work with code examples - Write code to show how simple Boolean operations are different from compound Boolean operations - Write code to compare if two objects are the same - Describe in writing what flow control is and how it is important in programming
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Using a compound Boolean statement, the student can create a code block(s) that respond to a narrow range of inputs - Can evaluate if two Boolean expressions are equivalent and justify why or present an alternate expression that is equivalent	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can create a series of “if” or “if-else” statements so several possible input varieties can be processed successfully - Can describe what operators are used for flow control and what they do - Can discuss what flow control is and how it can be enacted using “if” or “if-else” statements - Can describe how to set up a compound Boolean expression as well as describe what an equivalent Boolean expression is - Can describe how to compare two Objects	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. - Describe what an “if/else” is and its use in programming - Can show how different “if/else” options can be used	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, M01, PD01, PD02, PD03, PD05, PD07 - AP Standards - CON-2 Domain: Iteration
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write code to read the lines of a text file and print them to the terminal - Write code to signify how many times a value or substring occurs in a String - Write a description of how loops are practically useful to programmers - Create a diagram illustrating the parts of a For and While loops - Describe in writing when a For Loop would be preferable to a While Loop - Describe in writing when a While Loop would be preferable to a For Loop - Describe how code can locate a print an output based on the contents of a String
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Create a For or While loop to repeat a process multiple times, justifying their loop selection - Using Loops and Booleans, the student could illustrate how an algorithm could be constructed to produce a new String given initial parameters - Can explain the importance of and implement the use of a loop counter	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can describe the difference between For and While loops, including when they should be used - Can describe what are the parts of a For loop and what the parts of a While loop, as well as what these components do for the loop - Can explain how iteration can be nested - Can describe how an algorithm operates in the context of Strings	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Describe the types of loops in Java and when to use them	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, M01, PD01, PD02, PD03, PD05, PD07 - AP Standards - MOD-1, MOD-2 - VAR-1 - IOC-1 Domain: Writing Classes
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write code to create an accessor and a mutator for the attributes of a given class. - Describe in writing why accessors and mutators are necessary with respect to public and private variables. - Write code to create a constructor utilizing this keyword. - Add comments to computer code to illustrate and explain what the code’s objective(s) is/are. - Describe in writing what the public and private keywords are and why they are necessary in programming. - Describe in writing what accessors, mutators, and constructors are. - Describe in writing what the “this” keyword is used for. - Describe what encapsulation is and why programmers use it. - Illustrate how static methods and variables are different from normal methods and variables. - Describe how computer science can influence society in the context of human development.
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Can create an accessor and a mutator, explaining why they would have made those methods public or private - Can create a constructor, using the this keyword to assign non-default values to the attributes of an object - Can illustrate how to produce comments in computer code and explain their importance	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can describe the difference between the public keyword and the private keyword, including what part(s) of a class need to be public or private - Can describe the importance and function of accessors, mutators, constructors and the this keyword - Can describe how to document their progress as they write code - Can describe encapsulation and how it applies to classes - Can contrast the difference between static methods and static variables to regular methods and variables	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Describe what a class is and what its components are it up in Java	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01 - AP Standards - CON-2 - VAR-2 Domain: Arrays
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write the code to iterate through a given array and print only certain elements of the array using a For Loop or While Loop - Write the code to iterate through a given array and alter the data for each element - Write code to generate an array and then add elements to it. - Describe in writing what an “Array” is and why it is used - Write code to generate an array with initial elements already added - Describe in writing the properties of an array and how to locate elements of an array - Describe how you could transverse an array and access its elements
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Can create a For Each loop to search or process all the elements in an array - Can design a For loop and a While loop to access certain elements in an array while ignoring others	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can create an array, then add elements to it - Can describe what an array is and why it is used - Can explain how to set up an array, whether using initializer lists or just making an empty array - Can describe the properties of an array, such as its fixed size, syntax and where items are located in an array - Can describe means to traverse arrays, such as For loops, While loops, or For Each loops - Can explain different ways to access specific elements of an array	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Describe what an Array is in Java	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, PD01, PD02, PD03, PD05, PD07 - AP Standards - CON-2 - VAR-2 - IOC-1 Domain: ArrayLists
--	--

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write code to remove elements from an ArrayList using a loop - Write code to compare elements of an ArrayList, then alter the data based on specifications - Write code to create an ArrayList, then add elements to it - Describe in writing the difference between an Array and an ArrayList - Describe in writing what methods are available to ArrayLists and what they do - Describe in writing how you could transverse an ArrayList, as well as, how you could change elements in the ArrayList - Describe in writing how privacy is an important topic in programing and what its repercussions are
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Can write code to successfully remove a data set from an ArrayList - Can use the ArrayList methods to retrieve, compare, or change elements in an ArrayList - Can transverse an ArrayList using multiple types of loops	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can create an ArrayList and add elements to the ArrayList before and after its creation - Can describe the key differences between Arrays and ArrayLists, including the mutability and initialization - Can discuss the different methods of ArrayList and what kind of uses they might have - Can explain different ways to transverse an ArrayList and implement data changes to an ArrayList - Can discuss the ethical implications of privacy in programming	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Describe what an ArrayList is in Java	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, PD01, PD02, PD03, PD05, PD07 - AP Standards - CON-2 - VAR-2 Domain: 2-D Arrays
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write code to create a 2D- Array and add elements to it. - Write code to transverse an Array, locating requested information, using some style of loop - Describe in writing how a 2D array differs from a norma array - Describe in writing how 2D arrays are traverse
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Can create a 2D array, keeping in mind the convention of row - column ordering - Can iterate through a 2D array by selecting a row and then transversing the columns of that row - Can iterate through an array using for or enhanced for loops	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can compare and contrast a 1D array with a 2D array, including how they are used and created - Can describe how nested statements are used traverse 2D arrays	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Describe what a 2D array is in Java	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, PD01, PD02, PD03, PD05, PD07 - AP Standards - MOD-3 Domain: Inheritance
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Write code to create a class and extend its traits to another class - Write code to override the methods of a superclass and comment how this is polymorphism - Illustrate the relationship between a superclass and a subclass - Describe in writing the relationship between a “Superclass” and a “Subclass”, as well as how this can be time-saving to programmers - Describe in writing the difference between method overloading and method overriding - Describe in writing what the keywords extends and super mean
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - Can create a class, then extend its traits to a subclass - Can override the methods of a superclass in a subclass, then explain how this is an example of polymorphism - Can illustrate how a subclass “is a _____” of a given superclass	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can explain the difference between a Superclass and a Subclass - Can describe how a class hierarchy can be formed and why it could be beneficial, as well as the roll of the Superclass Object - Can describe what method overriding is, as well as what polymorphisms are - Can explain the uses of the following keywords: extends, super	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Describe what inheritance is in Java and why it helps with encapsulation	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	

Grade Level/Course: AP Computer Science A	Standard(s): - MO Computer Science Standards 11–12 Algorithms and Programming: V01, C01, PD01, PD02, PD03, PD05, PD07 - AP Standards - CON-2 Domain: Recursion
--	---

Performance Level Descriptor	Success Criteria	Sample Tasks (“How can I assess this standard?”)
4 Advanced	To achieve a 4, the advanced student displays the application of the standard beyond the “on target” success criteria.	- Given some code, the student can describe how many times it will run until the loop or method will terminate - Given some recursive code, the student can describe what the output would be - Describe in writing what a recursive method is and why it is used by programmers - Describe in writing what a “base case” is and why recursive methods cannot function without it - Describe in writing recursion’s role in sorting algorithms
3 On Target	To achieve a 3, the “on target” (proficient) student meets all success criteria for the priority standard listed. - The student can predict how many times a binary sort will run until it finds its value - Given a scenario, the student will be able to determine the output of a recursive algorithm	
2 In Progress	To achieve a 2, the “in progress” (basic) student meets all the success criteria below. - Can describe what a recursive method is and why it would be useful - Can explain what a base case is and why it is crucial to a recursive program - Can describe how recursion can be helpful in sorting data	
1 Getting Started	To achieve a 1, the “getting started” (below basic) meets all the success criteria below to show they are “getting started” toward the target goal. Describe what recursion is	
LND	The student’s current level of proficiency in the specified standard is not yet determinable, providing for the opportunity for targeted support.	