Grade 3 Technology Unit 4: Coding

Content Area: **Technology**
Course(s): **Technology Grade 3**
Time Period: **MP4**
Length: **7 days**
Status: **Published**

NJSLS - Computer Science and Design Thinking

CS.3-5.8.1.5.AP.1	Compare and refine multiple algorithms for the same task and determine which is the most appropriate.
CS.3-5.8.1.5.AP.2	Create programs that use clearly named variables to store and modify data.
CS.3-5.8.1.5.AP.3	Create programs that include sequences, events, loops, and conditionals.
CS.3-5.8.1.5.AP.4	Break down problems into smaller, manageable sub-problems to facilitate program development.
CS.3-5.8.1.5.AP.5	Modify, remix, or incorporate pieces of existing programs into one's own work to add additional features or create a new program.
CS.3-5.8.1.5.AP.6	Develop programs using an iterative process, implement the program design, and test the program to ensure it works as intended.

Rationale and Transfer Goals

Students will further develop their understanding of computer science using a unit of study developed by code.org. They will learn coding and problem solving through solving various coding puzzles, including how computers can complete some tasks significantly faster than people can without computers. This unit of study will help students understand the basics of how many digital apps, games and programs are created and could inspire them to further their knowledge of the subject as they get older. This unit will allow students to develop their critical thinking and problem solving skills, as well as their collaboration skills on challenging problems.

Enduring Understandings

Computers can be used to quickly solve problems, but they require people to think creatively and be the driving force behind the problem solving.

Essential Questions

How can I program a computer to complete a task?

How can coding affect the speed and efficiency with which the computer completes the task?

Content - What will students know?

- Definition of Program and algorithm.

- Definition of cardinal directions.

- Definition of Events.

- Definition of loops.

Skills - What will students be able to do?

- Follow an algorithm to create a pattern.

- Create algorithms to guide a character through a digital maze.

- Create small event algorithms to develop a simple game and performance.

- Use a repeated action or group of actions to solve problems.

Activities - How will we teach the content and skills?

- Large group instruction of graph paper programming, using 4x4 grids to create a pattern from a series of directions, followed by students creating their own algorithms to trade with classmates.

- Whole class instruction of commands to create an algorithm to move a character through a maze, followed by students working through the mazes independently.

- Whole class instruction of process of creating events that will react to things such as a mouse click or time interval, followed by students working independently to create a Flappy Bird game and Dance Party performance.

- Whole class discussion of actions in a dance that are repeated that can be combined into loops, followed by using loops to streamline their algorithms in the code.org modules.

Evidence/Assessments - How will we know what students have learned?

- Evaluation of students completed patterns and algorithms.

- Evaluation of students' progress through Introduction to Online Puzzles and Debugging with Laurel modules.

- Evaluation of students' progress through Events in Bounce, Build a Star Wars Game and Dance Party.

- Evaluation of students' progress through Loops in Ice Age, Drawing Shapes with Loops and Nested Loops in Maze.

Spiraling for Mastery

Content or Skill for this Unit	Spiral Focus from Previous Unit	Instructional Activity
How to combine movement and actions in algorithms	Sequential steps in a program	Whole class discussion of how to program a robot to move throughout the class and pushchairs in, followed by using the combination of movements and actions to move characters through a maze.

Key Resources
www.code.org course D

4x4 graph paper

21st Century Life and Careers

WRK.9.2.5.CAP.1	Evaluate personal likes and dislikes and identify careers that might be suited to personal likes.
WRK.9.2.5.CAP.4	Explain the reasons why some jobs and careers require specific training, skills, and certification (e.g., life guards, child care, medicine, education) and examples of these requirements.

Career Readiness, Life Literacies, & Key Skills

TECH.9.4.5.CT.1	Identify and gather relevant data that will aid in the problem-solving process (e.g., 2.1.5.EH.4, 4-ESS3-1, 6.3.5.CivicsPD.2).
TECH.9.4.5.CT.2	Identify a problem and list the types of individuals and resources (e.g., school, community agencies, governmental, online) that can aid in solving the problem (e.g., 2.1.5.CHSS.1, 4-ESS3-1).
TECH.9.4.5.CT.3	Describe how digital tools and technology may be used to solve problems.

Interdisciplinary Connections/Companion Standards

Literacy and language arts in the technology context: writing, programming, word processing, and creativity with language

Science: understanding of computer components, operations of touchscreens and other user devices

Social Studies: Computers in the context of society; our relationships to computers as a tool

Health: Limits to screen time and healthy relationships with technology, online technologies

Art: Extensive connections to visual art, design, and multimedia creation through movie-making software

| SCI.3-5-ETS1-1 | Define a simple design problem reflecting a need or a want that includes specified criteria for success and constraints on materials, time, or cost. |