Grade 7 STEM Unit 3: Coding

Content Area: **STEM**
Course(s): **STEM Grade 7**
Time Period: **MP1**
Length: **7 days**
Status: **Published**

NJSLS

MA.4.MD.C.5	Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:
MA.3.G.A.1	Understand that shapes in different categories (e.g., rhombuses, rectangles, and others) may share attributes (e.g., having four sides), and that the shared attributes can define a larger category (e.g., quadrilaterals). Recognize rhombuses, rectangles, and squares as examples of quadrilaterals, and draw examples of quadrilaterals that do not belong to any of these subcategories.
MA.5.G.B.4	Classify two-dimensional figures in a hierarchy based on properties.
CS.6-8.8.1.8.AP.2	Create clearly named variables that represent different data types and perform operations on their values.
CS.6-8.8.1.8.AP.3	Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.
CS.6-8.8.1.8.AP.4	Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs.
CS.6-8.8.1.8.AP.5	Create procedures with parameters to organize code and make it easier to reuse.
CS.6-8.8.1.8.AP.7	Design programs, incorporating existing code, media, and libraries, and give attribution.
CS.6-8.8.1.8.AP.8	Systematically test and refine programs using a range of test cases and users.

Rationale and Transfer Goals

Teaching coding in middle school provides students with valuable skills such as problem-solving, logical thinking, and creativity. It also introduces them to technology, an essential aspect of our world. Coding promotes computational thinking and can lead to future career opportunities in technology-related fields. Additionally, it fosters teamwork, as coding often involves collaboration on projects, enhancing communication skills. Overall, learning coding early equips students with tools to navigate the digital age effectively.

Enduring Understandings

Algorithmic Thinking: Students should understand that coding involves breaking down tasks into smaller steps (algorithms) to solve problems efficiently and logically.

Programming Logic: Students should grasp the concept of conditional statements, loops, and variables, which are fundamental to creating functional programs and applications.

Problem-Solving Skills: Students should learn that coding is a powerful tool for solving real-world problems and that different coding solutions can be applied to various scenarios.

Debugging and Persistence: Students should understand that debugging is a natural part of coding and that persistence and troubleshooting are essential skills in identifying and rectifying errors.

Essential Questions

How does coding contribute to solving real-world problems and improving daily life?

What are the fundamental concepts and principles underlying coding languages and algorithms?

How does computational thinking help break down complex problems into smaller, manageable components?

How does debugging and troubleshooting contribute to the development of coding skills and problem-solving abilities?

How can the skills acquired through coding education be transferable to other areas of learning and future career opportunities?

Content - What will students know?

- Beginner and intermediate coding techniques
- Complex problem identification
- Brainstorming and Problem Solving

Skills - What will students be able to do?

- Define Coding
- Develop code
- Create an interactive animation using behaviors in computer science.
- Develop geometrical shapes using coding

Activities - How will we teach the content and skills?

- Coding Intro Activity
- Coding Angry Birds
- Drawing Geometrical Shapes using Codes
- Coding Geometrical Star Quilt

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

- Question and answer worksheet accompanying google slides Pre/post quiz
- Final Code.org assignment for Angry Birds
- Final presentation for Star Quilt

Spiraling for Mastery

Content or Skill for this Unit	Spiral Focus from Previous Unit	Instructional Activity
Coding and Programming	Engineering design process and problem-solving.	Students will be reintroduced to the concepts of coding and programming through an introductory activity in Code.org

Key Resources

21st Century Life and Careers

WRK.9.2.8.CAP.12	Assess personal strengths, talents, values, and interests to appropriate jobs and careers to maximize career potential.

Career Readiness, Life Literacies, & Key Skills

TECH.9.4.8.CT.1	Evaluate diverse solutions proposed by a variety of individuals, organizations, and/or agencies to a local or global problem, such as climate change, and use critical thinking skills to predict which one(s) are likely to be effective (e.g., MS-ETS1-2).
TECH.9.4.8.CT.2	Develop multiple solutions to a problem and evaluate short- and long-term effects to determine the most plausible option (e.g., MS-ETS1-4, 6.1.8.CivicsDP.1).
TECH.9.4.8.CT.3	Compare past problem-solving solutions to local, national, or global issues and analyze the factors that led to a positive or negative outcome.

Interdisciplinary Connections/Companion Standards

LA.RI.7.4	Determine the meaning of words and phrases as they are used in a text, including figurative, connotative, and technical meanings; analyze the impact of a specific word choice on meaning and tone.
LA.W.7.4	Produce clear and coherent writing in which the development, organization, voice and style are appropriate to task, purpose, and audience. (Grade-specific expectations for writing types are defined in standards 1–3 above.)
LA.W.7.6	Use technology, including the Internet, to produce and publish writing and link to and cite sources as well as to interact and collaborate with others, including linking to and citing sources.