STEAM – Unit 2 (COMPUTER PROGRAMMING)

Content Area: **Computer Programming (Grade 1)**
Course(s):
Time Period: **Ongoing**
Length: **Ongoing**
Status: **Published**

Big Idea

Learning computer science encourages critical thinking, logic, persistence, and creativity. It can help students excel at problem-solving in all subject areas, no matter what their age. At the root of all computer science is something called an algorithm. The word "algorithm" may sound like something complicated, but really it's just a list of instructions that someone can follow to achieve a result. To provide a solid base for computer science education, students should build a secure relationship with algorithms. (CODE.ORG, 2017)

Enduring Understanding

SWBAT write algorithms to control a digital character's motion, sound, color, and response to stimuli. SWBAT identify patterns in code.

Skills

- Using digital block coding applications, students write algorithms to control a digital character and it's surroundings. Students write algorithms to control the digital character's motion, sound, color, and response to stimuli (cues).
- Solve problems using sequential algorithms, debugging algorithms, creating loops in algorithms, and creating conditional algorithms.
- Practice problem solving and perseverance techniques.

Standards

8.2.2.E.1   List and demonstrate the steps to an everyday task.
 8.2.2.E.2   Demonstrate an understanding of how a computer takes input through a series of written commands and then   interprets and displays information as output.
 8.2.2.E.3   Create algorithms (a sets of instructions) using a pre-defined set of commands (e.g., to move a student or a character through a maze).
8.2.2.E.4   Debug an algorithm (i.e., correct an error).
 8.2.2.E.5   Use appropriate terms in conversation (e.g., basic vocabulary words: input, output, the operating system, debug, and algorithm).

Assessments

- Tracking of student achievement in COURSE B of Code.org's (or other online coding application's) lessons.
- Teacher observation

Resources/Instructional Materials

CODE.ORG COURSE B. https://studio.code.org/s/courseb?section_id=1057462. Learn the basics of computer science and internet safety. At the end of the course, create your very own game or story you can share.

Lesson 4: Sequencing with Drag and Drop
This lesson will give students an idea of what to expect when they head to the computer lab. It begins with a brief discussion introducing them to computer lab manners, then they will progress into using a computer to complete online puzzles. The main goal of this lesson is to build experience with computers. By covering the most basic computer functions such as clicking, dragging, and dropping, we are creating a more equal playing field in the class for future puzzles. This lesson also provides a great opportunity to introduce appropriate computer lab behavior.

Lesson 7: Programming in Maze
Using characters from the game Angry Birds, students will develop sequential algorithms to move a bird from one side of a maze to the pig at the other side. To do this they will stack code blocks together in a linear sequence, making them move straight, turn left, or turn right. In this lesson, students will develop programming and debugging skills on a computer platform. The block-based format of these puzzles help students learn about sequence and concepts, without having to worry about perfecting syntax.

Lesson 8: Programming with Rey and BB-8
In this lesson, students will use their newfound programming skills in more complicated ways to navigate a tricky course with BB-8. With transfer of knowledge in mind, this lesson gives students a new environment to practice the skills that they have been cultivating. Star Wars fans will jump for joy when they see these puzzles. Each puzzle in this series has been added to provide a deeper understanding of the basic concepts that they will be using throughout the rest of this course.

Lesson 10: Loops in Collector
Building on the concept of repeating instructions from "My Loopy Robotic Friends," this stage will have students using loops to collect treasure more efficiently on Code.org. In this lesson, students will be learning more about loops and how to implement them in Blockly code. Using loops is an important skill in programming because manually repeating commands is tedious and inefficient. With these Code.org puzzles, students will learn to add instructions to existing loops, gather repeated code into loops, and recognize patterns that need to be repeated.

Lesson 11: Loops in Artist
Returning to loops, students learn to draw images by looping simple sequences of instructions. In the previous online lesson, loops were used to traverse a maze and collect treasure. Here, students use loops to create patterns. At the end of this stage, students will be given the opportunity to create their own images using loops. This lesson gives a different perspective on how loops can create things in programming. Students will test their critical thinking skills by evaluating given code and determining what needs to be added in order to solve the puzzle. Students can also reflect on the inefficiency of programming without loops here because of how many blocks the program would require without the help of repeat loops.

Lesson 13: Events in Play Lab
In this online activity, students will have the opportunity to learn how to use events in Play Lab and apply all of the coding skills that they've learned to create an animated game. It's time to get creative and make a game in Play Lab! In this online activity, students will learn how to use events in Play Lab. They will start by training the knight to move when an arrow key is pressed, then end with the opportunity to showcase the rest of the skills that they learned throughout this course, including sequence and looping, as part of the final freeplay puzzle.

WEB SITES

- Code.org lesson plans
- Scratch lesson plans
- Tynker lesson plans

Modifications

Individual accommodations

• Additional support

• Adapting lessons to meet various learning styles

Integration of 21st Century Skills

Focus on the development of 21st Century Content Skills:

- Global awareness
- Civic literacy
- Health and wellness awareness
- Environmental literacy

Focus on the Development of Learning and Thinking Skills:

- Critical Thinking and Problem Solving Skills
- Communication Skills
- Creativity and Innovation Skills
- Collaboration Skills
- Information and Media Literacy Skills
- Contextual Learning Skills

Focus on the Development of Life Skills:

- Leadership
- Ethics
- Accountability
- Adaptability

- Personal Productivity

- Personal Responsibility

- People Skills

- Self Direction

- Social Responsibility

Interdisciplinary Connections

- Academic and Technical Rigor - Projects are designed to address key learning standards identified by the school or district.

- Authenticity - Projects use a real world context (e.g., community problems) and address issues that matter to the students.

- Applied Learning - Projects engage students in solving problems calling for competencies expected in high-performance work organizations (e.g.,teamwork, problem-solving, communication, etc.).

- Active Exploration - Projects extend beyond the classroom by connecting to community explorations.

- Adult Connections - Projects connect students with the wider community.

- Assessment Practices - Projects involve students in regular, performance-based exhibitions and assessments of their work; evaluation criteria reflect personal, school, and real-world standards of performance.