

EPIISODE: FOUR	TITLE OF LESSON: Using Digital Tools to Solve Problems
AUDIENCE: 4-12th Grade	DURATION: 60 Minutes
LEARNING OUTCOMES: <ul style="list-style-type: none"> - Analyze complex puzzles using computational thinking - Demonstrate algorithmic thinking - Create an algorithm to solve a complex problem 	
LEARNING OVERVIEW: <p>Students begin understanding and demonstrating the concepts of algorithmic thinking and computational thinking. The instructional content will teach students how to break down complex problems into smaller parts in a way that lends to a greater understanding of the problem and possible solutions.</p> <p>This lesson will also utilize concepts drawn from the ISTE Student Standards, specifically standard 5, Computational Thinker.</p>	
IN THIS LESSON: <p>This instructional content will teach students about computational thinking and algorithmic thinking. The students will then use what they have learned to break down a puzzle/challenge and hopefully find a solution. The solution will be an algorithm that they custom create.</p>	
FINAL PRODUCT: <p>Students share the algorithm.</p>	
FACILITATOR RESOURCES: <p>SLIDES LINK: Instructional Content Video & Video Guide</p>	ADDITIONAL MATERIALS NEEDED: <ul style="list-style-type: none"> - Maze Robo Challenge image (in slides) - Google Slides - Algorithm BBC Learning Video
FLOW OF LESSON: <ul style="list-style-type: none"> - Introduce this lesson by asking the students what they know about algorithms and algorithmic thinking <ul style="list-style-type: none"> - (Complete the introduction that follows this lesson guide) - Introduce other vocabulary words - Algorithm & Cake example - Teach Instruction based on Slidedeck Presentation. <ul style="list-style-type: none"> - Teach the students how to use Google Slides to collaborate - Teach students how to Identify and articulate a problem (articulate the solution) - Begin challenge by explaining the challenge Scenario (leaving out the specific detail of up=Down, ETC...) - Allow students to attempt solving the problem and create their algorithm - How to recognize and articulate a problem/challenge 	

INTRO:

To begin this lesson ask the class what they think the word Algorithm and Algorithmic Thinking means.

- Many students will not have any idea what these words or terms mean. Some students may attempt to answer the question, and if they do, allow them to attempt to answer it
- Once a few students have attempted to answer the question, instruct the students to turn on their devices and have them use their Finding & Filtering skills to search the internet for the answers

Allow the students to share results. Some students may have answers close to the correct answers, and some students may have answers that are completely off. No matter what their results are, make sure to read the correct definitions for these terms as seen below in the **Vocabulary** section.

After you have completed sharing the correct definitions, illustrate how the recipe for baking a cake is an algorithm.

INSTRUCTION:

ASK STUDENTS: How often do you or your parents look up the answer to something you/they don't know?

CONTENT on SLIDES.....

ACTIVITY:**Overview:**

- If you have just completed the Episode 4 challenge called "Enigma" then your students have already attempted to solve a complex system of interconnected puzzles that required them to think in more abstract ways. As a class activity or homework assignment, students will be instructed that they must attempt this new challenge with the goal of working on the same skills they used in the episode. This task comes with an added twist; the students will be working with a robot program that will have "bugs/glitches" in the code that they will need to identify before they can solve the challenge
- Using the computational thinking process learned in the instruction portion, each tribe will need to guide the robot through a maze through a series of commands. Tribes will not be instructed on how to handle the challenge or what the twist is; it is up to the students to link together the algorithm that guides their robot to the end of the maze with minimal administrative guidance
- Students will need to write down their code in a google slide before submitting their program so they must visualise the path before submitting.

Objective:

- Use the computational thinking process to guide a robot through a maze using a chain of block code
- Record what happens to the robot when you input each directional cue
- Create a sequence of directions that will lead the robot through the maze and out the other side

Activity Set Up:

- Prepare to share the Maze Robo link and a Blank Google Slides

Activity Step 1:

- Begin the activity by going over the lesson 4 Presentation Slides
- Review the thinking strategies when encountering problems (Analyze, Design, Implement, Test)

Activity Step 2:

- Introduce the "Robot Maze" challenge to the students. Open the image onto a projector.
- Inform the students that they need to help guide the robot to the end of the maze

Activity Step 3:

- Tell students to visualise the path the robot needs to take to get out of the maze and then ask them to record it onto their google doc.
- Ask students to submit their solution via the google slide that was previously sent to them. This google slide can also be found here: [here](#). Have them rename the document to “Group name - sequence”. More info on this is on google slide 8
- Show the students how to add shapes to their presentations

Activity Step 4:

- After they have had enough time to write out their answers in their google slide for algorithm 1, we will throw in the twist, the robot is experiencing bugs.
- Emphasize that the robot “Is experiencing some bugs and errors that can cause issues, but we need to get the robot to the end of the maze regardless”
- Inform the students, “The robot thinks down is up, up is left, left is right, and right is down! How can we communicate our commands to the robot so that it still makes it through the maze?”
- Ask students to submit their solution via the google slide again. Tell students that their answer will go under, “Our 2nd sequence”.

Activity Step 5:

- Use discussion questions to stimulate an ending discussion around computational thinking

Activity Discussion Questions:

- Did you have any problems with visualizing the path for the robot?
- What did you do when the robot experienced the glitches?

Example:

- [Student Algorithm Example](#)

CONCLUSION:

Students share their Algorithms with the class or teacher.

Vocabulary

- **Algorithm** - a process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer
- **Algorithmic thinking** - The ability to develop precise instructions or sequences that form the basis for algorithms
- **Sequence** - a particular order in which related events, movements, or things follow each other
- **Computational thinking** - taking algorithmic thinking and expressing the solution in a way that a computer can understand and carry it out

Differentiation Tips:

The goal of differentiation is to give every student access to understanding or solving a problem. We do not want to take away the opportunity to think through the challenge, but rather give them the scaffolding necessary to reach their own conclusions.

- Spend additional time on helping the students understand the vocabulary terms.
- Stop frequently to check for understanding of the instructions and understanding of the vocabulary terms.
- Play the [BBC Learning Algorithm Video](#). This video is great for helping students grasp the concepts.
- Share with students this video explaining how to make arrows in Google Slides [Google Slides Arrows](#).
- Have the students use the simplified Maze on the slide after the normal Maze

Alternative activity prompt:

- Have each tribe elect one member as the “robot” and everyone else in the group as the “programmers”
- The goal of the programmers is to get their group’s robot to complete simple goals that you have chosen.
- The programmers will need to write on paper all of the steps their robot needs to do to accomplish the task. The robot should interpret the commands as literally as possible.
- Examples of tasks: write your name, do the macarena, make a silly face.

