LEGO® Education
Computer Science Learning Progression

SPIKE™ Essential Grade 5

Introduction

LEGO® Education believes that students learn best through play—by actively doing, exploring, and experimenting. This approach empowers them to become creative and engaged lifelong learners, which is essential for success in their future careers and lives.

Read this Introduction to explore ways to use this learning progression and find activities that support your learners.

This learning progression organizes activities in a recommended sequence that supports students' successful learning with LEGO® Education SPIKE™ Essential. For classroom convenience, it also clusters activities that use the same model.

Following the recommended sequence ensures that students build the necessary knowledge and experience for each successive activity. However, you may also choose activities according to your students' needs and prior knowledge/experience.

Some activities are reprinted or modified from published LEGO Education sources. Others are developed especially for these learning progressions.

Each activity

- ☑ contains anticipated timing, topics, relevant standards, learning objectives, and a ready-to-use prompt.
- ☑ is labeled with one or more topics, Modifying Programs (computer science) or Narrative Writing (ELA).
- ☑ lists the relevant standards, beginning with the most important standard in the learning.

To find what you need,

- ☑ scan the Topic(s) & Standards column or search with terms like *CSTA*, *ELA*, or *Math*.
- ☑ use the **Key** below to locate activities of different lengths and levels of instructional support.
- ☑ use the **Additional Resources** below to locate more support.

Key

1 Numbers show the recommended order in which to use activities.

Activities that will take approximately 20–30 mins

LESSON Longer activities with full lesson support

PROMPT Short activities to quickly expand or extend the learning

Activities that use only bricks and require no hardware/software

or Activities that will take approximately 45 or 90 mins

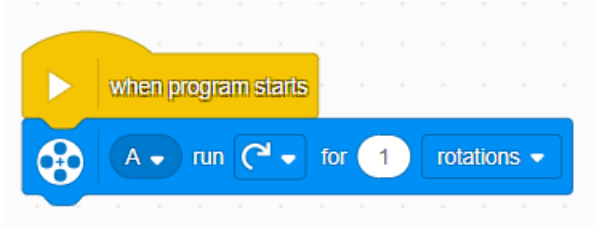
MORE DETAILS Links that lead to lesson details and teaching support

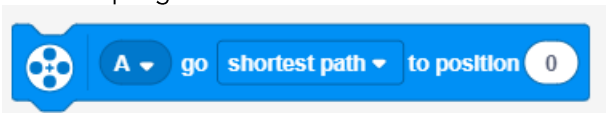
Additional Resources (also see the LEGO® Education Community)

- ☑ *SPIKE™ App Help* Definitions and directions for using the coding blocks located in the HELP section of the LEGO® Education SPIKE™ App
- ☑ *Curriculum Integration Guide* SPIKE Essential activities organized by domain Also contains a protocol for integrating activities into your curriculum
- ☑ *Coding Blocks in LEGO® Education SPIKE™ Essential Lessons*
- ☑ *Basic Coding Concepts in LEGO® Education SPIKE™ Essential Lessons*
- ☑ *Troubleshooting with LEGO® Education SPIKE™ Essential*
- ☑ *Computational Thinking in LEGO® Education SPIKE™ Essential Lessons*

#	Activity Name	Topic(s) and Standards	Objectives Students will	Prompt
1 (L)	PROMPT **Counter-balance the Bricks** 	**Design Engineering** **NGSS 3-5-ETS1-2** Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem. **NGSS 3-5-ETS1-3** Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.	• Design and build a model that requires a counterbalance for stability. • Test to determine if the model works as intended.	Introduce students to design engineering and model-building with a simple bricks-only activity. Share or reinforce that *counterbalance* "describes forces that can keep heavy objects from falling over, e.g., on the back of a crane." (Show pictures of cranes as needed.) Organize groups and provide each with the same number of bricks. Prompt groups to test how many bricks they can get to hang off the table (extending beyond the surface of the table). They will need to create a counterbalance on the table to keep the bricks from falling. **Say** *Build a model that works like a crane. It should use weight to counterbalance the load that is suspended. Use bricks to extend a model off the edge of the table. Then add a counterbalance to keep them from falling.*

2	**PROMPT** **Meet the Motor and New Motor Blocks**	**TESTING, COMPUTATIONAL PROBLEMS** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. **TROUBLESHOOTING STRATEGIES** **CSTA 1B-CS-03** Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies.	• Follow instructions to create a program. • Compare programs using two different block languages to turn a motor. • Describe coding steps in sequence.	• Introduce students to the motor in their set as they prepare to program it. • Using the Motor tutorial, have students start the motor. Then prompt them to describe the coding steps in words to a partner. • When students have completed the motor tutorial with icon blocks, have students click on tutorial 5 Word Blocks. • Have students discuss the similarities and difference in using the icon blocks to control the motor and the word blocks. **SAY** *Connect a small motor to your hub. Follow the tutorial steps to make it move. Then tell your partner step by step what the code does. Say what happens in order.* *Next, complete the word block tutorial and program the motor using word blocks. Discuss similarities and differences between the icon blocks and word blocks with your partner.* *Discuss strategies for troubleshooting hardware and software (e.g. low battery, Bluetooth disabled, hardware not connected properly).* **MORE DETAILS** *Motor Blocks* in the Help section of the LEGO® Education SPIKE™ App, available on the web or downloaded.

3	**PROMPT** **More with Motors Rotations and Degrees**	**TESTING, COMPUTATIONAL PROBLEMS** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. Sequencing **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **TROUBLESHOOTING STRATEGIES** **CSTA 1B-CS-03** Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies.		1. Have students begin lesson using word block tutorial with one motor. 2. *Ask students to click on the three dots found at the bottom of the block palette* ⋮ Lesson *and select" SPIKE Essential" to view all the blocks available.* 3. *Have students create the following program to explore the difference between program for rotation and programming for degrees. Ask students to take some observational notes while exploring.*  **Say:** *sometimes when programming the motor, we want the motor to run for a specific duration. What do you notice when programming in rotations? What do you notice when programming in degrees? What do you think degrees is referring to on the motor? How can understanding this programming block help with troubleshooting?*

4	**PROMPT** **More with Motors and Word Blocks**	**TESTING, COMPUTATIONAL PROBLEMS** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. Sequencing **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **TROUBLESHOOTING STRATEGIES** **CSTA 1B-CS-03** Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies.	• Describe the function of different block types. • Explore different ways to program a motor using word blocks. • Use appropriate terminology when using hardware and software.	1. Have students begin lesson using word block tutorial with one motor. **SAY** *Let's learn more about word block coding. Use the Motor Blocks that we learned about to program your motor to move in different ways. To start, program the motor to change directions and speed.* 2. *Share with students the technical names for the word blocks.* **SAY** *The word block "when program starts" is an event. When we use word blocks, events are all Hat Blocks (have the curved top so you can only stack blocks underneath). Hat Blocks are necessary to start a programming stack and are triggered when an event occurs.* *Stack Blocks are used for different commands in a program, in this case to turn a motor in a certain direction or with a specific power level. The notch on the top of the block and on the bottom of the block allow the blocks to be stacked together.* 3. Ask students to click on the three dots found at the bottom of the block palette ⋮ Lesson *f* and select" *SPIKE Essential*" to view all the blocks available. Then from the Motor blocks, select the ***Motor Go to Position*** block to program the motor.  Direct students to click on the drop-down menu to see the options: shortest path, clockwise and counterclockwise. Have students create a chart to collect observational data for testing. For each test, realign the gray dots on the motor and hub to start from the same position each time.

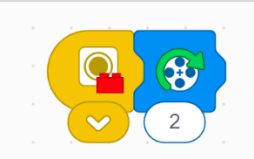
	To position 45	*To position 180*	*To position 315*
Shortest path			
Clockwise			
Counterclockwise			

Discuss with students what they noticed when programming the motor using the shortest path, clockwise or counterclockwise as well as number used for the position location. Ask students to consider how understanding the function of this programming block can help with troubleshooting.

MORE DETAILS *Word Blocks* tutorial in the START section of the SPIKE App, available on the web or downloaded.

More information about the *Motor Go To Position* programming block can be found in the help section of the SPIKE app.

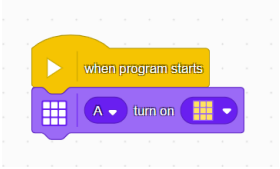
5 🕒 🕒	**PROMPT:** **Dance Party**	**COLLABORATING AROUND COMPUTING** **CSAT 1B-AP-16** Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development. **SEQUENCES** **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **TESTING, COMPUTATIONAL PROBLEMS** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate.		1. Discuss with students how to share roles when working with a partner when programming. What responsibilities could each member have during the activity? (e.g. "driver" of the computer, builder, parts manager, program developer, note taker, tester). 2. List roles on the board and talk about what each role would involve and when the role might be needed within the project. 3. For programming, have one student be a program developer (explain what programming blocks are needed and the order) and the second student be the driver of the computer (build the program on the computer). Students can switch roles during the lesson. 4. Using the word block tutorial, extend the activity to create a dancing robot. **SAY** *Connect an axle to your motor. Add other bricks to the axle to build a character that can move. Then connect the motor to the hub and program the motor to make the character dance.* 5. ***Then try this>*** *Now program the motor to move to a beat, playing slow and then faster music. Which programming blocks for the motor might work best? Remember to click on the three dots at the bottom of the block palette and select SPIKE Essential to see all the programming blocks. Be sure to switch roles.* Wrap up the activity with a gallery walk and discussion around why taking on different roles is helpful when designing programs.
6 🕒	**PROMPT** **Meet the Color Sensor**	**TESTING, COMPUTATIONAL PROBLEMS** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. **SEQUENCES** **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals.	• Create a simple pseudocode to describe the steps needed in a program. • Use pseudocode to create a program to complete a task. • Use appropriate terminology when	1. Introduce the Color Sensor with the tutorial in the App. Once students have the motor moving with the Color Sensor, prompt them to describe what each coding step does. 2. Introduce students to pseudocode. **SAY** *Pseudocode is the description of steps needed to complete a programming task written in everyday language. How can we describe the steps in this program using pseudocode?*

			using hardware and software.	When the color sensor detects red Run the motor clockwise for 2 rotations.  **SAY** *Now that we have our pseudocode, or steps needed, let's try to program the color sensor using word blocks.* Challenge students to create a program using the pseudocode in word blocks. Have students test their programs. What was similar and what is different in the programming languages? When the color sensor detects red Run the motor clockwise for 2 rotations.  **Note:** Students will need to navigate back to the home screen of the SPIKE App by clicking on the House Icon and selecting a new program using word blocks. The entire palette of programming blocks will appear. You could take the time at this point to share how blocks are organized and review the purpose of an event as well as hat blocks and stacking blocks. **MORE DETAILS** *The Color Sensor* tutorial in the START section of the LEGO® Education SPIKE™ App, available on the web or downloaded.

<table>
<tr>
<td>7</td>
<td>PROMPT
Meet the Light Matrix</td>
<td>SEQUENCES
CSTA 1B-AP-10 Create programs that include sequences, events, loops, and conditionals.

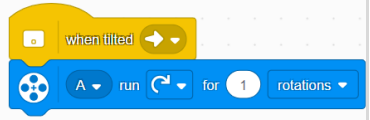
TESTING, COMPUTATIONAL PROBLEMS
CSTA 1B-AP-08 Compare and refine multiple algorithms for the same task and determine which is the most appropriate.</td>
<td>• Investigate the Light Matrix.
• Use word blocks to create sequences using the Light Matrix.
• Use appropriate terminology when using hardware and software.</td>
<td>1. Introduce the Light Matrix as students program it to show light patterns. Invite them to share their work.
SAY The SPIKE team wants to use the Light Matrix in their adventures. Plug the Light Matrix into the hub and try programming it with word blocks to make light patterns. Show your patterns to other groups.
2. Review the term pseudocode with students (the description of steps needed to complete a programming task written in everyday language)
3. Challenge students to write pseudocode for the icon block program they created.
SAY Let's write a description of the steps we needed to make the light block turn on the yellow lights. Remember we call this description pseudocode.
When play is pressed

The Light Block will play and turn all the lights yellow.
 
4. Challenge students to create a program using the pseudocode in word blocks. Have students test their programs. What was similar and what is different in the programming languages?
When play is pressed

The Light Block will play and turn all the lights yellow.
 

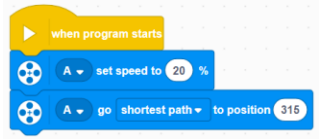
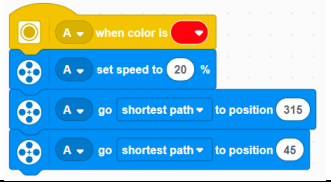
Note: Students will need to navigate back to the home screen of the SPIKE App by clicking on the House Icon and selecting a new program using word blocks. The entire palette of programming blocks will appear. You could take the time at this point to share how blocks are organized and review the purpose of an event as well as hat blocks and stacking blocks.

MORE DETAILS The Light Matrix tutorial in the START section of the SPIKE App, available on the web or downloaded.</td>
</tr>
</table>

8	**PROMPT** **Meet the Gyro Sensor**	**TESTING, COMPUTATIONAL PROBLEMS** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. **SEQUENCES** **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals.	• Follow instructions to create a program for the Built-in Gyro sensor. • Create word block sequences using sensors.	1. Have students complete the Built-In Gyro sensor tutorial found in START on the SPIKE app. 2. Review the term pseudocode with students. 3. Have students work with their partner to write the pseudocode for the icon block program in the tutorial. When the hub is tilted right Turn the motor clockwise for one rotation  4. Challenge students to create the program in word blocks using the pseudocode. When the hub is tilted right Turn the motor clockwise for one rotation  5. Have students compare the programs. What is similar and what is different. 6. Next have students connect the Light Matrix to the hub and try programming the Gyro Sensor (using word blocks) to control the Light Matrix so it creates light patterns. As needed, use gesture to clarify tilt and/or have students complete the Built-In Gyro Sensor tutorial. (note, the programming language in the Built-In Gyro Sensor tutorial used icon blocks and a motor) **SAY** *Practice using the Gyro Sensor that is built into the hub. Connect the Light Matrix to the hub. Write a program with word blocks where the Gyro Sensor makes the Light Matrix create different patterns based on movement of the hub.*

#	Activity Name	Topic(s) & Standards	Objectives Students will	Prompt
9 🕒	PROMPT **Meet the Team**	SEQUENCES **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals.	• Design and program a model to represent an idea.	1. As a class, read the bios for Maria, Daniel, Sofie, and Leo. 2. Ask students to think the character traits shared in the bio for each member of the team. 3. Have pairs of students build a model that represents one team member (Maria, Daniel, Sofie or Leo). Encourage students to use the motor and/or sensors in the build. 4. Have students share their models, describing the character and the specific details shared from the text.
10 🕒	PROMPT **Break Down Information**	NETWORKS AND THE INTERNET **CSTA B-NI-04** Model how information is broken down into smaller pieces, transmitted as packets through multiple devices over networks and the Internet, and reassembled at the destination	• Discuss how information is transmitted over networks.	Engage students in a discussion around how they think networks transmit information, then introduce the activity. • Have students select 6 bricks from the SPIKE Essential set. Ask students to build a model and then write the steps down for constructing the model on paper. • Organize students in groups of three and designate a student A (sender), B and C. • Have student A and student C sit far apart from one another with a clear path between. • Have student A take their model apart. Then have student B take the directions and one or two bricks at a time to student C. The intent is for the pieces to not arrive in order and the directions may arrive before all of the pieces. • Student C then reassembles the model. Discuss with students how the process is similar to the way information is transmitted over a network.

	**Good Morning Machine**			
#	**Activity Name**	**Topic(s) & Standards**	**Objectives** Students will	**Prompt**
11 Lesson		**Develop Programs** **CSTA 1B-AP-17** Describe choices made during program development using code comments, presentations, and demonstrations. **Design Engineering** **NGSS 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.	• Define and understand a problem. • Brainstorm and iterate to create a solution that meets the described needs. • Engage effectively in a range of collaborative discussions.	Have students use both engineering design and programming to build and program a waving machine so Leo can say "good morning." Prompt them to upgrade the build and program. **Say** *Leo is frustrated that he can't say "good morning" to his friends. He's too short and gets lost in the crowd. Help Leo by building and programming a waving machine.* **More Details** *Good Morning Machine lesson* or access in the LEGO® Education SPIKE™ App.

12	**PROMPT** **More with Computer Science**	**COMPUTATIONAL THINKING** **CSTA 1B-AP-15** Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. **CSTA 1B-AP-17** Describe choices made during program development using code comments, presentations, and demonstrations.	• Identify the parts of an existing program that should be modified. • Carry out tests to identify where a program can be modified. • Add a comment to document a change in a program.	Discuss debugging and prompt students to practice with the code examples below. For each example, have them try the program to see if they can find the mistake. Show students how to add a comment in the SPIKE app by right clicking in the program space and selecting "Add Comment". Ask students to document how they fixed the program. **SAY/ASK** *Leo is having problems with some programming. Can you help? Look at each example. Try it. What is the mistake? How can you fix it?* **Example**  **Example**  **Example** 
13	**PROMPT** **More with ELA**	**SEQUENCES** **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **RESEARCH FOR WRITING** **CCSS.ELA-Literacy.W.5.7** Conduct short research projects that use several sources to build knowledge through investigation of different aspects of a topic.	• Combine information from several research sources to build knowledge about greetings in different cultures. • Use personal experiences to identify different types of greetings.	Have students conduct a short research project about how people from different cultures greet each other in the morning. As needed, direct them to appropriate research sources or search terms. If you wish, start by prompting sharing of different personal experiences with greetings, and encourage students to use each other as sources. Modify the Good Morning Machine model to demonstrate how people from one of the researched cultures greet each other in the morning.

		CCSS.ELA-Literacy.W.5.8 Recall relevant information from experiences or gather relevant information from print and digital sources; summarize or paraphrase information in notes and finished work, and provide a list of sources.		**SAY/ASK** *How do you say good morning to your family or friends? How do we say good morning in class? Learn more about ways that people in different cultures greet each other in the morning. Use people you know and text or media research. Share what you learn.*

		Driving Around		
#	**Activity Name**	**TOPIC(S) & Standards**	**Objectives** Students will	**Prompt**
14 🕒🕒	**PROMPT** **More with Math and ELA**	**DECOMPOSITION** **CSTA 1B-AP-11** Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. **SEQUENCES** **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **INFORMATIVE WRITING** **CCSS.ELA-Literacy.W.5.2.D** Use precise language and domain-specific vocabulary to inform about or explain the topic. **GEOMETRY** **CCSS.MATH.Content.5.G.A.1-2** Graph points on the coordinate plane to solve real-world and mathematical problems.	• Decompose the steps needed to move from one location to another. • Measure distances on a route in order to create a gridded map. • Use precise language to write accurate directions from one place to another. • Build and program a robotic vehicle to follow a route.	Have students dictate or write directions from the classroom to another part of the school, such as the cafeteria or the playground, or from one part of the classroom to another. Then prompt students to try the directions, measuring and determining details for accuracy such as how far down the hall, how many footsteps to take, and which way to turn. Ask them to transfer their route to a gridded map that shows *exactly* how far to go and when to turn, so that a robot can use it to navigate the school. Explain to students the process of breaking down the steps is called decomposition. Then prompt students to build and program a robot to follow the directions. **SAY/ASK** *What's the way from our classroom to the cafeteria or playground? Write some directions to one of these places or another at school. Then try the directions, measuring and adding details for accuracy. How far "down the hall"? How many footsteps until I turn? Which way do I turn? Transfer the route to a gridded map that shows these details, so that a robot can use the map to navigate the school.* ***Then try this>*** *Build a robot vehicle, using any of the models in the Build section of the LEGO® Education SPIKE™ App. Program it to navigate your directions.* **MORE DETAILS** Model directions in the Build *section of the LEGO® Education SPIKE™ App*, available on the web or downloaded

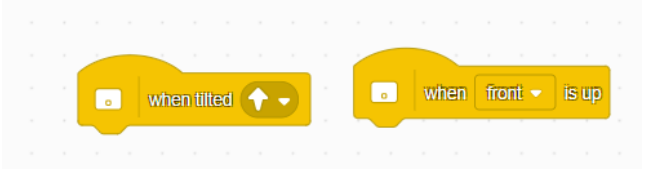
15	**LESSON**	**COMPUTATIONAL THINKING** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. **DESIGN ENGINEERING** **NGSS 3-5 ETS1-2** Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.	• Compare different ways to program the motor blocks to determine which is more efficient.	Have students build the *Taxi! Taxi!* model from the Build section of the LEGO® Education SPIKE™ App. Then ask them to program the model to navigate town in the taxi. Prompt them to try programming each motor with Motor Blocks + More Motor Blocks and then to try using More Movement Blocks to control both motors at once. (As needed, show students how to access the menu of Extension blocks from the + at lower left of the programming canvas.) Have students compare the two coding approaches to determine which is more efficient–fewer blocks, faster/easier for results–to code. **SAY/ASK** *Build a driving car using directions for the* Taxi! Taxi! *lesson. Then program it to navigate your town. Try programming the two motors in two different ways. First, use Motor Blocks and More Motor Blocks to program each one. Then use More Movement blocks to control both motors at once. Which approach is easier? Faster? Works best?* **MORE DETAILS** *Motor Blocks*, *More Motor Blocks*, and *More Movement Blocks* in the Help section of the LEGO® Education SPIKE™ App, available on the web or downloaded; (Access the menu of Extension (More...) Blocks from the + at lower left of the programming canvas.)
16	**PROMPT** **More with Math**	**DECOMPOSITION** **CSTA 1B-AP-11** Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. **SEQUENCES** **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **GEOMETRY** **CCSS.MATH.Content.5.G.A.1** Use a pair of perpendicular number lines, called axes, to define a coordinate system, with the intersection of the lines (the origin) arranged	• Use a coordinate grid to create a map of places to visit. • Program a model car to move from one place to another on the map.	Have students extend their mapping to use a coordinate grid for a map of places to visit in the community. Direct them to place the school at point (0,0) and LEGO® bricks at various other locations as points on the map. Ask students to record the coordinates for each point and then program the driving car (*Taxi! Taxi!* model) to move from point to point. **SAY** *Make a map of the community for the car model to use. Put school at (0,0) on a coordinate grid. Then place LEGO® bricks at different parts of the grid to represent other locations, like the park or grocery store. Record coordinates for each of these points and program the car model to move from one to another.*

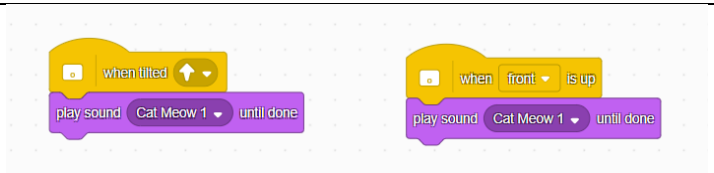
		to coincide with the 0 on each line and a given point in the plane located by using an ordered pair of numbers, called its coordinates. Understand that the first number indicates how far to travel from the origin in the direction of one axis, and the second number indicates how far to travel in the direction of the second axis, with the convention that the names of the two axes and the coordinates correspond (e.g., x-axis and x-coordinate, y-axis and y-coordinate).		

	**Big Little Helper**			
#	**Activity Name**	**Topic(s) & Standards**	**Objectives** Students will	**Prompt**
17 L L	LESSON	**Computational Thinking** **CSTA 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. **Design Engineering** **NGSS 3-5 ETS1-2** Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.	• Create a possible solution to a problem that has constraints. • Improve on others' ideas to develop a new program. • Engage effectively in a range of collaborative discussions.	Have students use build a robot to help Daniel take his things home from school. Then have them program the robot to follow Daniel on his route home. **Say/Ask** *Daniel's locker is overflowing. He needs to clean it out and take things home. How can he get all his things home? Build him a robot helper and program it to follow Daniel on his route home.* **More Details** *Big Little Helper* lesson or access in the LEGO® Education SPIKE™ App
18 L	PROMPT **More with Computer Science**	**Impacts of Computing** **CSTA 1B-AP-19** Brainstorm ways to improve the accessibility and usability of technology products for the diverse needs and wants of users. **Design Engineering** **NGSS 3-5 ETS1-2** Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem	• Identify ways to use programming to increase accessibility for users. • Rebuild and program a robot helper that can carry two people.	Lead discussion about why it's important for technology to meet users' needs, such as to increase accessibility for people that use wheelchairs. Have students rebuild and reprogram Daniel's robot helper so that it can carry Daniel and a friend to the playground. **Say/Ask** *Daniel wants all his friends to be able to use his robot helper. Rebuild and program a new helper that can carry Daniel and a friend to a new playground. Make sure your new robot helper meets the needs of both Daniel and his friends.*

19 (1 hr)	**PROMPT** **Rebuild the World: Food Video Production**	**CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **CSTA 1B-IC-18** Discuss computing technologies that have changed the world, and express how those technologies influence, and are influenced by, cultural practices. **CSTA 1B-IC-20** Seek diverse perspectives for the purpose of improving computational artifacts.	• Research how technology has impacted industries such as video production. • Design and program a tripod to assist a food video producer.	Have students research how computing technologies have changed industries such as video production. Then share with students Ries' Challenge from LEGO Education's Rebuild the World STEAM Heros Career Toolkit for designing a tripod for Food Video Production. Students can use the Big Little Helper model as a springboard for design ideas. As part of the design challenge, have students provide feedback to each other using the criteria outlined in the challenge. Then encourage students to use the peer feedback to improve the design.

	**High-Tech Playground**		
#	**Activity Name**	**TOPIC(S) & Standards**	**Objectives** Students will	**Prompt**
20 (2 hrs)	**LESSON**	**MODIFY PROGRAMS** **CSTA 1B-AP-12** Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features. **DESIGN ENGINEERING** **NGSS 3-5 ETS1-2** Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem.	• Use the design process to improve an existing object. • Develop, test, and refine prototypes as part of a design process. • Engage effectively in a range of collaborative discussions.	Have students build and program a fun seesaw for the playground. Ask them first to program it to rock. Then ask them to make it more exciting for Maria and her friends, perhaps by using the Light Matrix to add light patterns. **SAY/ASK** *Maria is disappointed. There isn't anything for her and her friends to play on during recess. How could they add high-tech features to make the playground more fun? Help Maria design something for her friends! Start with a seesaw and then improve it to be more exciting. What about adding a light show with the Light Matrix?* **MORE DETAILS** *High-Tech Playground* lesson or access in the SPIKE App

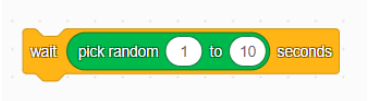
21 🕒	**PROMPT** **More with Math**	**MODIFY PROGRAMS** **CSTA 1B-AP-12** Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features. **PATTERNS AND RELATIONSHIPS** **CCSS.MATH.Content.5.OA.B.3** Generate two numerical patterns using two given rules. Identify apparent relationships between corresponding terms. Form ordered pairs consisting of corresponding terms from the two patterns, and graph the ordered pairs on a coordinate plane. For example, given the rule "Add 3" and the starting number 0, and given the rule "Add 6" and the starting number 0, generate terms in the resulting sequences, and observe that the terms in one sequence are twice the corresponding terms in the other sequence. Explain informally why this is so.	• Develop a numerical pattern. • Program a seesaw model to move according to the numerical pattern.	Prompt students to create a pattern in the movement of the seesaw, using numerical patterns such as starting at 0 and then adding 2. Have them pick a pattern to program the movement of the seesaw. For example, it would move 2 times, then pause, then move 4 times, then pause, then 6 and so on. **SAY/ASK** *Think about the way a seesaw moves up and down. What could make that more exciting? Create a numerical pattern, such as starting at 0 and adding 2. Brainstorm some ideas, then choose one and program the movement of the seesaw to reflect it.*
22 🕒	**PROMPT** **More with the Gyro Sensor**	**CSTA 1B-DA-07** Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea.	• Collect data to predict an outcome. • Describe the difference between two event blocks using the same sensor.	Ask students to open a new SPIKE Essential word block program from the app. Have students find the following program blocks and drag them into the programming in the scripts area.  Have students predict what they think the difference is between these programming blocks. Have students connect the hub to the program to investigate how these blocks function. Have students add a play sound block for testing.

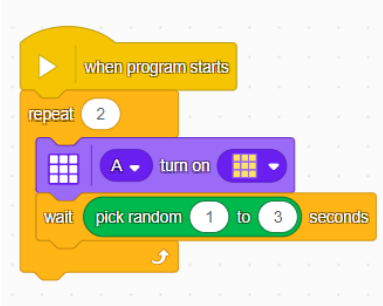
				 Have students run the program and collect observational data for similarities and differences between the two events. Have students change the direction for each block, predict an outcome, then test. Students should continue collecting data with each iteration. Discuss findings as a class and when it might be beneficial to use one event instead of another within a program. **MORE DETAILS** Information about the When tilted and When Hub Orientation is Up programming blocks can be found in the Help section of the SPIKE App
23 🕒 🕒	**LESSON: GRAVITY**	**CSTA 1B-AP-15** Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.	• Build a device that demonstrates that the gravitational force exerted by Earth pulls objects down, toward Earth's center. • Use the device to explain the bird's behavior in terms of the downward force of gravity.	Have students complete the Gravity lesson for SPIKE Essential. Have students explain the intended outcome for the programs, using the When tilted and When hub orientation is up programming blocks. **MORE DETAILS** Gravity lesson can be found in the unit Science We Cannot See.

	**Trash Monster Machine**			
#	**Activity Name**	**Topic(s) & Standards**	**Objectives** Students will	**Prompt**
24	**Prompt** **More with the Color Sensor**	**Data & Analysis** **CSTA 1A-DA-06** Organize and present collected data visually to highlight relationships and support a claim. **Math Skill Practice: Bar Graphs** Prepare for the *Trash Monster Machine* lesson.	• Use the Color Sensor and word blocks to create a bar graph that shows the colors detected by the Sensor.	To prepare for the *Trash Monster Machine* lesson, have students explore ways to create a bar graph. Prompt them to plug the Color Sensor into the hub and create a program that will track the different colors detected in a bar graph. **Say/Ask** *Practice using the Color Sensor to control aspects of your program. Plug it into your hub. Create a program that will track the different colors that the Sensor detects on a bar graph. (Use Bar Graph Blocks.)* **More Details** *Sensor* and *Bar Graph Blocks* in the Help section of the LEGO® Education SPIKE™ App, available on the web or downloaded. (For connection steps, review *The Color Sensor* tutorial in the START section.)
25	**Lesson**	**Develop Programs** **CSTA 1B-AP-13** Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences. **Design Engineering** **NGSS 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.	• Explore the benefits of automated solutions. • Refine a prototype as part of a cyclical design process. • Engage effectively in a range of collaborative discussions.	Have students build Sofie a machine to collect trash at her school. Then prompt them to program it so everyone at school will want to use it for all kinds (colors) of trash. If you wish, have students use the Bar Graph blocks in their SPIKE App to count trash. **Say/Ask** *Sofie has noticed that some of her friends don't throw out their trash after lunch. Help Sofie create a new way for her friends to throw out their trash. What could make everyone want to use the machine? Design, build, and program a machine to collect all the trash at school, no matter what color it is. Then improve it to work even better. Try using Bar Graph blocks to graph the colors.* **More Details** *Trash Monster Machine* lesson or access in the SPIKE App; *Bar Graph Blocks* in the Help section of the App, available on the web or downloaded.

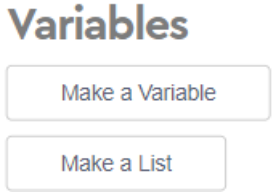
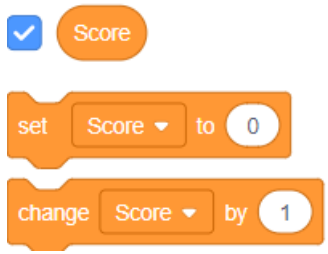
26	**PROMPT** **More ELA and Computer Science**	**SEQUENCES/LOOPS** **DEVELOP PROGRAMS** **CSTA 1B-AP-13** Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences. **CSTA 1B-AP-12** Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features. **INFORMATIVE WRITING** **CCSS.ELA-Literacy.W.5.9.B** Apply grade 5 Reading standards to informational texts (e.g., "Explain how an author uses reasons and evidence to support particular points in a text, identifying which reasons and evidence support which point[s]"). **READING INFORMATIONAL TEXTS** **CCSS.ELA-Literacy.RI.5.9** Integrate information from several texts on the same topic in order to write or speak about the subject knowledgeably.	• Research in several texts to learn about ways countries sort and dispose of trash. • Integrate information from research to redesign and reprogram a model.	Have students conduct a short research project comparing how two different countries sort and dispose of their trash. Prompt them to use the information to redesign and program their trash monster to meet the needs of people living in a country that they researched. **SAY** *Research to learn how two countries sort and dispose of trash. Use the information to redesign and program your trash monster machine to meet the needs of people in one of the two countries.*

	**Winning Goal**			
#	**Activity Name**	**Topic(s) & Standards**	**Objectives** Students will	**Prompt**
27 🕒	**Prompt** **More with Math**	Prepare to create bar graphs in the *Winning Goal* lesson. **Data & Analysis** **CSTA 1A-DA-06** Organize and present collected data visually to highlight relationships and support a claim. **Math Skill Practice: Bar Graphs** **CCSS.MATH.Practice.K-12.MP.5** Use appropriate tools strategically.	• Use programming tools to show data in bar graph form.	To prepare students for *Winning Goal*, have them practice using the Bar Graph feature in the SPIKE App. Prompt them to use the Bar Graph Blocks to graph data of their choosing or that you supply. **Say/Ask** *Practice using the Bar Graph feature in your SPIKE App more. What else can it do? How can you use it to graph data? Try some different ways. Share with the class.* **More Details** *Bar Graph Blocks* in the Help section of the LEGO® Education SPIKE™ App, available on the web or downloaded.
28 🕒🕒	**Lesson**	**Computational Thinking** **CSTA 1B-AP-11** Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process **Design Engineering** **NGSS 3-5-ETS1-3** Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved.	• Identify the failure points of a model or program. • Consider failure points in order to make improvements. • Engage effectively in a range of collaborative discussions.	Lead discussion about what makes computer games fun. Have students build a soccer game for Maria, applying some of the ideas generated by brainstorming. Prompt them to program the game to move the goal, iterating to find even wackier ways to move it around. **Say/Ask** *Maria isn't having fun playing soccer. She prefers the wackiness of her computer games. How can Maria's soccer game be more like a computer game? Brainstorm ideas. Then build and program a game for Maria. Make it move the goal around. See how much fun you can add!* **More Details** *Winning Goal* lesson or access in the SPIKE App
29 🕒	**Prompt** **More with Math and Computer Science**	**Math Skill Practice: Using Graphs** **CCSS.MATH.Practice.K-12.MP.5** Use appropriate tools strategically. **Data & Analysis** **CSTA 1A-DA-06** Organize and present collected data visually to highlight relationships and support a claim.	• Program to create a bar graph of points scored in a game.	Prompt students to add the Color Sensor to the goal machine and create a graph of the points scored. Each time a goal is scored students can hold up a brick to the sensor to keep points. As needed, show students how to include the Bar Graph extension in their programming blocks. (Available by clicking the small + at lower left of the programming canvas.)

				SAY/ASK *Add the Color Sensor to your model and program it to count the points scored on Maria's machine. Hold up a brick to keep points. Then make a bar graph of the data in your SPIKE App. You'll need to add the Bar Graph extension to your blocks. See the small + at the bottom left of the programming canvas.* **MORE DETAILS** *Bar Graph Blocks* in the Help section of the SPIKE App, available on the web or downloaded.
30	**PROMPT: Operators**	**SEQUENCES** **CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals.	• Create a simple program to describe the function of pick random block. • Describe the function of reporter blocks.	Have students attach the light matrix to the SPIKE Essential hub. Then launch a new word block program from the app. Explain to students that you would like for them to create a simple program where the light matrix will come on and stay on for 1 to 3 seconds. How can you create a program that can randomly select the amount of time the light matrix is displayed? Have students explore the green operator blocks to see if they can find a block that will select a random number.  Share with students that this block type is called reporter. Reporter blocks are round on the ends and be inserted into holes of other blocks with the same shape. These block types can hold numbers or something we call a string, which is a sequence of computer characters (like text) including spaces. Demonstrate for students how the pick random block can fit inside the control block called Wait for Seconds  Have students describe the outcome of this block. Next, have students create a program based on the following pseudocode:

				When program starts, Turn on the light matrix to all yellow Wait for one to three seconds Then repeat two times.  Have students discuss other ways to use the pick random block. When might this block be useful when creating a program?

	**Literary Randomizer**			
#	**Activity Name**	**Topic(s) & Standards**	**Objectives** Students will	**Prompt**
31 Lesson	Lesson	**Design Engineering** **NGSS ETS1-2** Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem. **Sequence/Loops** **CSTA 3-5 1B-AP-10** Create programs that include sequences, events, loops, and conditionals.	• Define success criteria to help evaluate a solution. • Compare and contrast different solutions to determine which one meets the specified need. • Engage effectively in a range of collaborative discussions	Have students build a machine to help Daniel decide what kind of books to read. Brainstorm some genres of books, like mysteries, how-to, and science fiction. After they build the machine, prompt students to program it to select book genres randomly. **Say/Ask** *Daniel can't decide which book to read. He feels overwhelmed by all the books in the library. How can Daniel's literary randomizer make picking a book more exciting? Build a machine for him. Program it to choose the kind of book randomly and surprise Daniel! He can try many kinds of books to choose the ones he likes best.* **More Details** *Literary Randomizer* lesson or access in the LEGO® Education SPIKE™ App
32	Prompt **Literary Remix Challenge**	**CSTA 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **CSTA 1B-AP-12** Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features. **CSTA 1B-AP-14** Observe intellectual property rights and give appropriate attribution when creating or remixing programs.	• Remix a model for a new purpose • Observe intellectual property rights and give attribution when remixing a program.	Challenge students to remix the Literary Randomizer model for a new purpose! Follow the guidelines for the Literary Randomizer Menu Remix Challenge to create a menu selector for a holiday season meal. Have students write a paragraph to accompany the new model, explaining how it functions. Discuss why it would be appropriate to give attribution to LEGO Education and the Literary Randomizer model as inspiration for the new build.

3 **3** 🕒 🕒	**PROMPT:** **Variables in a Game**	**CSTA 1A-AP-09** Create programs that use variables to store and modify data.	• Create a variable to keep track of a score	Spark a discussion around the term variable. Explain to students that a variable represents something that change over time. Sometimes programmers need data to be stored, displayed, manipulated or run within a program. So a variable is named storage location in a program to hold data. In the SPIKE App, we have a way to create a variable. One way the variable blocks are used is to create a way to keep score in a game. A score is something that can change and may need to be reset. Have students launch the SPIKE App and open a new project. Have students locate the variable options from the block palette.  Have students click on the block "Make a Variable" to create a new variable. Name this variable score. Once the variable is created, new programming blocks are displayed in the palette.  Challenge students to see if they can create a game using the gyro sensor on the hub that will keep score. Have students share their games and how they used the variable blocks in their programs.

		**Your School Creation**		
#	**Activity Name**	**Topic(s) & Standards**	**Objectives** Students will	**Prompt**
34 L L L	**Lesson**	**Design Engineering** **NGSS 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. **NGSS 3-5-ETS1-2** Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of the problem. **NGSS 3-5-ETS1-3** Plan and carry out fair tests in which variables are controlled and failure points are considered to identify aspects of a model or prototype that can be improved. **Computational Thinking** **CSTA 3-5 1B-AP-08** Compare and refine multiple algorithms for the same task and determine which is the most appropriate. **CSTA 3-5 1B-AP-11** Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. **Sequence/Loops** **CSTA 3-5 1B-AP-10** Create programs that include sequences, events, loops, and conditionals. **Develop Programs** **CSTA 3-5 1B-AP-13** Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.	• Apply engineering design skills in order to solve a problem. • Practice brainstorming as part of the design process. • Engage effectively in a range of collaborative discussions.	Have students consolidate their learning from the *Quirky Creations* activities to design, build, and program a model that will help the Spike Team during the school day. **Say** *The team thinks there's a way to bring a new creation to their classroom. It's time to create your very own invention for the team's classroom! Use at least one motor or sensor!* **More Details** *Your School Creation* lesson or access in the LEGO® Education SPIKE™ App

		CSTA 3-5 1B-AP-17 Describe choices made during program development using code comments, presentations, and demonstrations. **MODIFY PROGRAMS** **CSTA 3-5 1B-AP-12** Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.		
35 **PROMPT** **More with ELA**		**CSTA 1B-AP-14** Observe intellectual property rights and give appropriate attribution when creating or remixing programs. **INFORMATIVE WRITING** **CCSS.ELA-Literacy.W.5.1** Write opinion pieces on topics or texts, supporting a point of view with reasons and information.	• Write an advertisement for the school creation models the class has built. • Include an opinion on which one is the best creation. • Include reasons and information to support your opinion.	Have students create advertisements for their new school creations. They should include an opinion on why their creation is the best one for the team's classroom. Have students observe intellectual property rights and give appropriate attribution, if needed, for models or programming used in their new school creation. **SAY/ASK** *Think about all the creations the class designed. Which one is the best? Create an ad for some or all the creations. Include an opinion on why your creation is the best one. Support the opinion with information and reasons.*