wonder workshop

dash & dot™

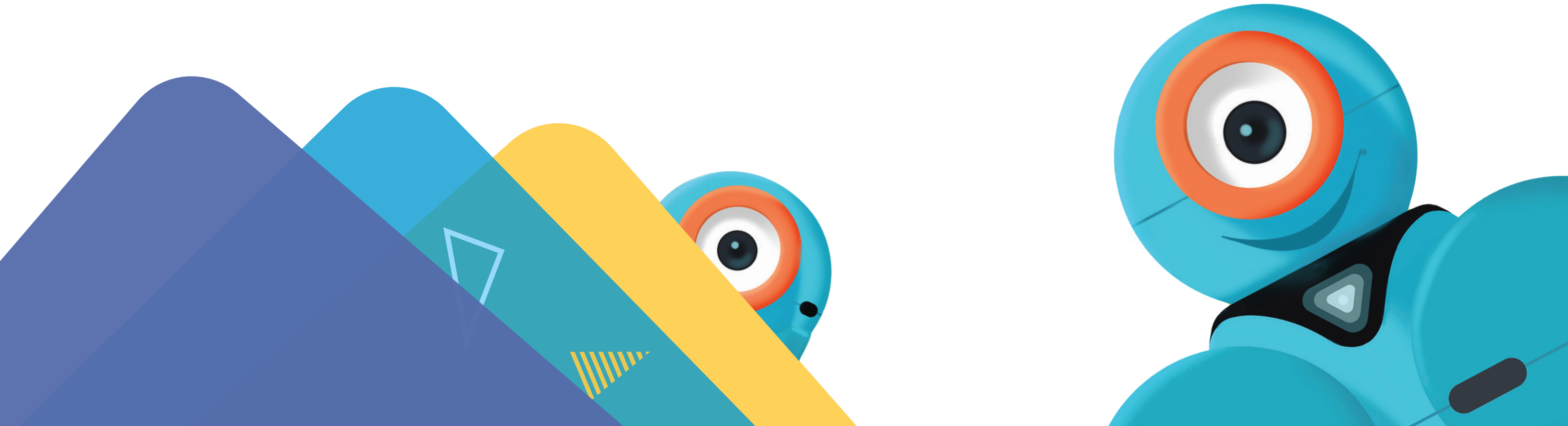
challenge cards

Dash the Collector (B 1.1 - B 1.3): Teacher Packet

In this packet, we've included resources that will help you and your students as they independently complete the **Dash the Collector Challenge Card** set:

- **Challenge Cards**: a set of 3 **Challenge Cards** for students to practice coding concepts
- **Solution Guides**: hints, suggestions, discussion questions, and cross curricular extension activities for each card
- **Worksheets and Resources**: implementation strategies, planning/reflection worksheets for students, and an evaluation worksheet
- **Lesson Plan (optional)**: a whole-class instruction lesson focusing on the coding concept(s) that students will practice while completing the Challenge Cards

Looking for More? Visit: www.education.makewonder.com



B | 1.1 | Sequences

Dash the Collector

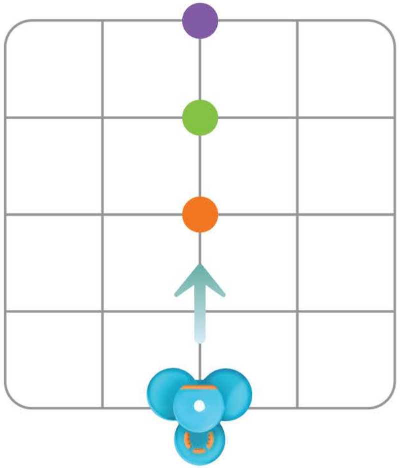
Dash is at the beach and sees
a lot of beautiful seashells.
Let's help Dash collect some!

wonder
workshop

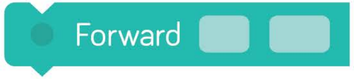
B | 1.1 | Sequences

Materials: 3 seashells or small toys

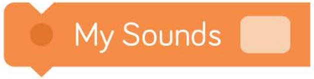
1. Put the **Bulldozer** on Dash.
2. Place **3 seashells** or **small toys** in a **straight line** on the floor.



3. Use **1 Forward** block to help Dash **collect** all the seashells.



4. **Record** a "Whoopee!" **sound** using the **My Sounds** block to help Dash celebrate.

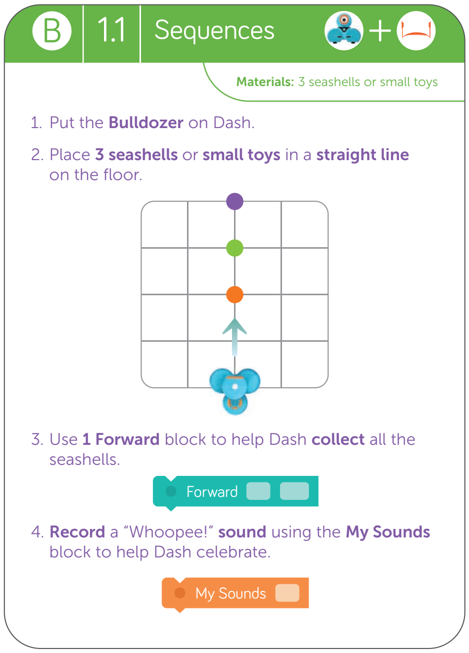


Dash the Collector

Time: 10 minutes

Hints

- If you don't have a **Bulldozer**, use the **Building Block Connectors** to create your own.
- Tap the **Forward** block and drag the **arrow up** or **down** to change how far Dash moves.
- You can record your own sound by tapping on the **My Sounds** block. Then press **Record a New Sound**, choose a number slot, and tap the microphone to record your sound.



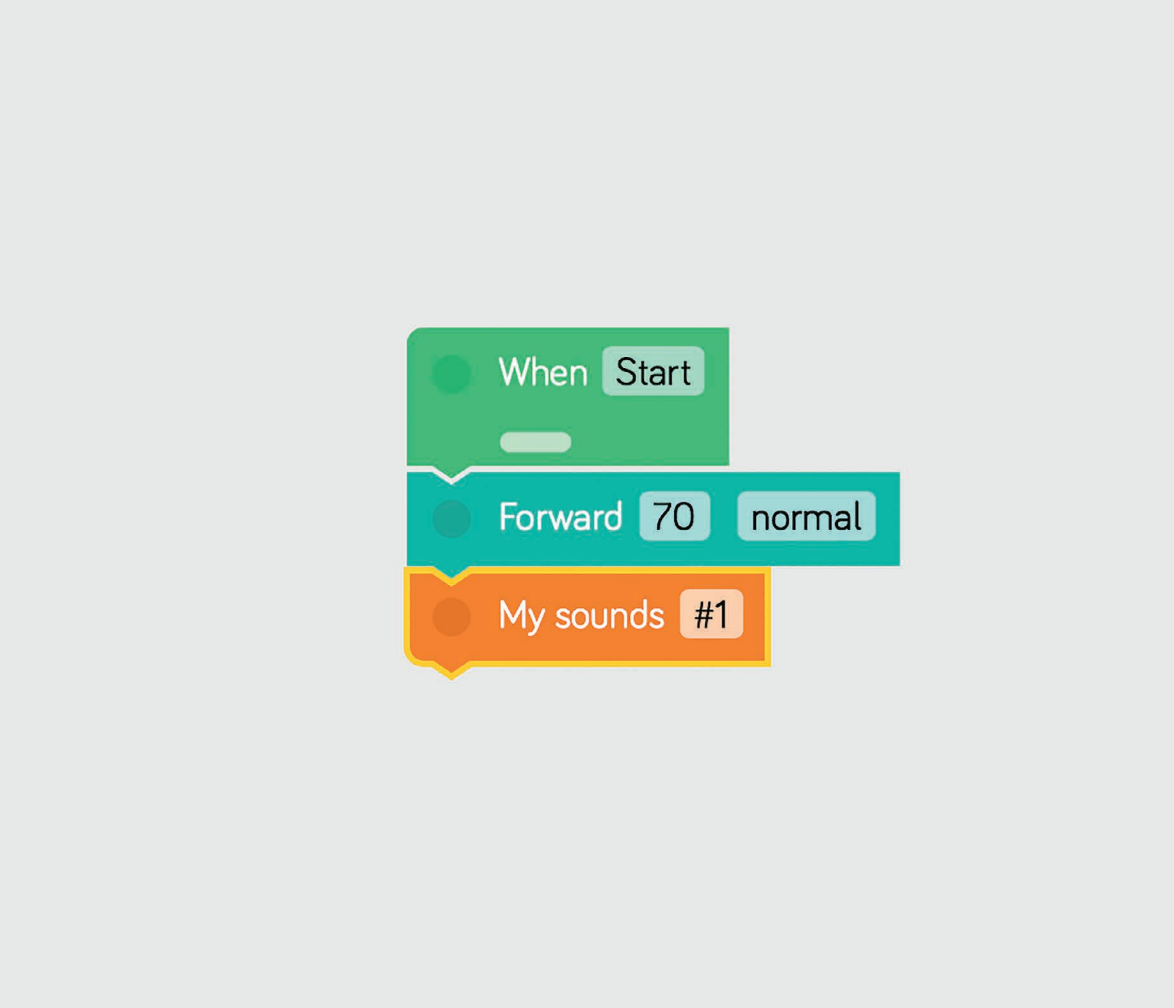
B | 1.1 | Sequences

Materials: 3 seashells or small toys

1. Put the **Bulldozer** on Dash.
2. Place **3 seashells** or **small toys** in a **straight line** on the floor.
3. Use **1 Forward** block to help Dash **collect** all the seashells.

4. **Record** a "Whoopee!" **sound** using the **My Sounds** block to help Dash celebrate.

Suggested Solution:



```
When Start
Forward 70 normal
My sounds #1
```

Discussion Questions

1. How would you need to change the program if there were **5 seashells** instead of 3?

2. What if you wanted Dash to make a sound after each seashell was collected? What would you need to add to the program? What would you need to change?

Cross-Curricular Connections

MATH

- Have students measure out the distance between each seashell before they decide how far Dash should move forward. (CCSS.MATH.MD.A.2)

ELA

- Have students complete the challenge using a grid on the floor (e.g., gridded mat or a grid made out of masking tape). Then have them turn the grid into a map, adding features like a key and compass. (CCSS.ELA.RI.K.7)

- Have students write their own version of the challenge using a different story and objectives. (CCSS.ELA.W.1.7)

NOTES:

B | 1.2 | Sequences

It's Candy Time!

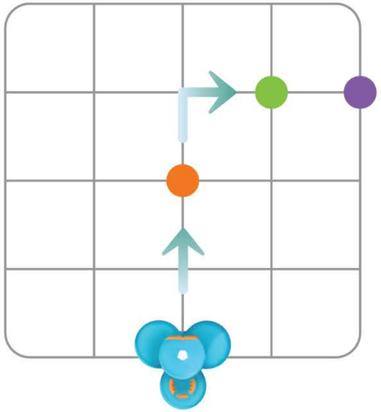
The party piñata has spilled candy all over the floor! Help Dash collect some candy, and you may just get a treat!

wonder workshop

B | 1.2 | Sequences

Materials: 3 pieces of candy or small toys

1. Put the **bulldozer** on Dash.
2. Place **3 pieces of candy or small toys** on the floor like in the picture below.



3. Use **Forward** and **Turn Right** blocks to help Dash collect all the candy.

Forward   Turn Right

4. Then **record** a "Yum, yum, yum!" **sound** using the **My Sounds** block to play after Dash collects all the candy.

My Sounds

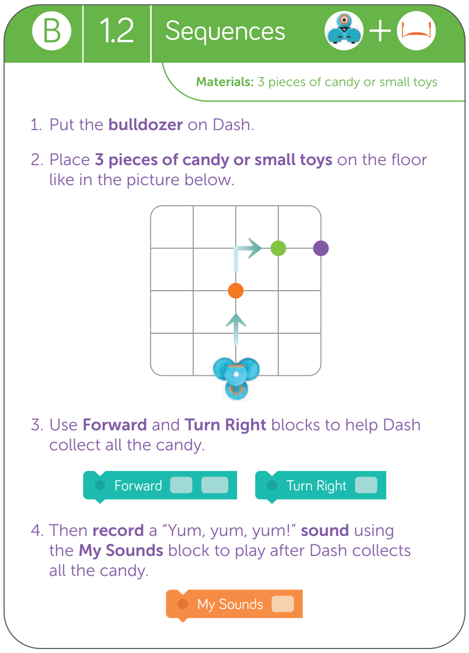
B | 1.2

It’s Candy Time!

Time: 15 minutes

Hints

- If you don’t have a **Bulldozer**, use the **Building Block Connectors** to create your own.
- Use your finger to trace the path that you want Dash to take. What’s the first block you’ll need to get Dash started on the path? Is it a **Forward** block or a **Turn Right** block?
- You can **record** your own sound by tapping on the **My Sounds** block. Then press **Record a New Sound**, choose a number slot, and tap the microphone to record your sound.



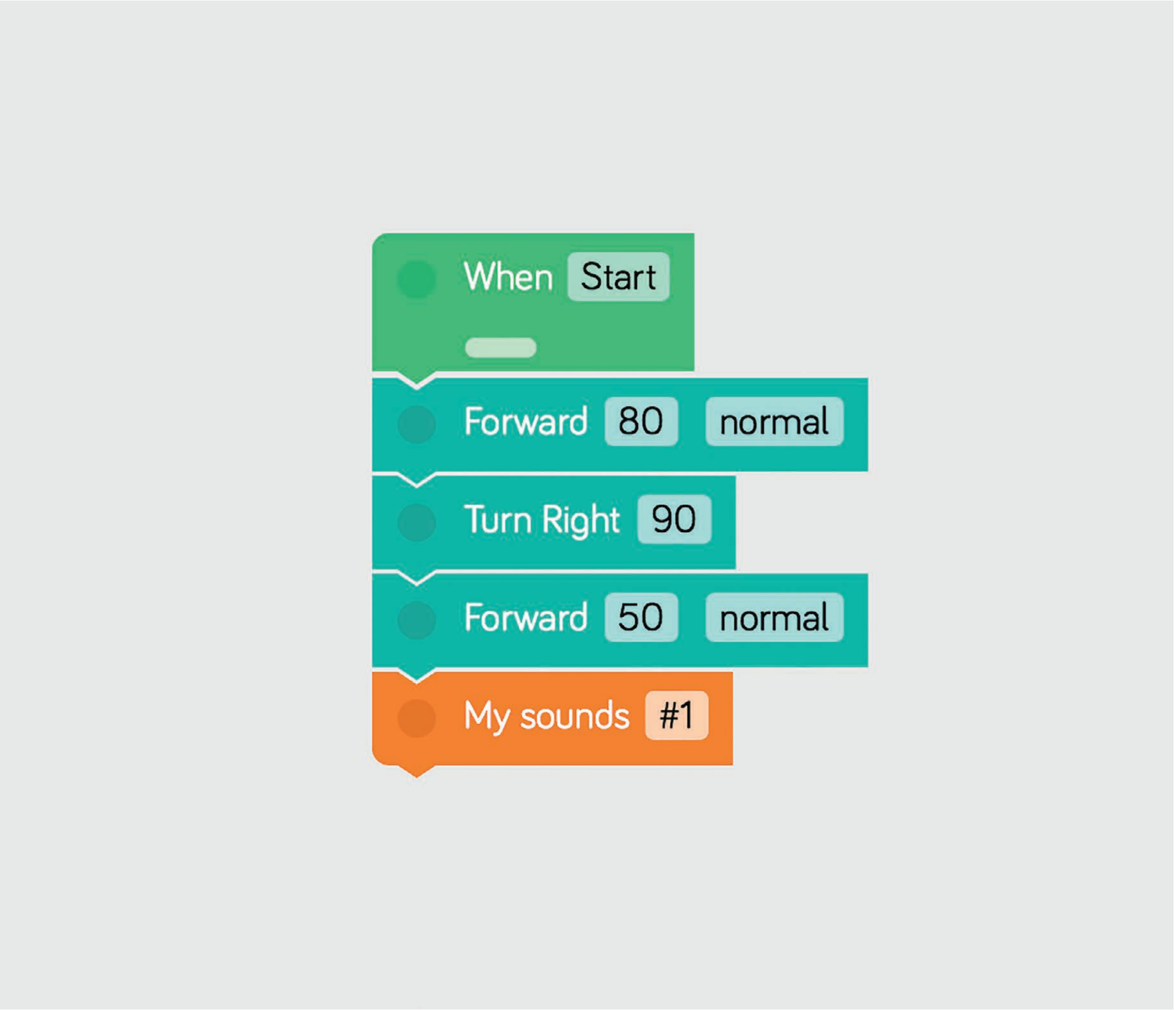
B | 1.2 Sequences

Materials: 3 pieces of candy or small toys

1. Put the **bulldozer** on Dash.
2. Place **3 pieces of candy or small toys** on the floor like in the picture below.
3. Use **Forward** and **Turn Right** blocks to help Dash collect all the candy.
4. Then **record** a “Yum, yum, yum!” **sound** using the **My Sounds** block to play after Dash collects all the candy.

Forward
Turn Right
My Sounds

Suggested Solution:



```
When Start
Forward 80 normal
Turn Right 90
Forward 50 normal
My sounds #1
```

Discussion Questions

1. How would you need to change the program if Dash were facing a different direction at the start of the path?

2. What if you wanted Dash to make a sound after each piece of candy was collected? What would you need to add to the program? What would you need to change?

Cross-Curricular Connections

MATH

- Have students measure out the distance between each piece of candy before they decide how far Dash should move forward. (CCSS.MATH.MD.A.2)

ELA

- Have students complete the challenge using a grid on the floor (e.g., gridded mat or a grid made out of masking tape). Then have them turn the grid into a map, adding features like a key and compass. (CCSS.ELA.RI.K.7)

- Have students write their own version of the challenge using a different story and objectives. (CCSS.ELA.W.1.7)

NOTES:

B | 1.3 | Sequences

Egg Help!

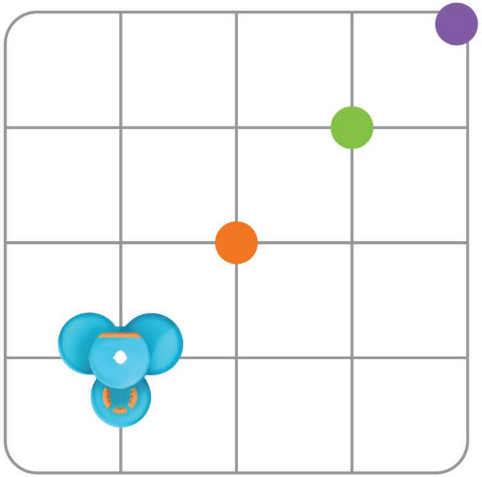
Oh no! Some eggs fell out of a bird's nest. Let's use Dash to help collect all of the eggs.

wonder workshop

B | 1.3 | Sequences

Materials: 3 small toys

1. Put the **Bulldozer** on Dash.
2. Place **3 small toys** on the floor like in the picture below. These are the bird's eggs.



3. Program Dash to **move** and **collect** the 3 eggs.

BONUS Add more eggs for Dash to collect.

Egg Help!

Time: 15 minutes

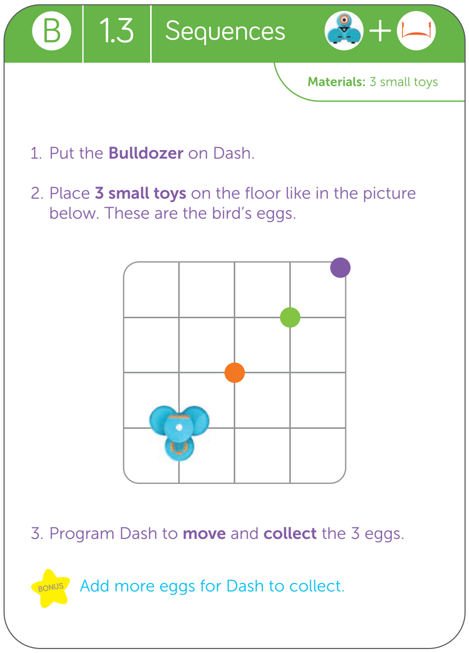
Hints

- On a piece of paper, draw the path that you want Dash to take. What blocks could you use to make Dash move that way?

- Dash needs to use **Forward** and **Turn Right** blocks. In order to collect all the eggs, how far forward does Dash need to move? How much does Dash need to turn?

- Use tape to mark Dash's starting spot and each toy's location. Remember to put Dash back at the starting spot each time you start the program.

- If you don't have a **Bulldozer**, use the **Building Block Connectors** to create your own.



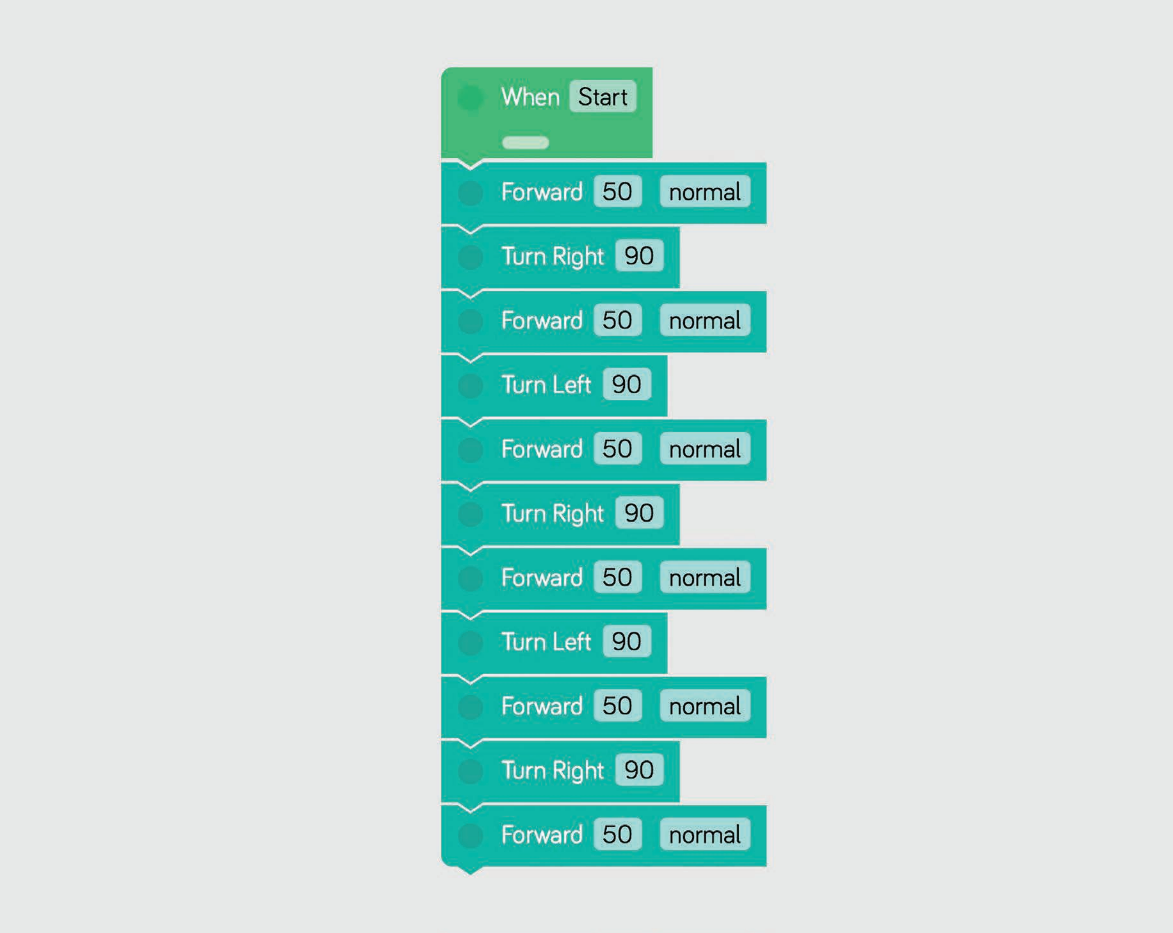
B | 1.3 | Sequences

Materials: 3 small toys

1. Put the **Bulldozer** on Dash.
2. Place **3 small toys** on the floor like in the picture below. These are the bird's eggs.
3. Program Dash to **move** and **collect** the 3 eggs.

BONUS Add more eggs for Dash to collect.

Suggested Solution:



```
When Start
Forward 50 normal
Turn Right 90
Forward 50 normal
Turn Left 90
Forward 50 normal
Turn Right 90
Forward 50 normal
Turn Left 90
Forward 50 normal
Turn Right 90
Forward 50 normal
```

Discussion Questions

1. What are some different ways Dash can collect the eggs?

2. What is the fewest number of blocks you can use to collect all the eggs?

3. What was the hardest part of the challenge?

Cross-Curricular Connections

MATH

- Have students complete the challenge using a grid on the floor (e.g., gridded mat or a grid made out of masking tape). Then have students measure out the distance between each egg before they decide how far Dash should move forward. (CCSS.MATH.1.MD.A.2)

- Have students add up how many centimeters Dash traveled in the entire challenge. (CCSS.MATH.1.OA.A.1)

ELA

- Have students complete the challenge using a grid on the floor (e.g., gridded mat or a grid made out of masking tape). Then have them turn the grid into a map, adding features like a key and compass. (CCSS.ELA.RI.K.7)

- Have students write their own version of the challenge using a different story and objectives. (CCSS.ELA.W.1.7)

NOTES:

wonder workshop

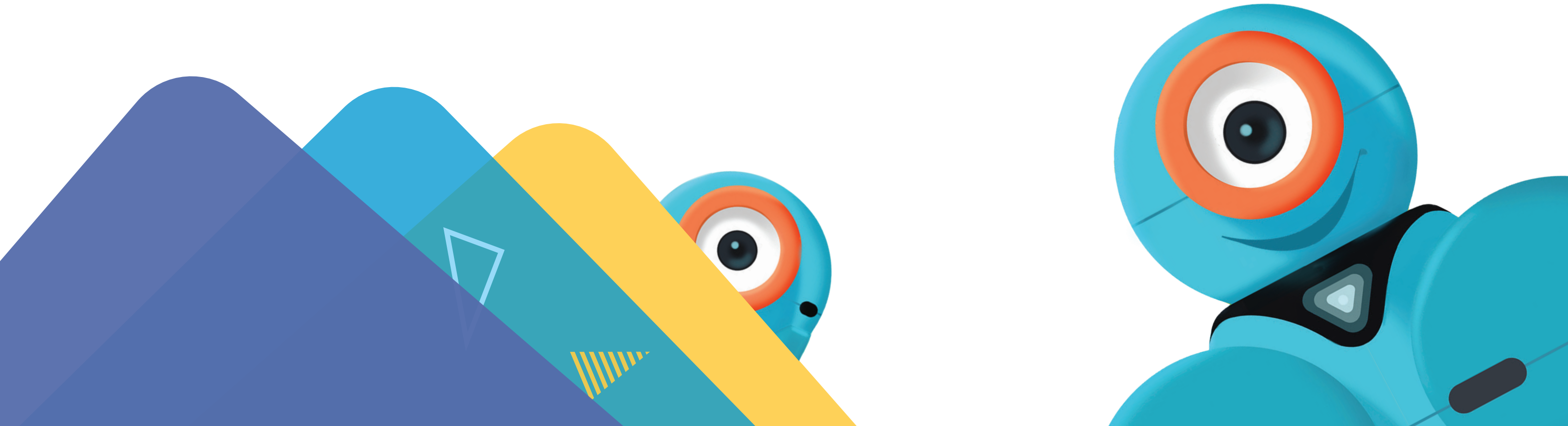
dash & dot™ challenge cards

Worksheets & Resources

In this section, you will find the following worksheets/resources:

- Challenge Card Tips & Tricks
- Planning Worksheets
- Reflection Worksheets
- Troubleshooting Strategies
- Problem Solving & Debugging Strategies
- Evaluation Rubric
- Glossary

Looking for More? Visit: www.education.makewonder.com



Challenge Card Tips & Tricks

Determine Team Roles

Swap roles with your teammates for each challenge. Team roles include lead programmer, robot wrangler, and documentarian.

Plan Your Path

Draw out the path you want Dash to follow. Then plan out the blocks you'll need. You can also get up and walk the path that you think Dash should take.

Mark Your Spots

Use tape to mark Dash's starting spot and the location of any obstacles/objects.

Go Back to Start

Always put Dash back at the starting spot before playing a program again.

Use the When Start Block

Place your blocks under the **When Start** block. The **When Start** block should always be on your screen.

Think in Centimeters

Dash moves in centimeters. A centimeter is about the width of your finger.

Check Off the Steps

Use a dry erase marker to check off each step as you complete it. Make sure you erase the marks after you're done.

Help Your Robots Hear You

If the classroom is noisy, use the **Hear Clap** cue instead of the **Hear Voice** cue. Ask the teacher if you may try out your program with Dash and/or Dot outside or in the hallway.

Set a Time Limit

Give yourself or your team a set amount of time in which to complete the challenge

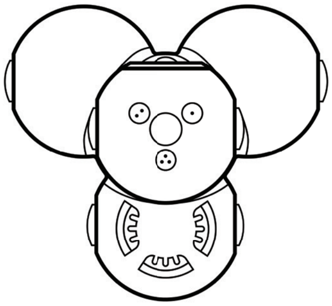
Dash Planning Worksheet

Name(s): ________________________________ Date: ________________

Coding Level: ________ Card #: ________

What do you want Dash to do?

Draw out the steps of the challenge or write a few sentences describing your goal.



General Planning Worksheet

Name(s): ______________________________ Date: ________________

Coding Level: _______ Card #: _______

1. What do you want Dash or Dot to do?

Draw out the steps of the challenge or write a few sentences describing your goal.

2. What will you do to achieve your solution?

What will each team member do? What steps will you need to take? What blocks will you use?

Reflection Worksheet

Name(s): ________________________________ Date: ____________________

Coding Level: __________ Card #: __________

1. What did Dash and/or Dot do when you ran your program?

2. Did you make any mistakes? If so, how did you fix them?

Advanced Reflection Worksheet

Write a reflection entry in your Wonder Journal. Try to answer these questions as part of your reflection:

Results

- What did Dash and Dot do when you ran your program?
- Did you make any mistakes? If so, how did you fix them?

Connections

- What did you like the most about this challenge? Why?
- What was the most difficult part of the challenge? What did you learn from it?

Next Steps

- If you had more time, how would you change or add to your code?
- What are you planning to do next? Will you try another Challenge Card or start a new coding project?

Troubleshooting

If your program is not running correctly . . .

- Check if Dash and/or Dot are turned on.
- Make sure Dash and/or Dot are connected to the app.
- Make sure your blocks are connected to the **When Start** block.
- Try restarting the app.

If Dash and/or Dot are disconnecting . . .

- Turn off the robots and turn them on again. Then reconnect the robots to the app.
- Press play and then press stop to make the robots reset.
- Try charging the robots.

Three, then me!

- Ask or get help from three of your classmates. If you still need help, then ask the teacher.

Problem Solving & Debugging

Break down the challenge

- What do you need for the challenge? Which robots? Which materials and/or accessories?
- What are Dash and/or Dot supposed to do?
- Have you solved similar challenges to this one?
- Focus on one step at a time.

Plan your solution

- Draw a picture or make a list of what you want Dash or Dot to do.
- What blocks will you need to complete the challenge?
- Are there any hints on the card that can help?
- Use tape to mark Dash's starting point.
- Use tape to mark each obstacle's location.

Test Your Code

- Does your code complete the challenge?
- If not, play your code again. Watch as the program goes through each block. Do you notice any mistakes?
- Do you need to change, delete, or add more blocks?
- Are your blocks telling Dash to do something when you actually want Dot to do something?

Improve your work

- Ask another student or group to check your program.
- Is there an easier way to complete the challenge? Can you use fewer blocks?
- How can you improve your program? Could you add more lights, sounds, or other customizations?

Evaluation Rubric

	Programming	Reflection & Documentation	Collaboration & Communication	Creativity
1 Novice	Completed part of the activity and needed assistance throughout the process.	Use a journal, worksheets, and/or multimedia tools (such as video and images) to document some of the activity results.	Participated little or not at all in classroom discussions. Demonstrated little to no cooperation with group members during the activity.	Demonstrated limited creativity in developing ways to complete the activity.
2 Developing	Used the targeted coding concept(s) to complete the activity with some assistance.	Incorporated some target vocabulary and some thoughtful reflection on the coding process while documenting activity results using journal entries and multimedia tools.	Occasionally participated in classroom discussions and cooperated somewhat with group members.	Developed a few different ways to complete the activity, but the solution was not particularly creative.
3 Proficient	Used the targeted coding concept(s) to complete the activity without assitance.	Incorporated target vocabulary and reflection on the coding process. Clearly documented activity results using journal entries and multimedia tools.	Actively participated in classroom discussions. Answered questions and cooperated with group members during the activity.	Applied the iterative process to develop creative and unexpected solutions for the activity.
4 Exemplary	Used the targeted coding concept(s) to complete the activity without assistance. Enhanced the solution with more efficient (e.g., fewer blocks) and/or advanced features (e.g., lights, sounds) in the code.	Incorporated advanced target vocabulary and in-depth reflection on the coding process. Thoroughly and clearly documented and presented activity results.	Actively participated in classroom discussions and cooperated with group members. Gave constructive feedback to others and effectively incorporated feedback from others.	Went above and beyond to develop, revise, and execute imaginative solutions for the activity.

wonder workshop

dash & dot™ challenge cards

Coding Lessons

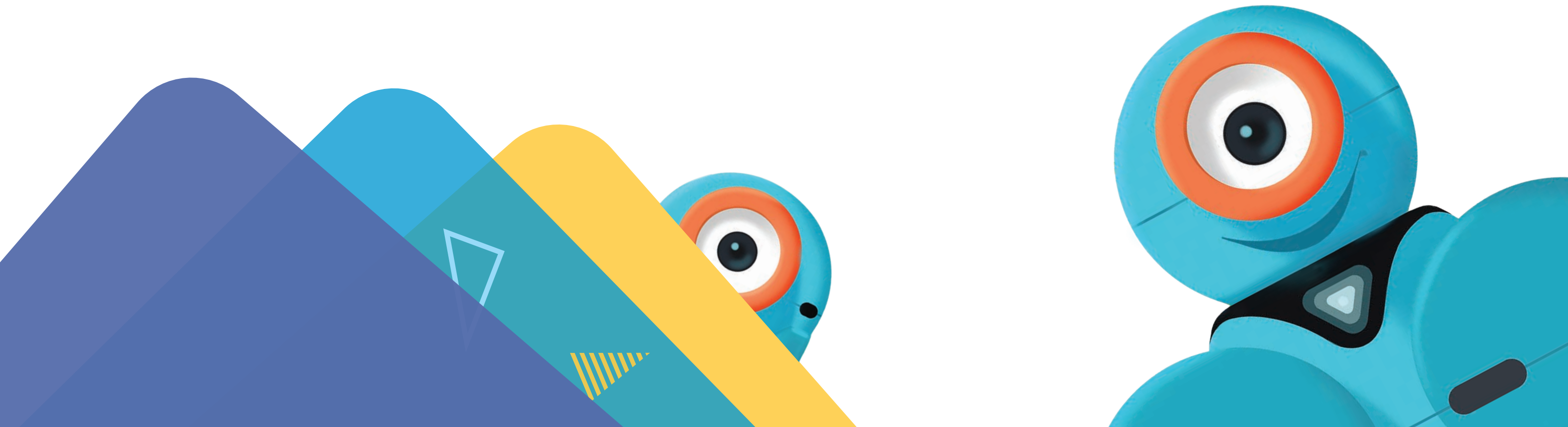
The following lesson plan is a part of our **Learn to Code Curriculum**.
To check out more free coding lessons, please visit:

www.education.makewonder.com/curriculum/learn_to_code

Cross Curricular Lessons

Looking to go beyond the **Hour of Code**? Your students can **code to learn** via our project-based cross curricular lessons!

www.education.makewonder.com/curriculum/code_to_learn



Level B: Lesson 1

Sequencing: Changing Parameters

Lesson Information

Overview/Description
Students will review their previous programming with **sequencing** and **loops.**

Students will learn how to change the **parameters** (distances, speeds, etc.) in programs.

They will use *Blockly* preset programs and **Challenge Cards** to practice their new skills.

Coding Level: B

Prior Experience:
Before this lesson, it is recommended that students complete:

- Level A: Lessons 1–5
- [Optional] Level A **Challenge Cards**: A1.1–3.6

Learning Objectives:
Students will:

- Understand what problems are best solved with **sequencing**.
- Change the **parameters** in a program.
- Use **sequencing** to revise code and complete coding challenges in an iterative process.

Target Grade Range: K–1

Suggested Group Size: 2–3 students per robot

Time Required: 45–60 minutes

Materials:

- 1 Dash robot per group
- 1 tablet per group
- projector or interactive display with mirroring capability
- pencils
- [Optional] **Challenge Cards**: B 1.1, B 1.2, B 1.3
- [Optional] access to **Twitter** and **Instagram**

Resources/Downloads:

- **Troubleshooting** handout
- **Problem Solving & Debugging** handout
- **Wonder Journal: Dash Planning** and/or **General Planning** worksheets
- **Wonder Journal: Reflection** worksheets
- **Evaluation Rubric**
- [Optional] **Challenge Card Checklist**
- [Optional] **Challenge Card Tips & Tricks** handout

Preparation:

- Fully charge the tablets and Dash robots.
- Install the **Wonder Workshop** *Blockly* app on each tablet.

Review

SEQUENCE AND LOOP REVIEW

1. Briefly review how students used **sequences** and **loops** with Dash during the previous lesson by asking: "What sequences and loops have you created with Dash so far?"
   - Sample responses: "We programmed Dash to use sounds and motions to help Dash act like a guard. We used a **Repeat Forever** block."
2. Ask, "Why would you use sequences when programing Dash or Dot?
   - Sample response: "I would use sequences to help me predict and organize my code."
3. Ask, "How do we use sequences in our daily lives?" (Sample response: "I use sequences when I am brushing my teeth.")

Direct Instruction

INTRODUCTION

1. Say, "As we've discussed, you make **sequences** when you create a list of steps. For example, what is the sequence for picking up your backpack?"
   - Sample response: "The **sequence** for picking up your backpack is: locate your backpack, lean over the backpack, grab a strap, and stand back up.")
2. Ask, "What would happen if you put these steps in the wrong order?"
   - Sample response: "If you put the steps in the wrong order, then you might never be able to pick up your backpack.")

QUICK CHECK

- Why is it important to make sure the **sequence** of our code is correct? (Sample response: “If the **sequence** of our code is wrong, then the program will not run the way we want it to run.”)
- Why do we use **sequences** when coding? (Sample response: “We use **sequences** to help us organize and plan our **algorithms**.”)

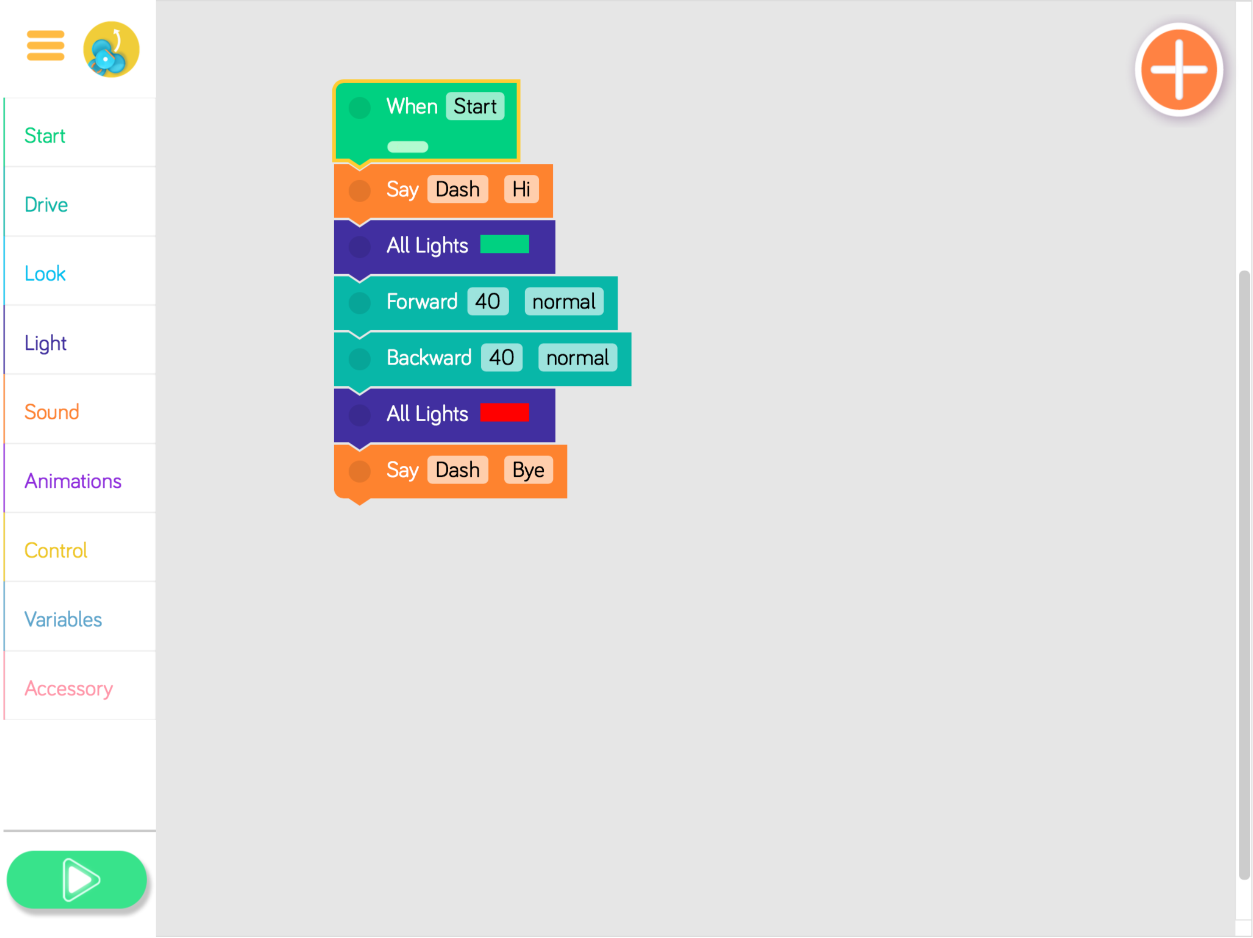
Guided Practice

ACTIVITY: DASH MOVES

1. Project your tablet screen, open the *Blockly* app, and go to the menu at the top left of the screen.

☰

2. Go to the **Create New Project** menu and select the **Dash Moves** preset program. Then tap “create.”



```
When Start
Say Dash Hi
All Lights 🟩
Forward 40 normal
Backward 40 normal
All Lights 🟥
Say Dash Bye
```

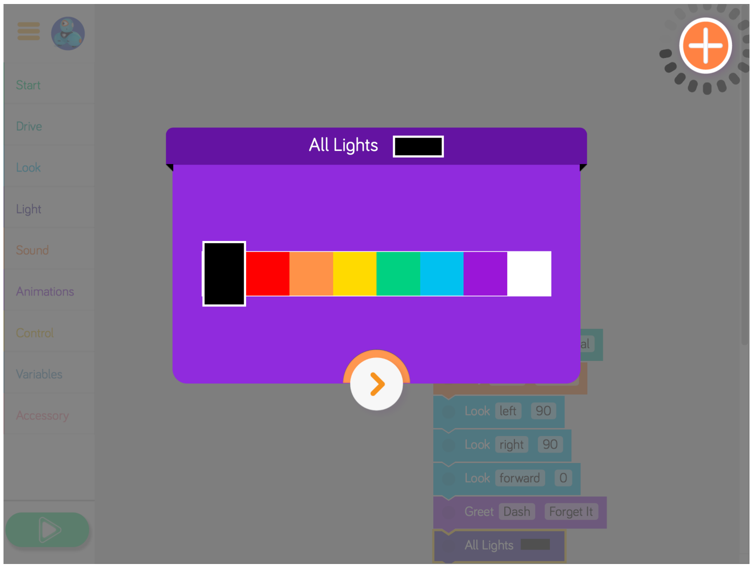
3. Say, “Dash wants to make a fun sequence!”

- Play the program and have students share what they notice. (Sample response: “Dash followed a list of steps.”)
- Select a student to point out the sequence.

4. Say, “Now we we going to make some changes to this program, but we’re not going to add new blocks. Instead, we’re going to change the **parameters** of some of the blocks that we already have. Parameters refer to the specific information inside each block.”

5. Say, “First, we’re going to change the colors of the lights.”

- Tap on the **All Lights** block and choose the color you would like.



All Lights

Look left 90

Look right 90

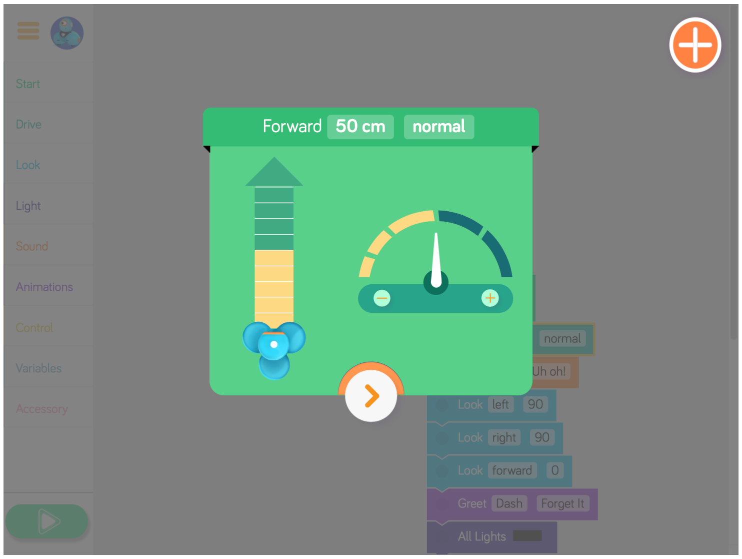
Look forward 0

Greet Dash Forget It

All Lights

6. Say, “Now we’re going to change how far and how fast Dash drives.”

- Tap on the **Drive Forward** block.



Forward 50 cm normal

normal

Uh oh!

Look left 90

Look right 90

Look forward 0

Greet Dash Forget It

All Lights

- Say, “In order to change how far Dash goes, drag the bar in front of Dash up or down.”
- Say, “In order to change how fast Dash goes, tap the plus or minus sign on the dial.”

7. Ask student volunteers what they would like to change.

- Make their suggested changes.
- Run the program to watch the effects of the changes.

QUICK CHECK

- How do you change Dash's speed? (Sample response: "Tap the **Drive Forward** or **Drive Backward** block and tap the plus or minus sign on the dial.")
- How do you change the distance Dash travels? (Same response: "Tap the **Drive Forward** or **Drive Backward** block and drag the bar in front of Dash up or down.")
- How do you change the color of **All Lights** on Dash? (Sample response: "Tap the **All Lights** block and choose the color you want from the bar.")

Independent Practice

Have students work on the following activities in small groups (ideally 2–3 students per robot).

Encourage students to share tablet and robot time. Have them establish and rotate through roles such as:

- **Lead Programmer**: Holds the tablet and manipulates the code.
- **Robot Wrangler**: Retrieves and resets the robot after every program attempt.
- **Documentarian**: Records group results, thoughts, and progress. Illustrates group designs and ideas.

When students work together while coding, they're able to help each other identify mistakes and develop creative solutions!

DASH MOVES EXTENSION

1. Have students use the **Wonder Journal: Dash Planning** and/or **General Planning** worksheets to design ways to add to or alter the **Dash Moves** preset program. They can:
    - Change the **sequence**.
    - Change the **parameters**.
    - Add **lights**, **sounds**, and/or **animations**.

2. After they finish revising the program, have students:
    - Complete a **Wonder Journal: Reflection** worksheet.
    - Take a screenshot of their *Blockly* code.
    - Take a video of Dash while the code is running.

CHALLENGE CARDS [Optional]

1. You can purchase our ***Learn to Code*** **Challenge Card** sets and **Curriculum Guide** here: https://store.makewonder.com/#/education.

2. Have students complete the following **Challenge Cards**:

- **B 1.1: Dash the Collector**
- **B 1.2: It's Candy Time!**
- **B 1.3: Egg Help!**

3. For each challenge, encourage students to:

- Use the **Wonder Journal: Dash Planning** and/or **General Planning** worksheets to discuss how they can complete the challenge.
- Review the **Troubleshooting** and **Problem Solving & Debugging** worksheet if they run into any problems with their code.

4. After they finish each challenge, have students:

- Complete a **Wonder Journal: Reflection** worksheet.
- Take a screenshot of their *Blockly* code.
- Take a video of Dash while the code is running.

Wrap Up

STUDENT PRESENTATIONS

1. Have student groups take turns sharing one of their programs with the class. Encourage them to:

- Explain their design thinking. (E.g., "We changed the parameters in the **Drive Forward** block so that Dash would go further.")
- Share any obstacles and difficulties they overcame during the activity. (E.g., "We weren't sure how to make Dash drive faster, but then we tapped on the plus sign in the dial that came up after tapping the **Drive Forward** block.")

2. Encourage students to ask each other how they accomplished different objectives and give each other feedback on their programs. Possible questions/feedback includes:

- "How did you (change the **parameters** in your program)?"
- "I like how you (changed the **sequence**)."
- "What if you (programmed Dash to create another **loop**)?"

FOLLOW-UP QUESTIONS/DISCUSSION

- How did **sequences** help you with your program?
- What are parameters? (Sample response: "Parameters refer to the specific information inside each block, such as the distance or speed Dash travels.")
- How can you improve your code by changing parameters? (Sample response: "I can make my code more flexible and interesting.")

ASSESSMENT

- Use our **Evaluation Rubric** to review students' work and presentations.
- [Optional] Share your students' work with the world using @wonderworkshop and #dashanddot!

    - Example Tweets/Posts

Standards

CSTA

- Develop programs with sequences and simple loops, to express ideas or address a problem.
- Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
- Develop a plan that describes a program's sequence of events, goals, and expected outcomes.
- Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops.
- Using correct terminology, describe steps taken and choices made during the iterative process of program development.

ISTE

4d: Exhibit a tolerance for ambiguity, perseverance, and the capacity to work with open-ended problems.
5a: Formulate problem definitions suited for technology-assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.
6b: Create original works or responsibly repurpose or remix digital resources into new creations.
7c: Contribute constructively to project teams, assuming various roles and responsibilities to work effectively toward a common goal.

NGSS

3-5-ETS1-2: Generate and compare multiple solutions to a problem based on how well they meet the criteria and constraints of the design problem.

Common Core

CCSS.ELA-LITERACY.W.2.8: Recall information from experiences or gather information from provided sources to answer a question.
CCSS.ELA-LITERACY.SL.3.1: Engage effectively in a range of collaborative discussions (one-on-one, in groups, and teacher-led) with diverse partners on grade 3 topics and texts, building on others' ideas and expressing their own clearly.
CCSS.ELA-LITERACY.SL.3.3: Ask and answer questions about information from a speaker, offering appropriate elaboration and detail.

CCSS.ELA-LITERACY.SL.3.4: Report on a topic or text, tell a story, or recount an experience with appropriate facts and relevant, descriptive details, speaking clearly at an understandable pace.