Eaglebots

Dr. Rhonda Weimann
Instructional Technology Specialist
Innovative Learning and Arts
600 SE 3rd Ave. 5th Floor
Fort Lauderdale, FL 33301
(754) 321-2632
rhonda.weimann@browardschools.com

For information concerning Teacher Grant opportunities, such as interschool visits, staff development, workshops, and Adapter and Disseminator grants, please contact:

The Broward Education Foundation
600 SE Third Avenue, 1st Floor
Fort Lauderdale, FL 33301
754-321-2032
www.BrowardEdFoundation.org

Table of Contents

Goals, Objectives and Project **1**

Standards **2**

- **SCIENCE Next Generation Sunshine State Standards (NGSSS)** **2**

  Third Grade 2
  Fourth Grade 2
  Fifth Grade 2

- **ENGLISH LANGUAGE ARTS Florida Standards (LAFS)** **3**

  Third Grade 3
  Fourth Grade 3
  Fifth Grade 4

- **MATHEMATICS Florida State Standards (MAFS)** **4**

  Third Grade 4
  Fourth Grade 4
  Fifth Grade 5

- **Technology Standards (ISTE)** **6**

Code.org Strategies **7**

Rethinking Classroom Strategies for Teaching Computer Science 7
Code.org Debugging Recipe 8
Code.org Puzzle Solving Recipe 9

Code.org Courses **10**

Lego Robotics Course **11**

Lesson Plans Code.org **12**

- **Unplugged Lesson from Course 1 Move It, Move It** **12**
  - **Move It Activity** **18**
  - **Move It Assessment** **22**
- **Unplugged Lesson from Course 2 Real Life Algorithms: Paper Planes** **25**
  - **Paper Planes Activity** **33**
  - **Algorithms Assessment** **34**
- **Unplugged Lesson from Course 3 Algorithms: Dice Race** **35**
  - **Dice Race Activity** **43**
  - **Dice Race Assessment** **44**

Lego Robotics **45**

- **Video Tutorials for EV3** **45**
- **Robot designs** **46**

Materials List **48**

- **Websites** **49**

Bibliography **50**

Goals, Objectives and Project

Goals

(1) Teach and reinforce science, math and technology skills

(2) Improve students' cooperative learning skills.

(3) Help students develop interpersonal skills.

(4) Expose students to the field of computer science.

Objectives

OVERALL:

- Students will understand that scientists work collaboratively.
- Students will use computational thinking and programming.
- Students will learn to persevere to solve problems.
- Students will improve their critical thinking skills.
- Students will work collaboratively to find solutions.

The Project

The students participate in a 20-hour course of Computer Coding combined with Lego Robotics. The course outline is provided on-line from Code.org. The K-8 Intro to Computer Science is a free course that aims to demystify computer science and show K-8 students that it's fun, collaborative, and creative. The course is designed to motivate students and educators to continue learning computer science to improve real world relationships, connections, and life. Currently, Code.org has developed three new courses for elementary age children specifically developed to meet the needs of students from early reading ages to higher level readers. In addition to teaching this course I incorporated Lego Robotics and programming the robot to complete missions.

For the Lego Robotics part of the course the students watched tutorial videos on how to use the Lego Mindstorms software to program the robots. After viewing the tutorials and having a discussion with the class, the students would use the software to program the robot. Students were grouped in cooperative teams to participate in the programming. Although the students worked in collaborative groups, each student was given the opportunity to write an individual program to run on the robots. Each program needed to include key elements the robot was to perform such as display a picture on the screen, make a sound or talk and move forward, backward, in a circle or other direction.

Standards

**All activities include elements that address the standards listed below in the areas of: Nature of Science (Science NGSSS), (Mathematics MAFS), Comprehension and Collaboration (English Language Arts LAFS-Speaking and Listening) and Technology (International Society for Technology Education-ISTE).*

SCIENCE Next Generation Sunshine State Standards (NGSSS)

Third Grade

The Nature of Science

SC.3.N.1.4
Recognize the importance of communication among scientists.

SC.3.N.1.5
Recognize that scientists question, discuss, and check each other’s evidence and explanations.

SC.3.N.1.6
Infer based on observation.

SC.3.N.3.2
Recognize that scientists use models to help understand and explain how things work.

SC.3.N.3.3
Recognize that all models are approximations of natural phenomena; as such, they do not perfectly account for all observations.

Fourth Grade

SC.4.N.1.7

Recognize and explain that scientists base their explanations on evidence.

SC.4.N.1.8
Recognize that science involves creativity in designing experiments.

SC.4.N.3.1
Explain that models can be three dimensional, two dimensional, an explanation in your mind, or a computer model.

Fifth Grade

SC.5.N.1.6
Recognize and explain the difference between personal opinion/interpretation and verified observation.

SC.5.N.2.1
Recognize and explain that science is grounded in empirical observations that are testable; explanation must always be linked with evidence.

SC.5.N.2.2
Recognize and explain that when scientific investigations are carried out, the evidence produced by those investigations should be replicable by others.

LANGUAGE ARTS Florida Standards (LAFS)

Comprehension and Collaboration

Third Grade

LAFS.3.SL.1.1

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.

LAFS.3.SL.2.6

Speak in complete sentences when appropriate to task and situation in order to provide requested detail or clarification.

Fourth Grade

LAFS.4.SL.1.1

Engage effectively in a range of collaborative discussions (one-on-one, in groups, and teacher-led) with diverse partners on grade *4 topics and texts*, building on others' ideas and expressing their own clearly.

LAFS.4.SL.2.4

Report on a topic or text, tell a story, or recount an experience in an organized manner, using appropriate facts and relevant, descriptive details to support main ideas or themes; speak clearly at an understandable pace.

Fifth Grade

LAFS.5.SL.1.1

Engage effectively in a range of collaborative discussions (one-on-one, in groups, and teacher-led) with diverse partners on *grade 5 topics and texts*, building on others' ideas and expressing their own clearly.

LAFS.5.SL.2.4

Report on a topic or text or present an opinion, sequencing ideas logically and using appropriate facts and relevant, descriptive details to support main ideas or themes; speak clearly at an understandable pace.

MATHEMATICS Florida State Standards (MAFS)

Number and Operations in Base Ten

Third Grade

MAFS.3.NBT.1.2

Fluently add and subtract within 1000 using strategies and algorithms based on place value, properties of operations, and/or the relationship between addition and subtraction.

Fourth Grade

MAFS.4.NBT.2.4

Fluently add and subtract multi-digit whole numbers using the standard algorithm.

Fifth Grade

MAFS.5.NBT.2.7
Add, subtract, multiply, and divide decimals to hundredths, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method and explain the reasoning used.

Measurement and Data

Third Grade

MAFS.3.MD.3.5

Recognize area as an attribute of plane figures and understand concepts of area measurement.

MAFS.3.MD.3.6

Measure areas by counting unit squares (square cm, square m, square in, square ft, and improvised units).

Fourth Grade

MAFS.4.MD.3.5

Recognize angles as geometric shapes that are formed wherever two rays share a common endpoint, and understand concepts of angle measurement:

a. An angle is measured with reference to a circle with its center at the common endpoint of the rays, by considering the fraction of the circular arc between the points where the two rays intersect the circle. An angle that turns through 1/360 of a circle is called a "one-degree angle," and can be used to measure angles.
b. An angle that turns through *n* one-degree angles is said to have an angle measure of *n*degrees.

MAFS.4.MD.3.7

Recognize angle measure as additive. When an angle is decomposed into non-overlapping parts, the angle measure of the whole is the sum of the angle measures of the parts. Solve addition and subtraction problems to find unknown angles on a diagram in real world and mathematical problems, e.g., by using an equation with a symbol for the unknown angle measure.

Geometry

Fifth Grade

MAFS.5.G.2.3

Understand that attributes belonging to a category of two-dimensional figures also belong to all subcategories of that category. *For example, all rectangles have four right angles and squares are rectangles, so all squares have four right angles.*

Technology ISTE Standards for Students

1. Creativity and innovation

Students demonstrate creative thinking, construct knowledge, and develop innovative products and processes using technology.

2. Communication and collaboration

Students use digital media and environments to communicate and work collaboratively, including at a distance, to support individual learning and contribute to the learning of others.

3. Research and information fluency

Students apply digital tools to gather, evaluate, and use information.

4. Critical thinking, problem solving, and decision making

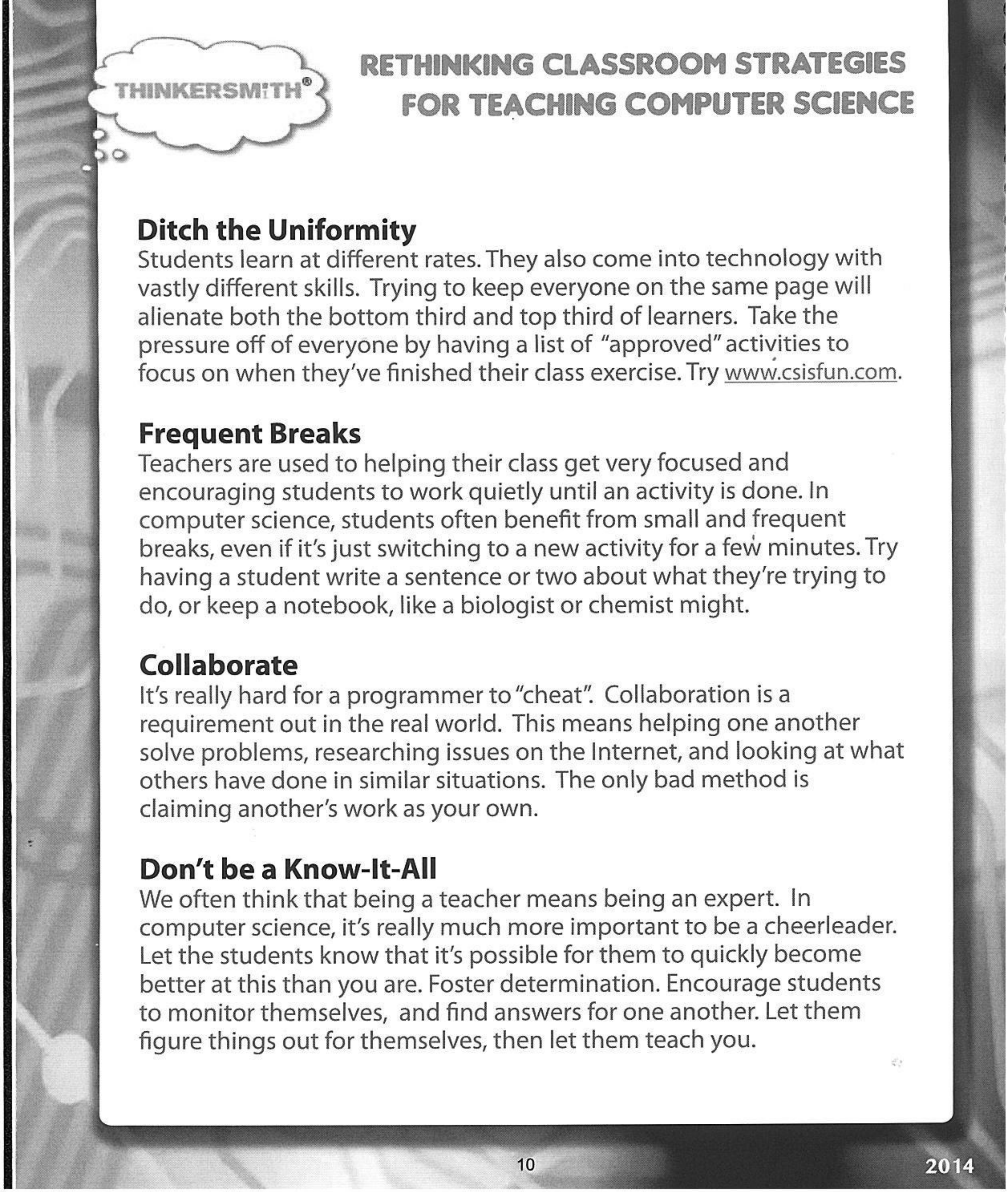
Students use critical thinking skills to plan and conduct research, manage projects, solve problems, and make informed decisions using appropriate digital tools and resources**.**

5. Digital citizenship

Students understand human, cultural, and societal issues related to technology and practice legal and ethical behavior.

6. Technology operations and concepts

Students demonstrate a sound understanding of technology concepts, systems, and operations.



Ditch the Uniformity

Students learn at different rates. They also come into technology with vastly different skills. Trying to keep everyone on the same page will alienate both the bottom third and top third of learners. Take the pressure off of everyone by having a list of "approved" activities to focus on when they've finished their class exercise. Try www.csisfun.com.

Frequent Breaks

Teachers are used to helping their class get very focused and encouraging students to work quietly until an activity is done. In computer science, students often benefit from small and frequent breaks, even if it's just switching to a new activity for a few minutes. Try having a student write a sentence or two about what they're trying to do, or keep a notebook, like a biologist or chemist might.

Collaborate

It's really hard for a programmer to "cheat". Collaboration is a requirement out in the real world. This means helping one another solve problems, researching issues on the Internet, and looking at what others have done in similar situations. The only bad method is claiming another's work as your own.

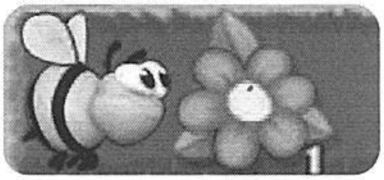
Don't be a Know-It-All

We often think that being a teacher means being an expert. In computer science, it's really much more important to be a cheerleader. Let the students know that it's possible for them to quickly become better at this than you are. Foster determination. Encourage students to monitor themselves, and find answers for one another. Let them figure things out for themselves, then let them teach you.

These debugging tips will help you keep moving when you get stuck!

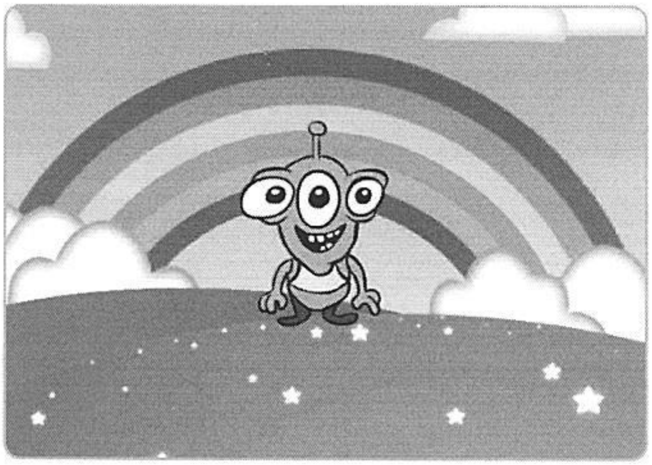
Work to Avoid Mistakes

- Read the directions.
- What is the goal of the puzzle?
- Take it slow and go one step at a time.
- Can you talk about the problem in your own words?
- Were you given any code to start?
  - What does it do?
  - Why do you think it's there?



Debugging

- Look for problems each step of the way.
- Describe what was supposed to happen.
- Describe what is going wrong.
- Does the difference between what was supposed to happen and what did happen give you any clues?



- Fix one thing at a time, then describe how the result changed.
- Try leaving "breadcrumbs" in your program. You can put clues inside your code (like having your program "say" something) to let you know when each chunk runs.
- Try doing each task as its own chunk, then put all of the pieces together at the end so it is easier to see what each thing does.
- Try at least three ways of fixing problems before you ask for help.
- Talk to a friend. Maybe one of your classmates can help you figure out where your plan goes awry.

(Based on Polya's Four Step Problem Solving Process)

D E

These tips will help you get unstuck when solving Code.org puzzles!

Step 1: Understand the Puzzle

- What does the puzzle want you to do?
- Can you talk about the problem in your own words?
- Were you given any code to start?
  - What does it do?
  - Why do you think it's there?
- What is the goal of the puzzle?
- Have you solved any other puzzles that are like this one?

Step 2: Create a Plan *(Pick one or more)*

- Write an algorithm.
- Guess and check as you go.
- Draw a picture of what you want to do.
- Try working backward.
- Solve one small piece at a time.
- Compare to a puzzle that you've already solved.

Step 3: Perform and Perfect the Plan

- Did you solve the puzzle?
- If not, hunt for one error at a time.
- Retest your plan after every change.
- If you start to get frustrated, take a deep breath, or leave your screen for a minute. When you come back, you may see what was causing the trouble!
- Ask questions. Maybe one of your friends can help you figure out where your plan goes awry.

Step 4: Check Your Work

- Does your answer solve the puzzle?
- Did you hit all of the goals of this puzzle?
- Now that you one way to solve the puzzle, is there an easier way to do it?
- If you change this solution a little, will it work for any other puzzles?
- Could you explain your solution to help someone else figure out the puzzle?

Code.org Courses

Course Structure

The courses are structured using the spiraling curriculum design, in which concepts and skills are revisited in each course while delving deeper each time.

Each course experience is a blend of online activities and "unplugged" activities, lessons in which students can learn computing concepts with or without a computer. The online experiences are composed of self-guided and self-paced tutorials, which use scaffolded sets of programming instructions to explore and practice algorithmic thinking. The unplugged lessons take a hands-on, often kinesthetic approach, making use of physical manipulatives to model computational concepts.

Each course consists of about 18 to 20 lessons, each lasting between 25 and 45 minutes. They can be taught at a comfortable pace whether in consecutive days as a sub-unit or one day a week for 18 weeks. The content of each course builds conceptually on the previous course, so that a student can progress through all three experiences learning new concepts along the way.

Course 1 (Designed for Early-Readers in Lower Elementary Grades- K/1st grade Currently Available starting in Sept. 2014)

Students create computer programs with loops and events and write algorithms for everyday tasks. During this course students learn to collaborate with others meaningfully, investigate different problem-solving techniques, persist in the face of difficult tasks and learn about Internet safety. By the end of this course, students create their very own custom game or story that they can share.

Course 2 (Designed for Readers in Middle Elementary Grades- 2nd and 3rd grade Currently Available starting in Sept. 2014)

Students create programs with loops, events, and conditionals and write algorithms for everyday tasks. They will translate their names into binary investigate different problem-solving techniques, and discuss societal impacts of computing. By the end of the curriculum students create interactive games or stories they can share. The complexity and depth of topics discussed are scaffolded appropriately to provide all students a rich and novel experience.

Course 3-Prerequisite is Course 2 (Designed for Early-Readers in Upper Elementary Grades- 4th and 5th grade Currently Available starting in Sept. 2014)

Students create programs with different kinds of loops, events, functions, and conditions and write algorithms for everyday tasks. Through this they will investigate different problem-solving techniques, discuss societal impacts of computing and the Internet, and learn about Internet transmission methods. By the end of the curriculum, students create interactive stories and games they can share with anyone. Students taking Course 3 will have already taken Course 2.

K-8 Computer Science Course (Original Course Offered)

The K-8 Intro to Computer Science is a free 20-hour course that aims to demystify computer science and show K-8 students that it's fun, collaborative, and creative. The course consists of unplugged hands-on activities and interactive on-line learning experiences. The course is designed to motivate students and educators to continue learning computer science to improve real world relationships, connections, and life.

Lego Robotics Course

Students learn to use the Lego Mindstorms software to program robots to complete missions. Students work together to build the Lego Mindstorms robot. After building the basic model robot, students choose different robot designs to build and program. Students learn to program the robot to speak, display a visual symbol, move forward, backward, turn left and right and various other actions. Students learn to collaborate with others to solve problems and persist until a solution is reached. Students learn to think critically to design actions in order to move the robot in the direction needed. Students are encouraged to try different resolutions to achieve desired outcomes.

U

UNPLUGGED

Move It, Move It

Lesson time: 20 Minutes

to delve deeper whe Basic lesson time includes activity only. Introductory
time allows. and Wrap-Up suggestions can be used

LESSON OVERVIEW

This lesson will help students realize that in order to give clear instructions, they need a common language.

Students will practice controlling one another using a simple combination of hand gestures. Once they understand the language, they will begin to "program" one another by giving multiple instructions in advance.

TEACHING SUMMARY

Getting Started - 15 minutes

1) Review
2) Vocabulary
3) Let's Control Ourselves

Activity: Move It, Move It - 20 minutes

4) Move It, Move It: Multi-Step Adventure

Wrap-up - 5 minutes

5) Flash Chat - What did we learn?
6) Vocab Shmocab

Assessment - 10 minutes

7) Move the Flurbs 2

LESSON OBJECTIVES

Students will:

Recognize situations where they can create programs to complete tasks
Predict moves necessary to get teammate from start to finish
Convert movements into symbolic instructions
Relate algorithms as programs to teammates

TEACHING GUIDE

MATERIALS, RESOURCES AND PREP

For the Student

Maps and Key from Move It, Move It: Multi-Step Adventure
Assessment Worksheet: Move the Flurbs 2
Scissors
Glue

For the Teacher

Teacher Lesson Guide
Print one Move It, Move It: Multi-Step Adventure activity pack on Cardstock for each group
Print Assessment Worksheet: Move the Flurbs 2 for each student

GETTING STARTED (15 MIN)

1) Review

This is a great time to review the last lesson that you went through with your class. You can do this as one large group or have students discuss with an elbow partner.

Here are some questions that you can ask in review:
What did we do last time?
What do you wish we had had a chance to do?
Did you think of any questions after the lesson that you want to ask?
What was your favorite part of the last lesson?
Lesson Tip: Finishing the review by asking about the students' favorite things helps to leave a positive impression of the previous exercise, increasing excitement for the activity that you are about to introduce.

2) Vocabulary

This lesson has one new and important word:
Program - Say it with me: Pro-gram
An algorithm that has been coded into something that can be run by a machine.

New Word!

Program

Say it with me: Pro-gram

An algorithm that has been coded
into something that can be run by a machine.

3) Let's Control Ourselves

Ask your students if they remember helping you draw a smiley face last time.
Review the instructions by drawing as they guide you one step at a time.

Let them know that you're going to make a small change.

Now, I want you to give me two instructions at a time before I move my pen.
Can you do it again, but give me three instructions at a time?
When you give me multiple instructions at a time, you're providing me with an "algorithm" to draw each piece of the smiley face.
Now, suppose we were to have a secret "code" for each of those instructions. For example, "Draw an Eye" could look like this: (make a large circle with your hands). If we had special codes for each of those steps, then our algorithm would become a program.
We're going to play a game that allows us to program each other...and you'll do it all with your arms!
Lesson Tip: Feel free to do an example map with the students as a class before breaking them into groups or even describing the rules (beyond how to react to each of the arm gestures). Learning through play is often more effective than spouting off all of the rules at this age.

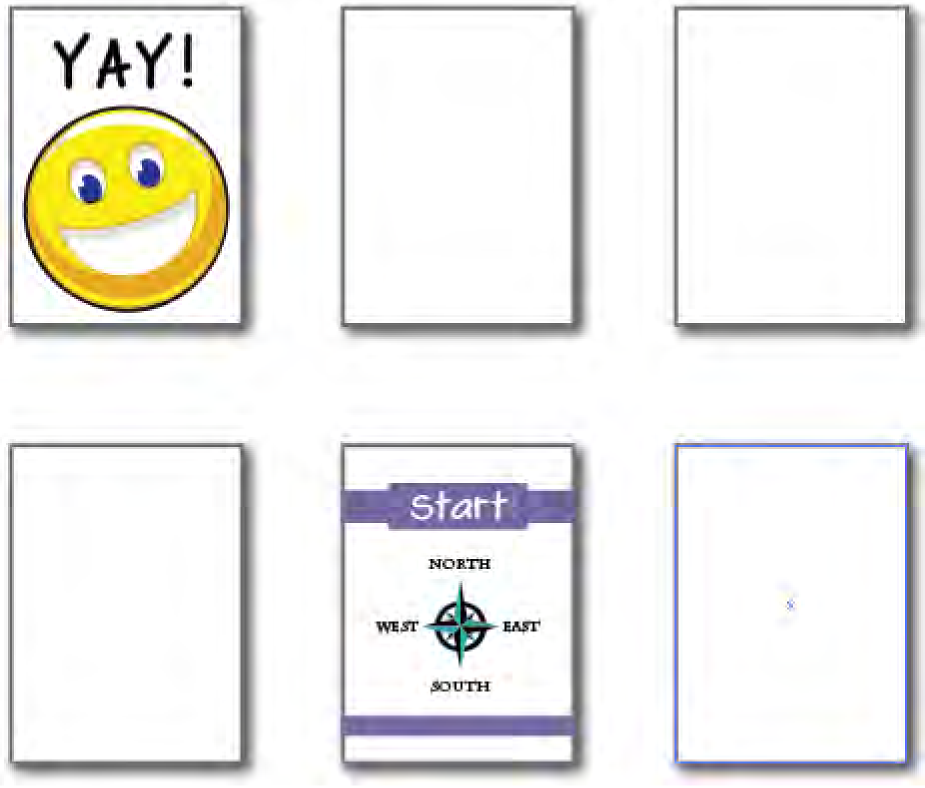
ACTIVITY: (20 MIN)

4) Move It, Move It: Multi-Step Adventure

This worksheet helps teach students how to think ahead in multiple steps, as they plan a short route from their friend's start location to the hidden smiley face, up to three steps away.
Print out an activity packet for every group (ideally 2 to 4 students) and cut the Map Cards apart.
Explain the rules to the class, making sure to emphasize the new word "program."
Directions for Class:
1. Decide who will be the Walking Machine and who will be the Controller.

2. Have the Controller set up a grid on the floor made up of pieces of paper as shown on one of the Move It Maps, except with the smiley face upside down, facing the ground.
3. The Walking Machine will start by standing on the page with the compass rose.
4. The Controller will then lead the Walking Machine step-by-step through the paper maze that they created, using the provided arm signals.
5. When the Controller gives the signal to "STOP," the Walking Machine will flip over the page that they are on. If that page is a smiley face, then the maze was a success!

The Controller (and anyone else in the group who is not the Walking Machine) can set up a map made of paper, based on one of the Move It Map cards.



Remember that the smiley face map page should actually be set facing the ground, so that the Walking Machine cannot easily tell where their final location is.

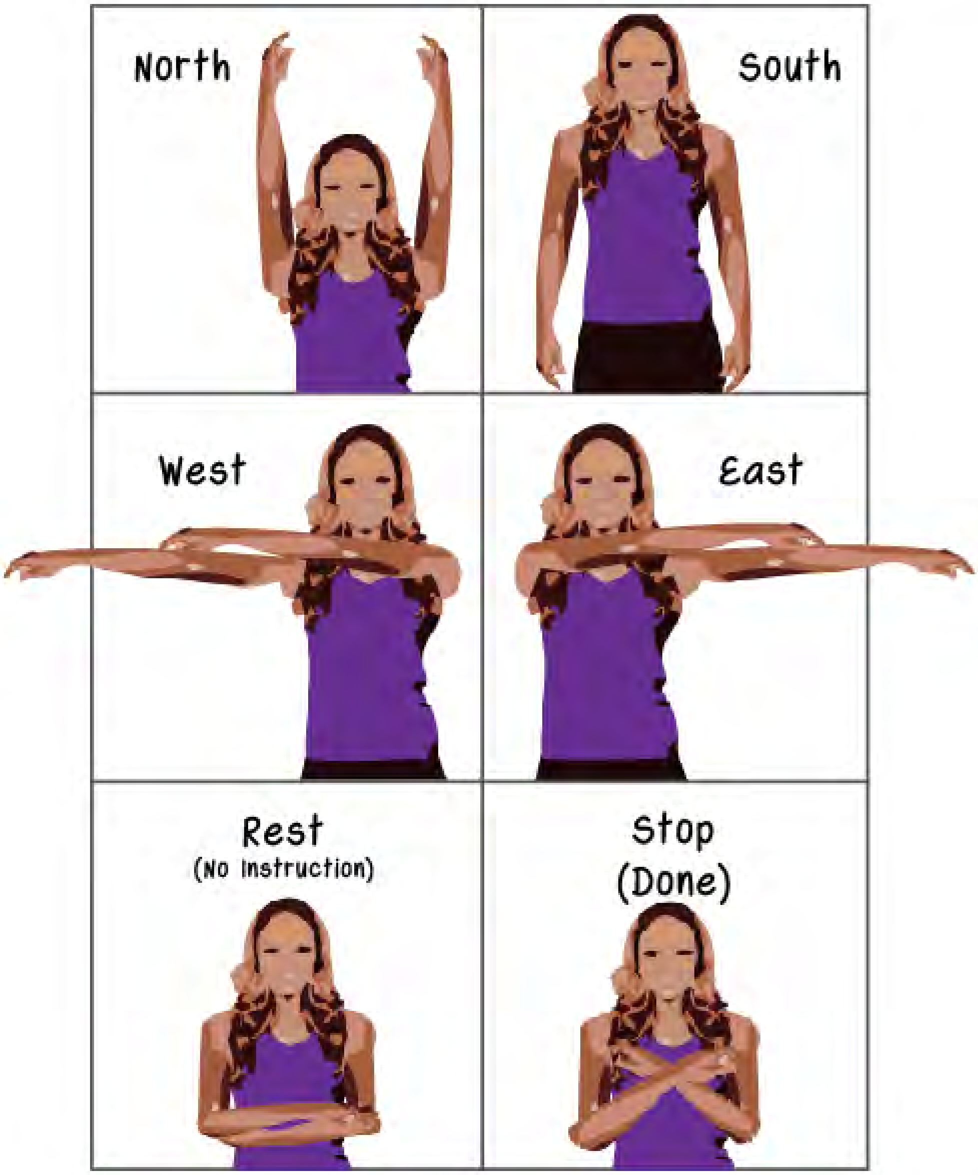
The Walking Machine begins by standing on the piece of paper imprinted with the compass rose.

Start
NORTH
WEST
EAST
SOUTH

Lesson Top: Here are some useful links in case your class hasn't yet talked about the compass rose and cardinal directions:
The Cardinal Directions Geography Song
Cardinal Direction Mnemonics Lesson
These topics will be important in the online lessons that follow, so taking an extra couple of minutes to be sure that your students correlate North with Up, South with Down, East with Right, and West with Left will continue to be helpful hereafter.

The Controller uses arm movements to guide the Walking Machine. Encourage the Controller to be facing the same direction as the Walking Machine to avoid having them get confused by "East is Right" and "West is Left."
Controllers should start by giving one direction at a time, allowing the Walking Machine to take a step before they move on to the next direction.

Halfway into the activity, you can encourage your students to Control with two instructions before they allow the Walking Machine to take a step, and then three.
Ideally, by the time the lesson is complete, the students will relay the entire "program" to the Walking Machine before the Walking Machine even takes their first step.
[together] [tip]



LESSON TIP

Note that the rules are not the most important thing here. Feel free to clarify if the students have questions, but if the students are playing a bit differently than described, you don't need to hold them to the letter of the game. The crucial bit is that they are moving from immediate instructions to giving two or three instructions before the WalkingMachine moves.

[/tip] [/together]

WRAP-UP (5 MIN)

5) Flash Chat: What did we learn?

In the game we just played, who do you suppose was more like a programmer, and who was more like a computer?

What were the four directions on the compass rose?

What tricks can we use to remember North, South, East and West?

How could we have given instructions without using our arms?

What was your favorite part about that game?

6) Vocab Shmocab

Which one of these definitions did we learn a word for today?
"The fluffy feathers of a baby bird"
"The circuit board that controls a robot"
"An algorithm that has been coded into something that can be run by a machine"
...and what is the word that we learned?

ASSESSMENT (10 MIN)

7) Move the Flurbs 2

Hand out the worksheet titled "Move the Flurbs 2" and allow students to complete the activity independently
after the instructions have been well explained.
This should feel familiar, thanks to the previous two activities.

EXTENDED LEARNING

Use these activities to enhance student learning. They can be used as outside of class activities or other
enrichment.

X's and O's

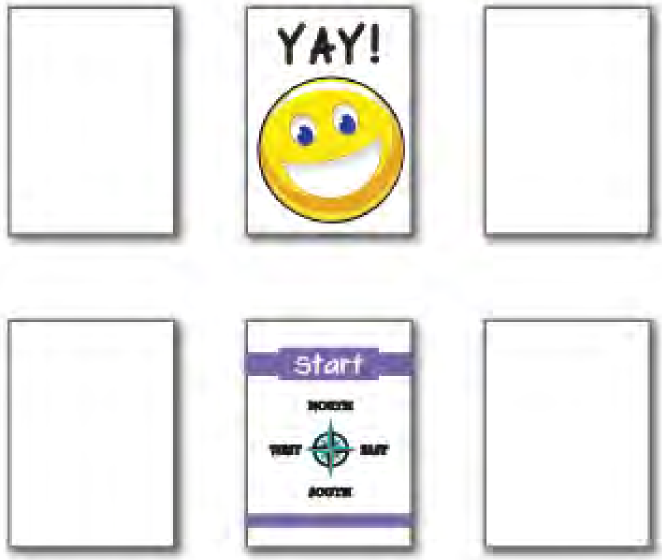
Draw a tic-tac-toe board for the class.
Place a single X and a single O somewhere on the board.
Ask the class if they can get the X to the O using arm gestures as a class.

X's, O's, and Arrows

Similar to the activity above, but have the students write their programs in advance using arrows instead of hand gestures.
This can be done in groups.
Groups can share their solutions for the class.

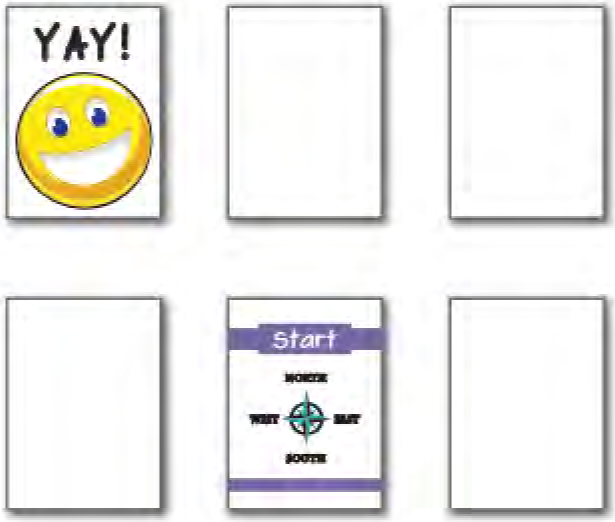
1

Move It Map 1

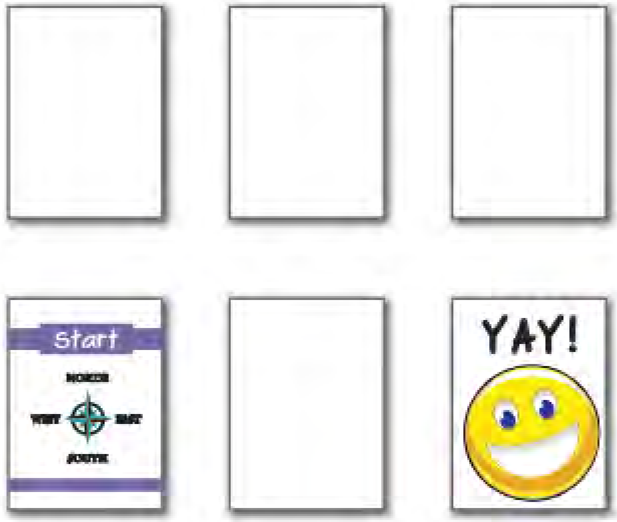


2

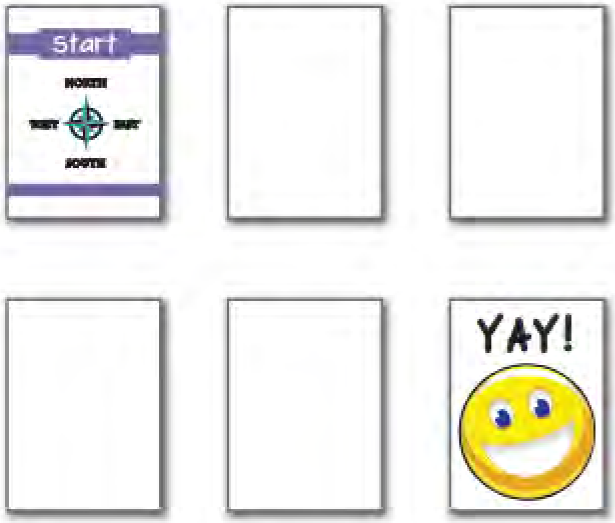
Move It Map 2



3 Move It Map 3



4 Move It Map 4



Name: ____________________ Date: ______________

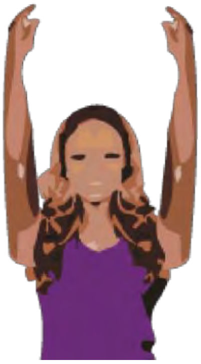
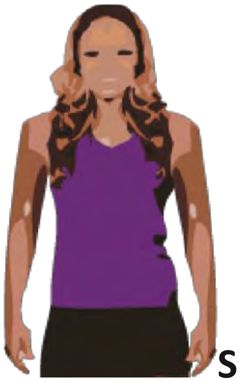
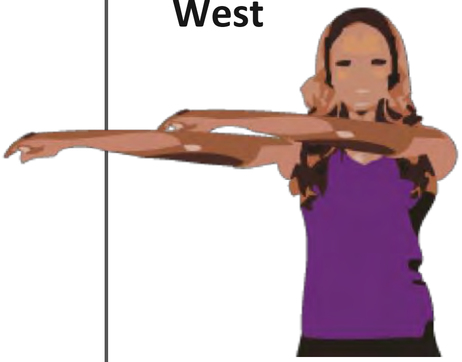
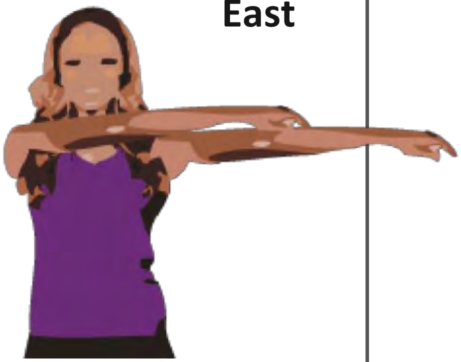
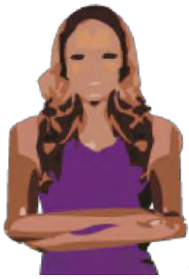
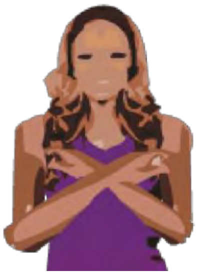
Unplugged

Move It, Move It

Multi-Step Adventure Activity Key

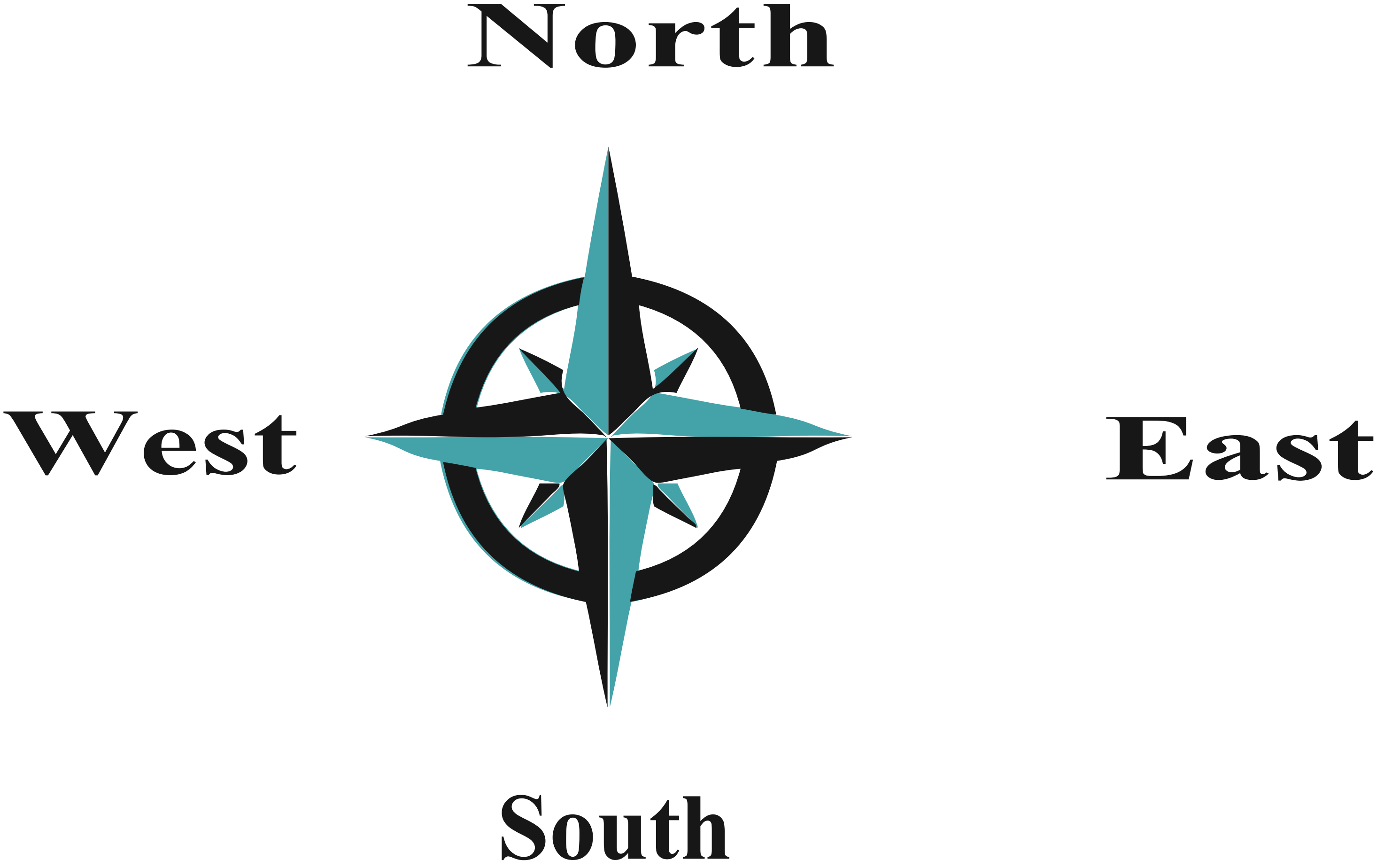
These are the moves that you can do to help guide your friend.

Practice a few times to be sure that you both understand what each move does.

 North	 South
West 	East 
Rest  (No Instruction)	Stop  (Done)

YAY!

Start



Name: ______________________________ Date: ____________________

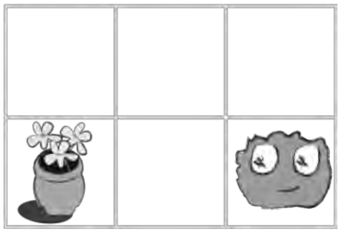
U

Unplugged

Move the Flurbs 2

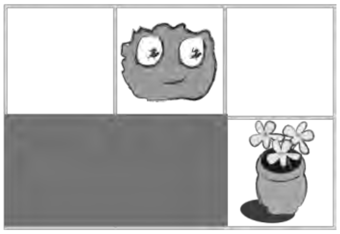
Assessment Worksheet

The weather is getting hot. Help the Flurb get to her flowers so she can water them. To show the Flurb how to get to her flowers, cut out the correct arrows from the bottom of the page and paste them in the program slots by each of the picture maps.



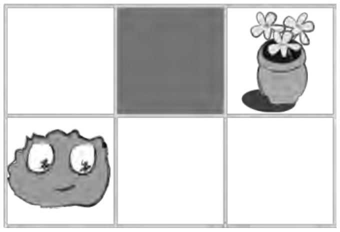
STEP 1	STEP 2

GLUE



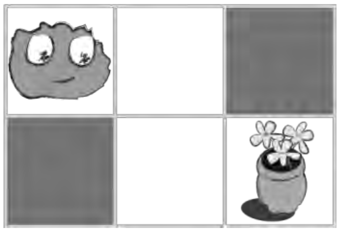
STEP 1	STEP 2

GLUE



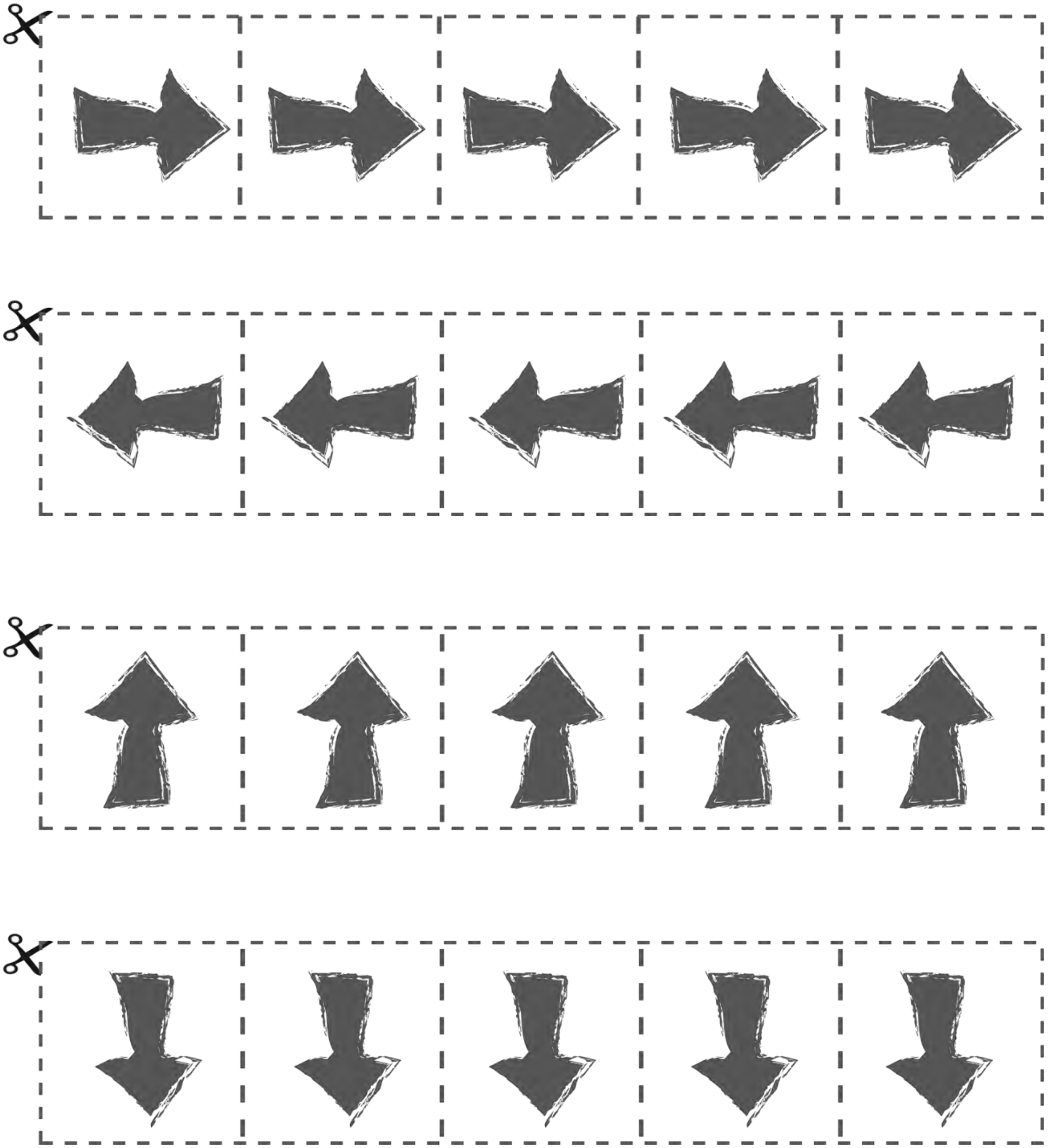
STEP 1	STEP 2	STEP 3

GLUE



STEP 1	STEP 2	STEP 3

GLUE



U

UNPLUGGED

Real-Life Algorithms: Paper Airplanes

Lesson time: 20 Minutes Basic lesson time includes activity only. Introductory and Wrap-Up suggestions can be used to delve deeper when time allows.

LESSON OVERVIEW

In this lesson, students will relate the concept of algorithms back to everyday real-life activities by making paper airplanes. The goal here is to start building the skills to translate real-world situations to online scenarios and vice versa.

TEACHING SUMMARY

Getting Started - 15 minutes

1) Review
2) Vocabulary
3) What We Do Daily

Activity: Real-Life Algorithms - 20 minutes

4) Real-Life Algorithms: Paper Airplanes

Wrap-up - 5 minutes

5) Flash Chat - What did we learn?

Assessment - 10 minutes

6) Daily Algorithms

LESSON OBJECTIVES

Students will:

- Name various activities that make up their day
- Decompose large activities into a series of smaller events
- Arrange sequential events into their logical order

TEACHING GUIDE

MATERIALS, RESOURCES AND PREP

For the Student

- Paper for folding into airplane
- Real-Life Algorithms Worksheet: Paper Airplanes
- Assessment Worksheet: Daily Algorithms
- Scissors
- Glue

For the Teacher

- Teacher Lesson Guide
- Real-Life Algorithms Worksheet: Paper Airplanes
- Print Assessment Worksheet: Daily Algorithms for each student

GETTING STARTED (15 MIN)

1) Review

This is a great time to review the last lesson that you went through with your class. We suggest you alternate between asking questions of the whole class and having students talk about their answers in small groups.

Here are some questions that you can ask in review:

- What did we do last time?
- What do you wish we had had a chance to do?

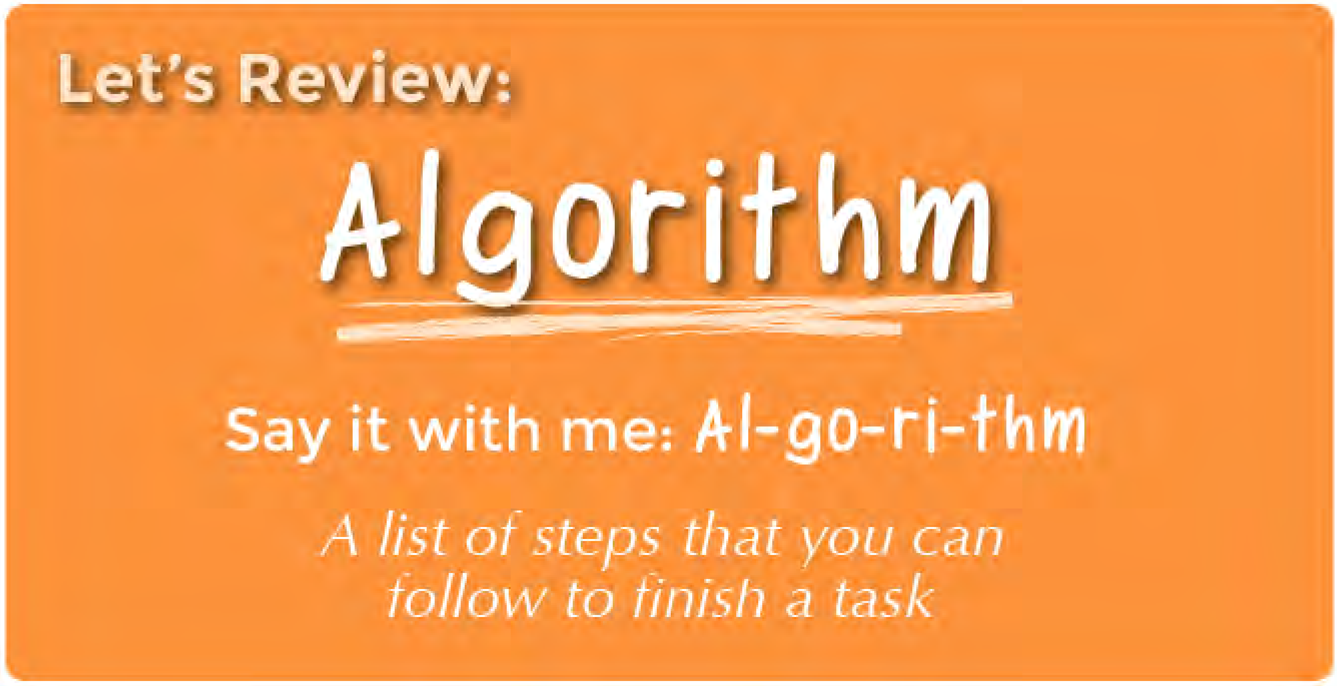
- Did you think of any questions after the lesson that you want to ask?
- What was your favorite part of the last lesson?

LESSON TIP

Finishing the review by asking about the students' favorite things helps to leave a positive impression of the previous exercise, increasing excitement for the activity that you are about to introduce.

2) Vocabulary

This lesson has one vocabulary word that is important to review:



Let's Review:

Algorithm

Say it with me: Al-go-ri-thm

A list of steps that you can follow to finish a task

Algorithm - Say it with me: Al-go-ri-thm
A list of steps that you can follow to finish a task

3) What We Do Daily

- Ask your students what they did to get ready for school this morning.
  - Write their answers on the board.
  - If possible, put numbers next to their responses to indicate the order that they happen.
    - If students give responses out of order, have them help you put them in some kind of logical order.

        - Point out places where order matters and places where it doesn't.

- Introduce students to the idea that it is possible to create algorithms for the things that we do everyday.
    - Give them a couple of examples, such as making breakfast, brushing teeth, and planting a flower.

- Let's try doing this with a new and fun activity, like making paper airplanes!

ACTIVITY: (20 MIN)

4) Real-Life Algorithm Worksheet: Paper Airplanes

LESSON TIP

You know your classroom best. As the teacher, decide if students should do this individually or if students should work in pairs or small groups.

- You can use algorithms to help describe things that people do every day. In this activity, we will create an algorithm to help each other fold a paper airplane.

Directions:

1. Cut out the steps for making a paper airplane provided worksheet.
2. Work together to choose the six correct steps from the nine total options.
3. Glue the six correct steps, in order, onto a separate piece of paper.
4. Trade the finished algorithm with another person or group and let them use it to make their plane!
5. If you are concerned about injury when your students begin flying their paper airplanes, we recommend having them blunt the tip of the plane by either folding it inward or ripping it off and covering the ripped edges with tape.

LESSON TIP

If deciding on the correct steps seems too difficult for your students, do that piece together as a class before you break up into teams.

WRAP-UP (5 MIN)

5) Flash Chat: What did we learn?

- How many of you were able to follow your classmates' algorithms to make your airplanes?
- Did the exercise leave anything out?
    - What would you have added to make the algorithm even better?
    - What if the algorithm had been only one step: "Fold a Paper Airplane"?
        - Would it have been easier or harder?
        - What if it were forty steps?
- What was your favorite part about that activity?

ASSESSMENT (15 MIN)

6) Assessment Worksheet: Daily Algorithms

- Hand out the worksheet titled "Daily Algorithms" and allow students to complete the activity independently after the instructions have been well explained.
- This should feel familiar, thanks to the previous activities.

EXTENDED LEARNING

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

Go Figure

- Break the class up into teams.
- Have each team come up with several steps that they can think of to complete a task.
- Gather teams back together into one big group and have one team share their steps, without letting anyone know what the activity was that they had chosen.
- Allow the rest of the class to try to guess what activity the algorithm is for.

CONNECTIONS AND BACKGROUND INFORMATION

ISTE Standards (formerly NETS)

- 1.a - Apply existing knowledge to generate new ideas, products, or processes
- 1.c - Use models and simulation to explore complex systems and issues.
- 2.b - Communicate information and ideas effectively to multiple audiences using a variety of media and formats.
- 2.d - Contribute to project teams to produce original works or solve problems.
- 4.b - Plan and manage activities to develop a solution or complete a project.
- 6.a - Understand and use technology systems.

CSTA K-12 Computer Science Standards

- CT.L1:3-03. Understand how to arrange information into useful order without using a computer.
- CT.L1:6-01. Understand and use the basic steps in algorithmic problem-solving.
- CT.L1:6-02. Develop a simple understanding of an algorithm using computer-free exercise.
- CT.L1:6-05. Make a list of sub-problems to consider while addressing a larger problem.
- CPP.L1:3-04. Construct a set of statements to be acted out to accomplish a simple task.
- CPP.L1:6-05. Construct a program as a set of step-by-step instructions to be acted out (e.g., make a peanut butter and jelly sandwich activity).
- CT.L2-03. Define an algorithm as a sequence of instructions that can be processed by a computer.
- CT.L2-06. Describe and analyze a sequence of instructions being followed.

NGSS Science and Engineering Practices

- K-2-PS3-2. Use tools and materials provided to design and build a device that solves a specific problem or a solution to a specific problem.
  - 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.

Common Core Mathematical Practices

- 1. Make sense of problems and persevere in solving them.
- 2. Reason abstractly and quantitatively.
- 3. Construct viable arguments and critique the reasoning of others
- 6. Attend to precision.
- 7. Look for and make use of structure.
- 8. Look for and express regularity in repeated reasoning.

Common Core Math Standards

- 1.G.1 - Distinguish between defining attributes versus non-defining attributes; build and draw shapes to possess defining attributes.
- 2.G.3 - Partition circles and rectangles into two, three, or four equal shares, describe the shares using the words halves, thirds, half of, a third of, etc., and describe the whole as two halves, three thirds, four fourths.

Common Core Language Arts

- SL.1.1 - Participate in collaborative conversations with diverse partners about grade 1 topics and texts with peers and adults in small and larger groups.
- SL.1.2 - Ask and answer questions about key details in a text read aloud or information presented orally or through other media.
- L.1.6 - Use words and phrases acquired through conversations, reading and being read to, and responding to texts, including using frequently occurring conjunctions to signal simple relationships.
- SL.2.1 - Participate in collaborative conversations with diverse partners about grade 2 topics and texts with peers and adults in small and larger groups.
- SL.2.2 - Recount or describe key ideas or details from a text read aloud or information presented orally or through other media.
- L.2.6 - Use words and phrases acquired through conversations, reading and being read to, and responding to texts, including using adjectives and adverbs to describe.
- 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.
- SL.3.3 - Ask and answer questions about information from a speaker, offering appropriate elaboration and detail.

- L.3.6 - Acquire and use accurately grade-appropriate conversational, general academic, and domain-specific words and phrases, including those that signal spatial and temporal relationships.

Created in partnership with
THINKERSMITH®

Name: ______________________ Date: ______________

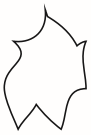
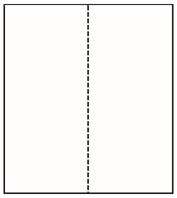
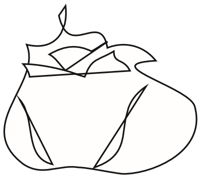
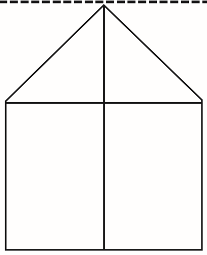
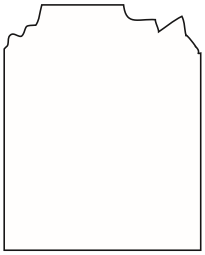
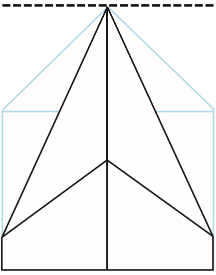
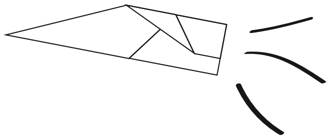
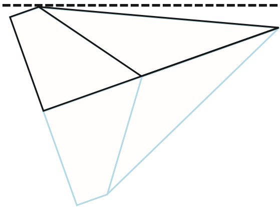
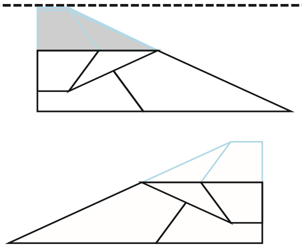
Unplugged

Real-Life Algorithms

Paper Airplane Worksheet

You can use algorithms to help describe things that people do every day. In this activity, we will create an algorithm to help each other make paper airplanes.

Cut out the steps of making an airplane below. Glue the six the correct steps, in order, onto a separate piece of paper. Trade your finished algorithm with another person or group and let them use it to make an actual flying model paper plane!

 CUT CENTER OUT OF PAPER	 CREASE PAPER DOWN THE CENTER	 CRUMBLE PAPER
 FOLD TOP CORNERS TO CENTER	 RIP CORNER OFF PAPER	 FOLD CORNER SIDES TO CENTER
 TOSS FINISHED PLANE	 FOLD PAPER IN HALF AGAIN	 PULL SIDES DOWN

Name: ________________________ Date: ________________

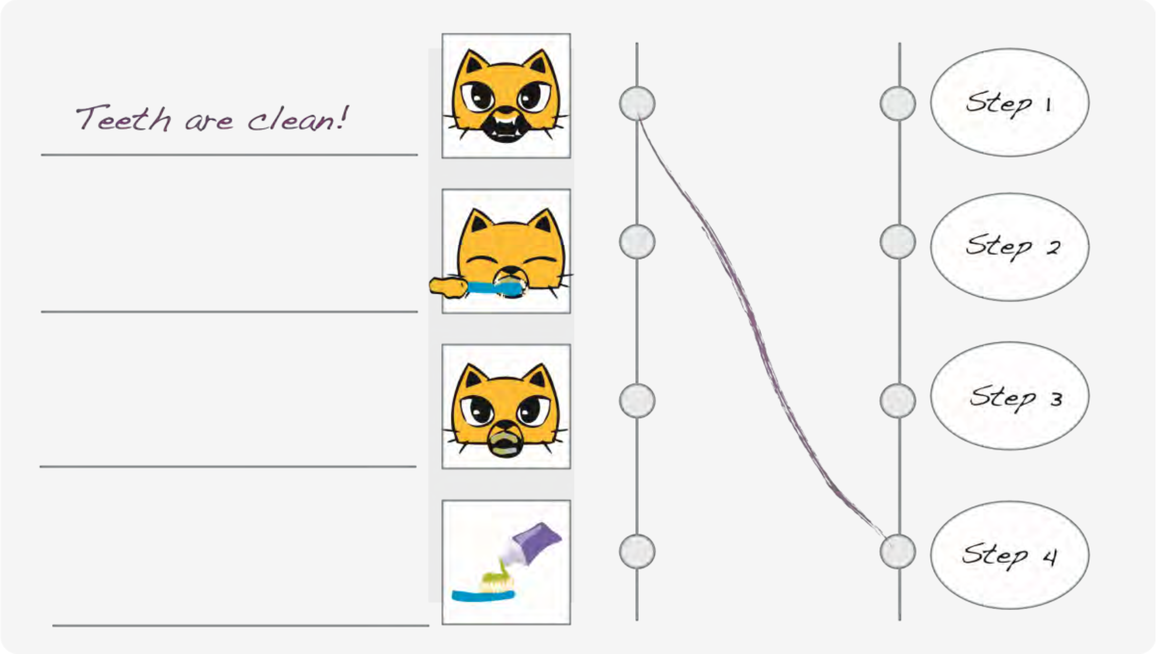
Unplugged

Daily Algorithms

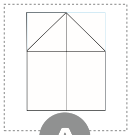
Assessment Worksheet

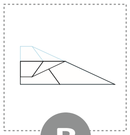
An algorithm is a list of instructions for acomplishing a task. We follow algorithms everyday when it comes to activities like making the bed, making breakfast, or even getting dressed in the morning.

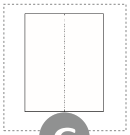
These images are not in order. First, describe what is happening in each picture on the line to its left, then match the action to it's order in the algorithm. The first one has been done for you as an example.

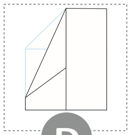


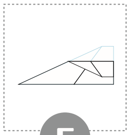
Sometimes you can have more than one algorithm for the same activity. The order of some of these steps can be changed without changing the final product. Use the letters on the images below to create two algorithms for making a paper airplane.

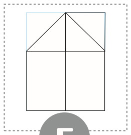

A

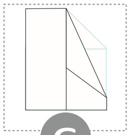

B

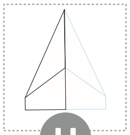

C


D


E


F


G


H

ALGORITHM 1:
ALGORITHM 2:

_______ _______ _______ _______ _______ _______ _______ _______

_______ _______ _______ _______ _______ _______ _______ _______

U

UNPLUGGED

Dice Race

Lesson time: 20 Minutes Basic lesson time includes activity only. Introductory and Wrap-Up suggestions can be used to delve deeper when time allows.

LESSON OVERVIEW

In this lesson, students will relate the concept of algorithms back to everyday real-life activities by playing the Dice Race game. The goal here is to start building the skills to translate real-world situations to online scenarios and vice versa.

TEACHING SUMMARY

Getting Started - 15 minutes

1) Review
2) Vocabulary
3) What We Do Daily

Activity: Real-Life Algorithms - 20 minutes

4) Real-Life Algorithms: Dice Race

Wrap-up - 5 minutes

5) Flash Chat - What did we learn?

Assessment - 10 minutes

6) Daily Algorithms

LESSON OBJECTIVES

Students will:

- Name various activities that make up their day
- Decompose large activities into a series of smaller events
- Arrange sequential events into their logical order

TEACHING GUIDE

MATERIALS, RESOURCES AND PREP

For the Student

- Dice (1 per pair)
- Pens/Pencils/Markers
- Real-Life Algorithms Worksheet: Dice Race
- Assessment Worksheet: Daily Algorithms

For the Teacher

- Teacher Lesson Guide
- Print one Real-Life Algorithms Worksheet per group
- Print one Assessment Worksheet: Daily Algorithms per student

GETTING STARTED (15 MIN)

1) Review

This is a great time to review the last lesson that you went through with your class. We suggest you alternate between asking questions of the whole class and having students talk about their answers in small groups.

Here are some questions that you can ask in review:

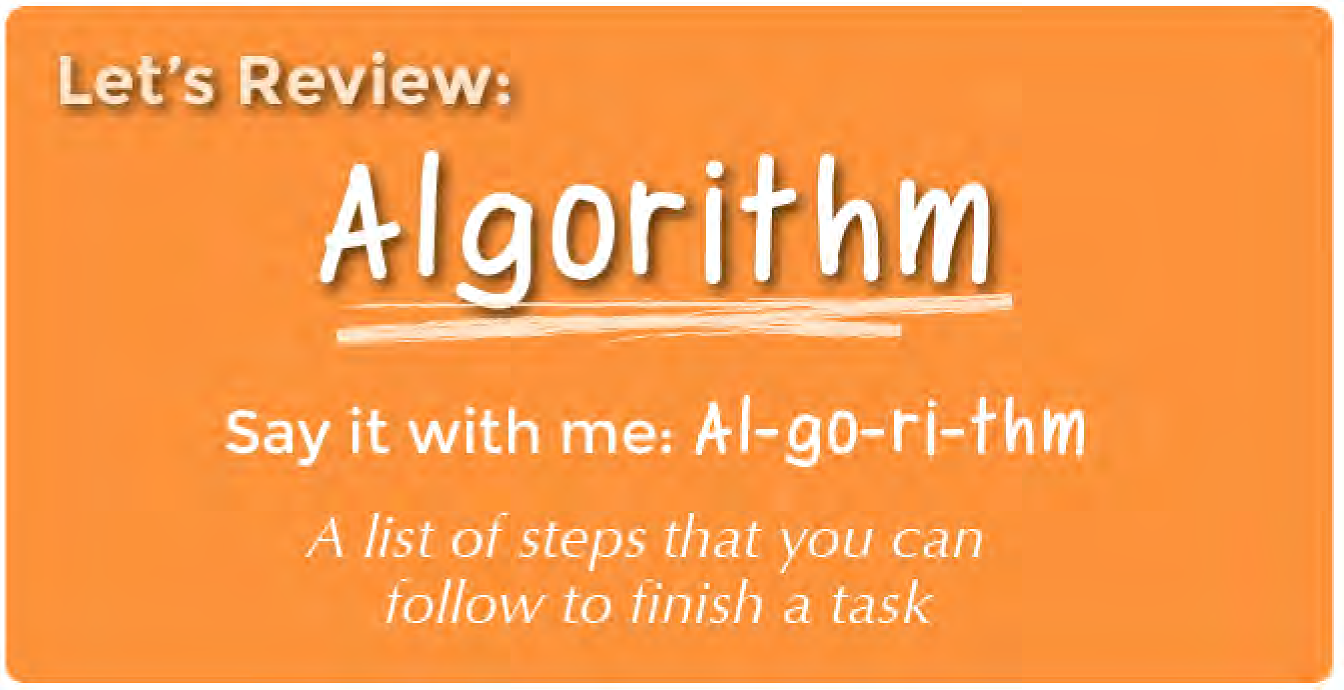
- What did we do last time?
- What do you wish we had had a chance to do?
- Did you think of any questions after the lesson that you want to ask?
- What was your favorite part of the last lesson?

LESSON TIP

Finishing the review by asking about the students' favorite things helps to leave a positive impression of the previous exercise, increasing excitement for the activity that you are about to introduce.

2) Vocabulary

This lesson has one vocabulary word that is important to review:



Let's Review:

Algorithm

Say it with me: Al-go-ri-thm

A list of steps that you can follow to finish a task

Algorithm - Say it with me: Al-go-ri-thm
A list of steps that you can follow to finish a task

3) What We Do Daily

- Ask your students what they did to get ready for school this morning.
    - Write their answers on the board.
    - If possible, put numbers next to their responses to indicate the order that they happen.
        - If students give responses out of order, have them help you put them in some kind of logical order.
        - Point out places where order matters and places where it doesn't.

- Introduce students to the idea that it is possible to create algorithms for the things that we do everyday.
    - Give them a couple of examples, such as making breakfast, brushing teeth, planting a flower, and making paper airplanes.
- Computers need algorithms and programs to show them how to do even simple things that we can do without thinking about them.
    - It can be challenging to describe something that comes naturally in enough detail for a computer to replicate.

- Let's try doing this with a new and fun activity, like playing the Dice Race Game!

ACTIVITY: (20 MIN)

4) Real-Life Algorithm Worksheet: Dice Race

LESSON TIP

You know your classroom best. As the teacher, decide if students should do this in pairs or small groups.

- You can use algorithms to help describe things that people do every day. In this activity, we will create an algorithm to describe how we play the Dice Race Game.
- The hardest part about getting a problem ready for a computer can be figuring out how to describe real-life activities. We're going to get some practice by playing and describing the Dice Race game.

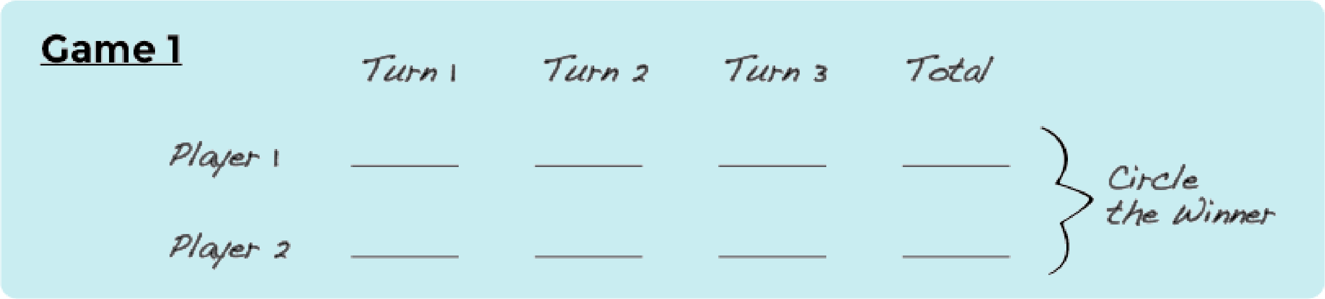
Directions:

1. Read the rules below.
2. Play a couple rounds of the Dice Race game.
    - As you're playing, think about how you would describe everything that you're doing.
    - What would it look like from the computer's point of view?

Rules:

1. Set each player's score to 0
2. Have the first player roll
3. Add points from that roll to player one's total score

4. Have the next player roll
5. Add points from that roll to player two's total score
6. Each player should go again two more times
7. Check each player's total score to see who has the most points
8. Declare Winner



Game 1

	Turn 1	Turn 2	Turn 3	Total	
Player 1	______	______	______	______	Circle the Winner
Player 2	______	______	______	______	

LESSON TIP

Help the students see the game from a computer's point of view. If they need to roll the dice, then the computer needs to provide dice. If the student needs to play three turns, then the computer needs to loop through the steps multiple times.

WRAP-UP (5 MIN)

5) Flash Chat: What did we learn?

- How many of you were able to follow your classmates' algorithms to play the Dice Race Game?
- What's the difference between an algorithm and a program?
  - An algorithm is the thinking behind what needs to happen, while the program is the actual instruction set that makes it happen.
  - An algorithm has to be translated into a program before a computer can run it.
- Did the exercise leave anything out?
  - What would you have added to make the algorithm even better?
  - What if the algorithm had been only one step: "Play Dice Race"?
    - Would it have been easier or harder?
    - What if it were forty steps?
- What was your favorite part about that activity?

ASSESSMENT (15 MIN)

6) Assessment Worksheet: Daily Algorithms

- Hand out the assessment worksheet and allow students to complete the activity independently after the instructions have been well explained.
- This should feel familiar, thanks to the previous activities.

EXTENDED LEARNING

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

Go Figure

- Break the class up into teams.
- Have each team come up with several steps that they can think of to complete a task.
- Gather teams back together into one big group and have one team share their steps, without letting anyone know what the activity was that they had chosen.
- Allow the rest of the class to try to guess what activity the algorithm is for.

CONNECTIONS AND BACKGROUND INFORMATION

ISTE Standards (formerly NETS)

- 1.a - Apply existing knowledge to generate new ideas, products, or processes.
- 1.c - Use models and simulation to explore complex systems and issues.
- 2.d - Contribute to project teams to solve problems.
- 4.b - Plan and manage activities to develop a solution or complete a project.
- 4.d - Use multiple processes and diverse perspectives to explore alternative solutions.
- 6.a - Understand and use technology systems.

CSTA K-12 Computer Science Standards

- CD1:6-06. Recognize that computers model intelligent behavior.
- CPP.L1:6-05. Construct a program as a set of step-by-step instructions to be acted out.

- CT.L1:6-02. Develop a simple understanding of an algorithm using computer-free exercises.
- CT.L2-01. Use the basic steps in algorithmic problem solving to design solutions.
- CT.L2-06. Describe and analyze a sequence of instructions being followed.
- CT.L2-12. Use abstraction to decompose a problem into sub-problems.

NGSS Science and Engineering Practices

- 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.

Common Core Mathematical Practices

- 1. Make sense of problems and persevere in solving them.
- 2. Reason abstractly and quantitatively.
- 4. Model with mathematics.
- 6. Attend to precision.
- 7. Look for and make use of structure.
- 8. Look for and express regularity in repeated reasoning.

Common Core Math Standards

- 4.NBT.B.4 - Fluently add and subtract multi-digit whole numbers using the standard algorithm.

Common Core Language Arts Standards

- 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.
- SL.3.3 - Ask and answer questions about information from a speaker, offering appropriate elaboration and detail.
- L.3.6 - Acquire and use accurately grade-appropriate conversational, general academic, and domain-specific words and phrases, including those that signal spatial and temporal relationships.
- SL.4.1 - Engage effectively in a range of collaborative discussions (one-on-one, in groups, and teacher-led) with diverse partners on grade 4

topics and texts, building on others' ideas and expressing their own clearly.

- L.4.6 - Acquire and use accurately grade-appropriate general academic and domain-specific words and phrases, including those that signal precise actions, emotions, or states of being and that are basic to a particular topic.
- SL.5.1 - Engage effectively in a range of collaborative discussions (one-on-one, in groups, and teacher-led) with diverse partners on grade 5 topics and texts, building on others' ideas and expressing their own clearly.
- L.5.6 - Acquire and use accurately grade-appropriate general academic and domain-specific words and phrases, including those that signal contrast, addition, and other logical relationships.

Created in partnership with
THINKERSMITH®

Name: ______________________ Date: ____________

U

Unplugged

Real-Life Algorithms

Dice Race Activity

You can use algorithms to help describe things that people do every day. In this activity, we will create an algorithm to help each other understand the Dice Race game.

The hardest part about getting a problem ready for a computer can be figuring out how to describe real-life activities. We're going to get some practice by playing and describing the Dice Race game.

Read the rules below, then play a couple rounds of the Dice Race game. As you're playing, think about how you would describe everything that you're doing. What would it look like from the computer's point of view?

The Rules:

1) Set each player's score to 0.
2) Have the first player roll.
3) Add points from that roll to player one's total score.4) Have the next player roll.
5) Add points from that roll to player two's total score.
6) Each player should go again two more times.
7) Check each player's total score to see who has the most points.
8) Declare Winner.

Game 1

	Turn 1	Turn 2	Turn 3	Total
Player 1				
Player 2				

Circle the Winner

Game 2

	Turn 1	Turn 2	Turn 3	Total
Player 1				
Player 2				

Circle the Winner

U

Unplugged

Dice Race

Assessment Worksheet

CO
DE

Use the space below to play through the Dice Race game.

When you're done, use the bottom of the page to create an algorithm (list of steps) that someone else could use to learn how to play.

	Turn 1	Turn 2	Turn 3	Total
Player 1	______	______	______	______
Player 2	______	______	______	______

Circle the Winner

Now, take the steps that you've used to play the game above, and write them down in the slots below. Take advantage of the repeat loop to avoid having to write down instructions more than once.

Step 1 : ______________________________

Repeat 3 times

Step 2 : ______________________________

Step 3 : ______________________________

Step 4 : ______________________________

Step 5 : ______________________________

Step 6 : ______________________________

Step 7 : ______________________________

Lego Robotics

Students worked together to build the first basic model robot. Following the pictorial directions, provided by Lego Mindstorms, the students built the robot. Students rotated positions to give each student the opportunity to build a section of the robot. Students worked as builders, part sorters, part gatherers, and organizers.

http://robotsquare.com/wp-content/uploads/2013/10/45544_educator.pdf

Lego Robotics Tutorials are from the following website:

http://www.stemcentric.com/ev3-tutorial/

Video Tutorials on how to operate the EV3 Robot:

ESSENTIALS

- EV3 Introduction
  Move Exercise More Action Blocks

- Loops
  Loop Exercise

- Wait Blocks
  Wait Exercise

- Light Sensors
  Light Sensor Exercise

- Switch Basics
  Switch Exercise

ADVANCED

- Switches - Advanced
- Multitasking - Running more than one sequence
- Data Wires and Advanced Display Blocks
- Data Operations and Variables

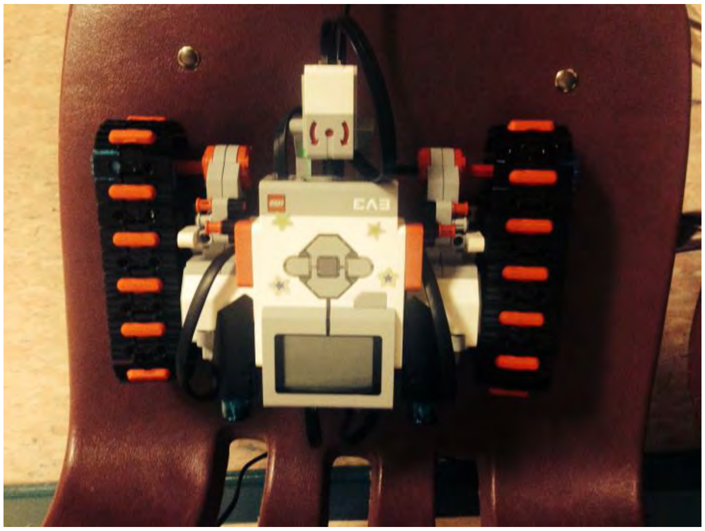
Students were given time to view the videos, ask questions and then work in small groups of three or four members to practice programming the robot. Students would rotate positions: programmer, navigator, trouble shooter, and analyst. Each student was given the opportunity to write their own program, and they could write programs in small groups of two, three or four students. The students were given objectives to achieve such as:

- **Write a program to have the robot display a picture on the screen.**
- **Write a program to have the robot make a sound, speak a word or sentence.**
- **Write a program to have the robot move forward, backward, turn left and right.**
- **Write a program to have the robot perform three or more functions.**
- **Write a program to enable the robot to use a tool or extension.**

Students first built the basic model robot and then added extensions.

Students wrote programs to have the robot complete a task such as pick up an object, move an object or carry an object.

After practicing programming with the basic model robot, the students were allowed to vote on new designs for the robots. They chose the "Puppy" and the "Tankbot."



Materials

EV3 Core Set

$339.95

This core set is optimized for classroom use and contains all you need to teach using LEGO® MINDSTORMS® Education EV3. It enables students to build, program, and test their solutions based on real-life robotics technology. It contains the EV3 Intelligent Brick, a powerful small computer that makes it possible to control motors and collect sensor feedback. The set includes:

- Three interactive servo motors with built-in rotation sensors.
- Color sensor, gyro sensor, ultrasonic sensor, and two touch sensors
- Rechargeable battery and charger
- Ball wheel
- Connecting cables.
- Building instructions.
- LEGO Technic building bricks for creating a vast variety of models

EV3 Expansion Set

$99.95

This set contains a wide range of elements and is an ideal supplement to the EV3 Core Set. It has been designed to enable students to take their experience of robotics to the next level. There are plenty of special elements such as different gears, a large turntable, robot personalization parts, and unique structural elements. These are joined by many extra standard elements such as beams, axles, and connectors. This set helps students build larger and more complex models while at the same time providing extra or replacement elements. The set is optimized for use in the classroom and after-school programs or robotics competitions. It will be delivered in a sturdy and stackable plastic storage bin. Online building instructions for six EV3 models that require the Expansion Set are available in the software.

Websites

(Code Courses and Lessons)
http://code.org/
(EV3 Tutorials)
http://www.stemcentric.com/ev3-tutorial/
(Robot Designs)
http://robotsquare.com/2013/10/01/education-ev3-45544-instruction/
http://robotsquare.com/2013/10/01/lego-mindstorms-ev3-education-expansion-set-45560-instructions/
(Robot Products for Purchase)
https://shop.education.lego.com/legoed/catalog/product.jsp?productId=5003400 (Core Kit)
https://shop.education.lego.com/legoed/catalog/product.jsp?productId=45560 (Expansion Kit)

Bibliography

Code, Code Studio Course 1,2, and 3 Curriculum Guide from http://studio.code.org Created in Partnership with THINKERSMITH www.thinkersmith.org Attribution NonCommercial ShareAlike

STEMcentric, Dale Yocum is founder and director of the Engineering Program at Catlin Gabel School. http://www.stemcentric.com/ev3-tutorial/