Building Opportunities with Teachers in Schools

2019-2020 BOTS Teacher Cohort Capstone Projects

A. Andereck

Lesson Plan One - Unplugged: Eddy Gets His Ball

Game Objective: Toy dog must retrieve his toy ball by following a path laid out using playing cards. This activity helps students learn about algorithms and the job of computer programmers.

Vocabulary: algorithm, sequence, coding, debugging, computer programmer, robot, commands, move forward # card, move backward # card, turn right, turn left

Materials: deck of playing cards, toy dog, toy ball, vocabulary cards, sentence frames with commands

Learning Objective: Students learn computer vocabulary. Students learn to work cooperatively with their peers and the importance of teamwork.

Standard: K-2.CS.3

Lesson: Review vocabulary with students. Vocabulary should be posted and students should be encouraged to use it throughout the lesson. Sentence frames with commands should also be visible.

Students work as a team to design and lay out a pathway using a deck of playing cards. Eddy, the toy dog, is placed at the beginning of the path and his toy ball is at the end of the path.

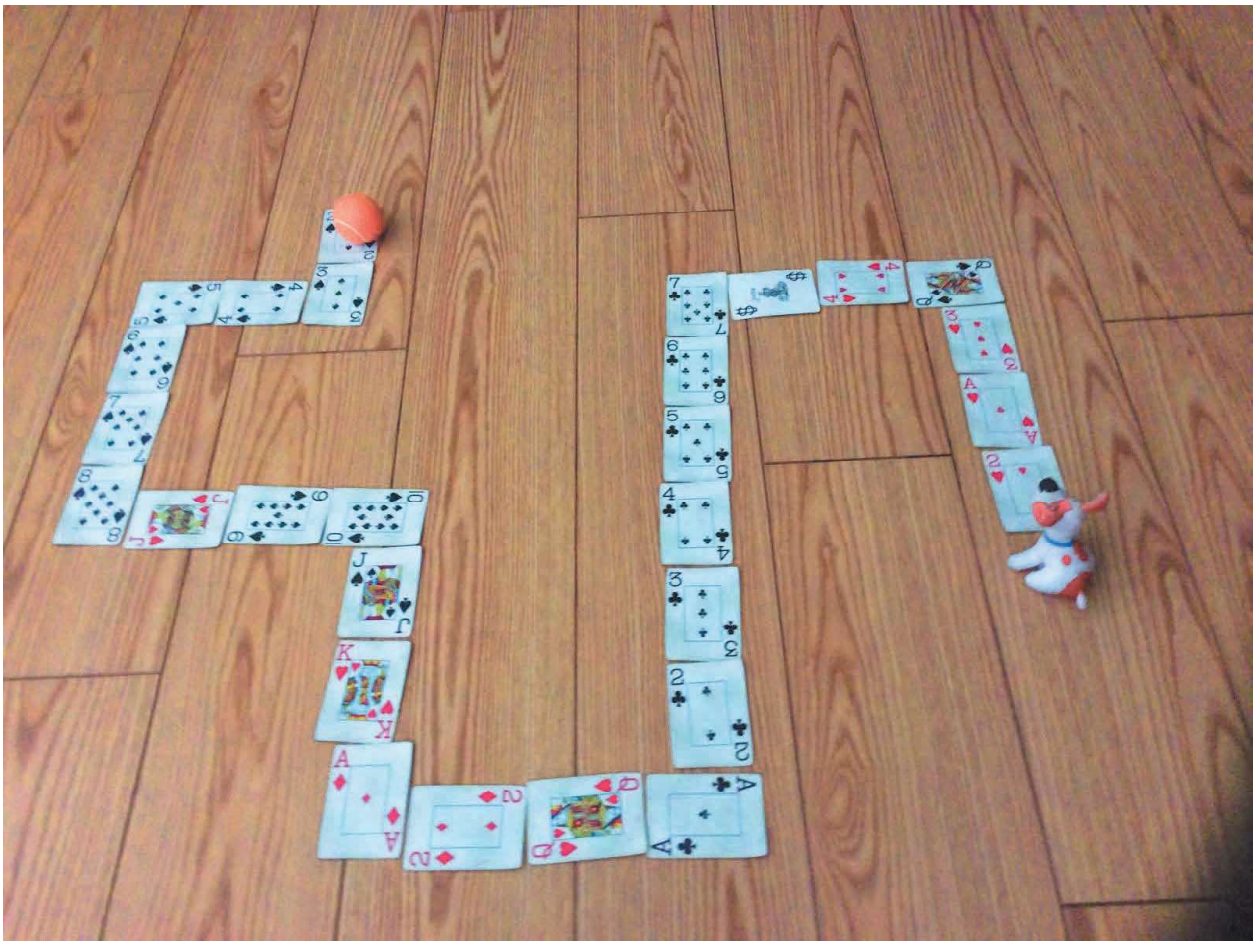
Students work in teams of 2 or 3. One or two students take the role of computer programmer, their job will be to give exact directions to Eddy to retrieve his ball. One or two students take the role of Eddy, and must follow the commands given by the computer programmer.

The student taking the role of the computer programmer uses the following commands to direct Eddy to his toy ball, i.e., 'move forward # card spaces,'

‘move backward # card spaces,’ ‘turn right,’ ‘turn left’. Student taking the role of Eddy must follow commands exactly.

Students can use less cards to simply the game or increase the number of cards used and make a more difficult path. Obstacles can also be used to create a more challenging game.

See attached picture of example game.



A. Andereck

Lesson Plan Two - Unplugged: **Making Binary Numbers**

Objective: Students are introduced to the binary system and learn that the binary system is a language that computers understand. Students learn that the binary number system is a combination of the digits 0 and 1. Information can be expressed through a combination of these two numbers.

Vocabulary: binary numbers, smallest, biggest

Materials: vocabulary cards, set of cards with numbers 1, 2, 4, 8, and 16 dots, paper, pencil

Standards/Math: K.CC.A.3, K.CC.B.4, K.OA.A.1

Lesson: Introduce binary numbers to students. Explain to students that if they flip a card over, showing no dots, that represents the number zero. Cards showing dots of either 1, 2, 4, 8, and 16, represent the number one. Explain to students that we can express any number showing a precise combination of 0s and 1s.

Distribute number cards to students. Have the students place the cards in order, from left to right, 1 being the first

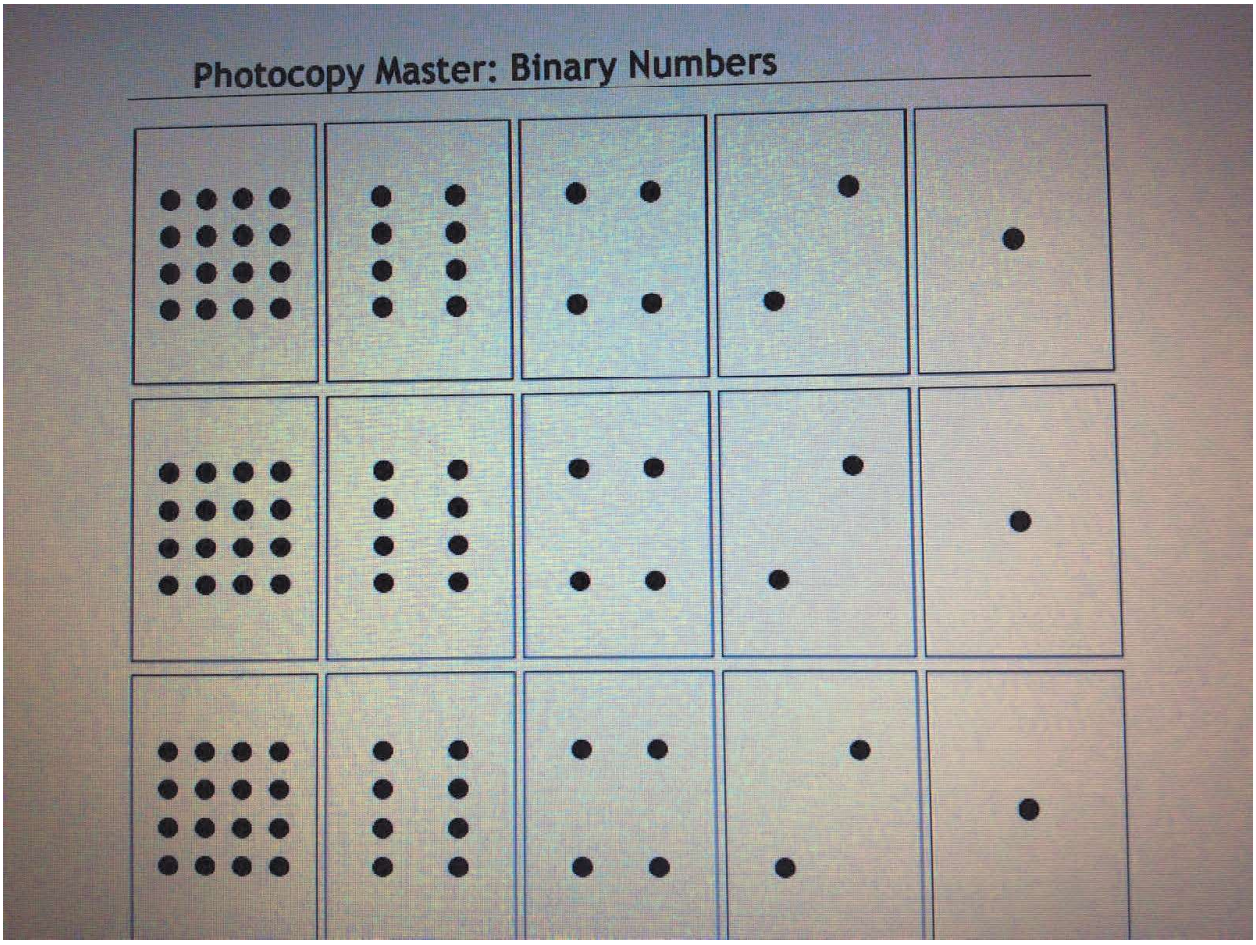
card on the right. Ask the students to show cards to make the number 6, i.e., the cards 2 and 4 are showing, the other cards are flipped over to show the blank side. Have the students translate the binary number on to paper, i.e., 0 0 1 1 0. Students should understand this combination expresses the number 6 in the binary system. This is the binary number system. Continue having the students make different binary numbers and recording their results.

Questions to ask students:

Can there be more than one way to make a number?
What is the smallest number you can make?
What is the biggest number you can make?
How long do you think a binary number can be?

Conclusion: Have students share what they learned about binary numbers. Ask students if they think we can use binary numbers to represent the alphabet and how that might work. Have students share what they know about computers.

Photocopy Master: Binary Numbers



J. Cisneros

2nd Grade

Lesson #1 Plan:

Lesson Name: Pokemon Unplugged

Code.org/Sphero Topic: Loops

Standards Alignment: K-2.AP.12 Create programs with sequences of commands and simple loops, to express ideas or address a problem.

Overview: Using a set of symbols in place of code, students will design algorithms to instruct a "robot" (sphero) to collect pokemon on a board game. Students will take turns participating as the robot (moving the paper manipulative along the board), responding only to the algorithm defined by their peers. This segment teaches students to identify when loops can be used to simplify the code and reach their objective.

Learning Objectives - Students will be able to:

- Attend to precision when creating instructions
- Identify and address bugs or errors in sequenced instructions

  Identify and add a loop when appropriate

Objectives (Standard): I can identify when a loop is needed and add it appropriately to my code for my sphero

Formative Assessment: Create a code (program) without bugs and 1 loop to catch your pokemon

Estimated Time: 45 minutes

Materials:

- board game (sheet with grid that imitates angry birds sphero grid)
- Display or handouts of the symbols from My Robotic Friends Symbol Key where students can reference throughout the lesson
- Game piece that will play the part of the sphero for the scholar following their peer's code
- Pencil/ grid paper to write code with loop reference for them to add to their code

Lesson Guide:

Warm Up (5 min)

Hook/Activating Prior Knowledge (APK): Check for understanding: students will share with their partner and class What a bug is in a code and how to get rid of them if they come across one. Teacher will scaffold discussion on loops for partners to discuss further

Say: Robots can only do what they've been told to do, but we don't just tell them using words. In order to do something, a robot needs to have a list of steps that it can read. Today, we are going to learn what it takes to make that happen.

Activity (30 min)

Introduction and Modeling

Set Up: Board game sheets out and board game piece to move (a standing pokeball students will color while their peer creates a code for it)

Display: Display My Robotic Friends - Symbol Key or write the allowed actions on the board - make sure these are in a place where they can be seen for the whole activity. Explain to the class that these will be the only four actions that they can use for this exercise.

Model/Whole Class:

Teachers will model what will be done in small groups: will model how to add a loop into the code.

Display: Hold up the pattern you plan to model. A simple one loop program will warm them up to create more.

Prompt: Ask the class what the first instruction should be, using only the four instructions allowed. The first move should be to "pick up the piece." If students suggest something else from the list, perform that action and allow them to see their error. If they suggest something not from the list, make a clear malfunction reaction and let them know that the command is not understood.

Continue asking for instructions from the classroom until you have completed the entire design. Point out the repetitive parts to the class. That can be shortened into a loop. A loop is like turning a line into a loop saving “space” it is the same thing with the code. We are turning this line of code into a loop to shorten the code and make it easier to both create it and read it.

Ask the class to help you write a "loop" in the code where it is repetitive. Then work with them to write down the rest of the moves necessary to complete the loops. Depending on the confidence of your students, you might switch back and forth frequently between acting as the "robot" and writing down the code, or you might push them to write the whole program before you will implement it.

Programming Your Robots

Group: Place students into groups of 4. Each group should then further break down into two pairs - each pair will develop their own program to be "run" by the other pair.

Distribute: Give each group a game board and a “pokeball cut out to color”.

Display: Show My Robotic Friends - for code guidance

Guided/Partner: Class will be divided in half. Half of them will be with the teacher creating a code for the floor game while identifying and looping in their code to catch as many pokemon as they can.
This will lead into them creating a program with at least one loop for their Sphero

Instructional Technology: The other half of the class will work on code.org: lesson 11 independently or in pairs if they are behind on code.org (partners will be assigned to support students that are behind on code, so they can catch up to lesson 10

Independent: exit ticket: code lesson 10, Program for Sphero

When you were the robot, what was the hardest part of following the instructions you were given?

Wrap Up (10 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.
Journal Prompts: Draw one of the Feeling Faces - Emotion Images that shows how you felt about today's lesson in the corner of your journal page.

Example Lesson #2 Plan:

Lesson Name: **Pokemon - Spelling Words**

Code.org/Sphero Topic: Loops Debugging

Standards Alignment:

CSTA K-12 Computer Science Standards (2017). AP - Algorithms & Programming - 1A-AP-09: Model the way programs store and manipulate data by using numbers or other symbols to represent information.

L.2.1f Produce, expand, and rearrange complete simple and compound sentence

Overview: Given a maze with several spelling words on different paths, students will program their sphero to follow their desired path and create a story with vocabulary words picked up on their path. Students may take turns programming parts of the path, selecting to follow one or two words of their choice. This teaches reading comprehension and storytelling to students, whilst enforcing the topic of angles/degrees.

Learning Objectives - Students will be able to:

- Cooperatively program their sphero through a path
- Create short stories from a pool of spelling words
- Create a program with loops and debug if necessary

Estimated Time: 45 minutes

Materials:

- Paper tiles
- Spelling words
- Sphero(s)
- Sphero Edu app on compatible device(s)
- White boards *optional*

Lesson Guide:

Warm Up (5 min)

Hook/Activating Prior Knowledge (APK): Check for understanding: students will share with their partner and class what a bug is in a code and how to get rid of them if they come across one. The teacher will scaffold discussion on loops for partners to discuss further. Students will review commands with their partners using my robotic friends.

Activity (30 min)

Introduction and Modeling

Set Up: Paper tiles out and sphero

Display: Display Symbol Key or write the allowed actions on the board - make sure these are in a place where they can be seen for the whole activity. Explain to the class that these will be the only four actions that they can use for this exercise.

Model/Whole Class: Teachers will model what will be done in small groups: will model how to add a loop into the code. Students will be divided in groups of four for each tile set up. Students will be given one sphero and one compatible device to program with. *Each group should also be given 3 white boards to simulate code and fix any bugs.

Guided/Partner: Class will be divided in half. Half of them will be with the teacher creating a code for the floor game while identifying bugs and adding loops in their code to catch as many pokemon as they can.

This will lead into them creating a program with at least one loop for their Sphero. For each Pokemon they collect, they also collect a spelling word from that week; collecting several spelling words at the end. At the end of the lesson each student will write sentences for each spelling word ultimately writing a short silly story.

Two students in each group are to create a program for the sphero to collect all pokemon with spelling words while the other 2 students will create their own code on whiteboards to support their peers if they get stuck on the program or need to fix a bug.

Instructional Technology: The other half of the class will work on code.org: lesson 11 independently

Wrap Up (10 min)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future. Journal Prompts: Draw one of the Feeling Faces - Emotion Images that shows how you felt about today's lesson in the corner of your journal page.

Y. House

Lesson #1 Plan:

Lesson Name: Valentine Unplugged Cupid Challenge

Code.org/Sphero Topic: Loops

Standards Alignment: K-2.AP.12 Create programs with sequences of commands and simple loops, to express ideas or address a problem.

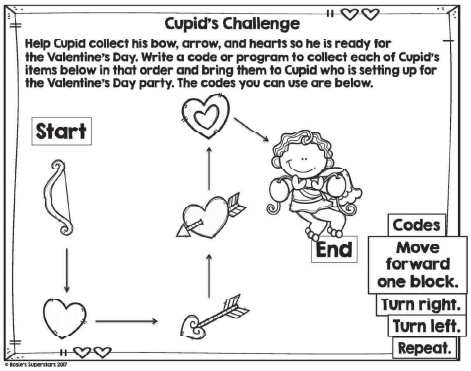
Overview: Using a set of commands (program) in place of code, students will design algorithms to direct a "Cupid" to collect different items in a specific order. Students will work with a partner. Together they will come up with the program for cupid to collect all his items. This segment teaches students to identify when loops can be used to simplify the code and reach their objective.

Learning Objectives - Students will be able to:
Write commands to collect all of Cupid's items in the given order.

- Attend to precision when creating instructions
- Identify and address bugs or errors in sequenced instructions
  Identify and add a loop when appropriate

Estimated Time: 45 minutes

Materials:
- Cupid's Challenge worksheet
- White boards and expo markers
- Pencils, crayons, markers
- Projector for projecting the codes that can be used. (Move forward one block, turn right, turn left, repeat)



Lesson Guide:

Warm Up (5 min)

Hook/Activating Prior Knowledge (APK): Check for understanding: We will review which direction is up, down, front, and bac. Students will share with their partner and class what a bug is in a code and how to get rid of them if they come across one. Teacher will scaffold discussion on loops for partners to discuss further

Say: Robots can only do what they've been told to do, but we don't just tell them using words. In order to do something, a robot needs to have a list of steps that it can read. Today, we are going to learn what it takes to make that happen.

Activity (30 min)

Introduction and Modeling

Set Up: Pass out the Valentine Cupid Challenge worksheets. Students will go with their partners.

Display: Project the Valentine Cupid Challenge worksheet - Code Key highlight the allowed actions on the board - these will be projected on the white board so that they can be seen throughout the activity for reference. I will explain to the students that these will be the only four actions that they can use for this activity.

<u>Model/Whole Class:</u>

- Teachers will model what will be done in small groups: will model how to add a loop into the code.

Display: Hold up the pattern you plan to model. A simple one loop program will warm them up to create more.

Prompt: Ask the class what the first instruction should be, using only the four instructions allowed. The first move should be to "move forward" to pick up the bow and arrow. If students suggest something else from the list, perform that action and allow them to see their error. If they suggest something not from the list, make a clear malfunction reaction and let them know that the command is not understood.

Continue asking for instructions from the students until they have completed the entire challenge. Point out the repetitive parts to the class. That can be shortened into a loop. A loop is like turning a line into a loop saving "space" it is the same thing with the code. We are turning this line of code into a loop to shorten the code and make it easier to both create it and read it.

Ask the students to help you write a "loop" in the code where it is repetitive. Then work with them to write down the rest of the moves necessary to complete the loops. Depending on the confidence of your students, you might switch back and forth frequently between acting as the "cupid" and writing down the code, or you might push them to write the whole program before you will implement it.

Programming Your Robots

Group: Place students into groups of 4. Each group should then further break down into two pairs - each pair will develop their own program to be "run" by the other pair.

Distribute: Give each group a game board and a “game piece cut out to color”.

Display: Show My Robotic Friends - for code guidance

Guided/Partner: Class will be divided in half. Half of them will be with the teacher creating a code for the floor game while identifying and looping in their code to catch as many Cupid items as they can.
This will lead into them creating a program with at least one loop for their Sphero

Instructional Technology: The other half of the class will work on code.org: lesson 11 independently or in pairs if they are behind on code.org (partners will be assigned to support students that are behind on code, so they can catch up to lesson 7

Independent: exit ticket: code lesson 7, Program for Sphero

When you were the robot, what was the hardest part of following the instructions you were given?

Wrap Up (10 min)

Journaling- Reflection

I will ask the students to write about what they learned.I will provide some questions to help them reflect. Why was it helpful to work with a partner? How did you feel when a certain move didn’t work? How did you feel when your cupid collected all the items and made it to the end? Draw a picture to go with your writing.

Example Lesson #2 Plan:

Lesson Name:Cupid's - Addition and Subtraction True or False

Code.org/Sphero Topic: Angles

Standards Alignment: K-2.AP.12 Create programs with sequences of commands and simple loops, to express ideas or address a problem.

1.0A.D7 - Understand the meaning of the equal sign, and determine if equations involving addition and subtraction are true or false. For example, which of the following equations are true and which are false? 6 = 6, 7 = 8 - 1,5 + 2 = 2 + 5, 4 + 1 + 5 + 2.

Overview: Given a maze with several Valentine images on the different paths, students will program their sphero to follow their desired path to determine if the addition and subtraction equation task card they pick up along the way on their path is true or false. Students may take turns programming parts of the path. This will give the students practice determining if addition and subtraction equations are equal or not (true or false) enforcing the topic of angles/degrees.

Learning Objectives - Students will be able to:

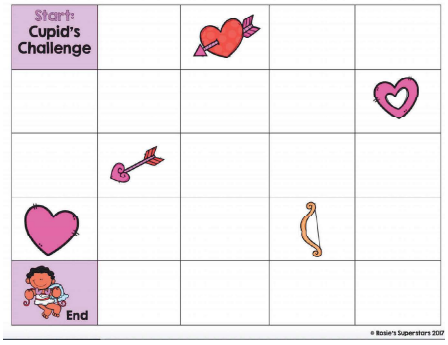
I can determine if the addition and subtraction equations are true or false. 1.0A.D7

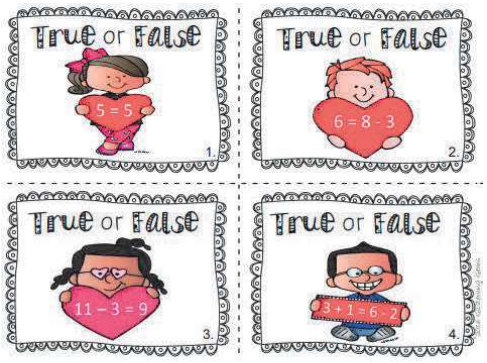
- Cooperatively program their sphero through a path
- Practice with understanding the meaning of the equal sign.
- Determining if the addition and subtraction sentences are true of false.

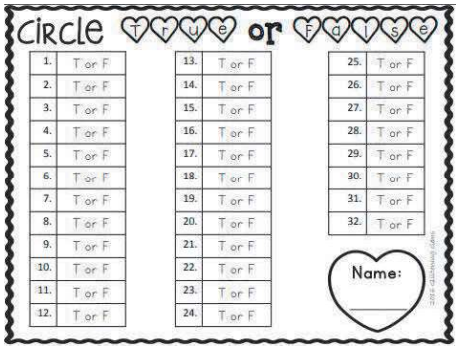
Estimated Time: 45 minutes

Materials:

- Paper tiles- Cupid's Challenge
- Valentine's day math true or false task cards
- Sphero(s)
- Student response sheet
- Sphero Edu app on compatible device(s)
- White boards *optional* for solving equations
- Sphero Edu app on compatible device(s)







Lesson Guide

Warm Up (5 min)

Hook/Activating Prior Knowledge (APK): Check for understanding: students will share with their partner and class what a bug is in a code and how to get rid of them if they come across one. The teacher will scaffold discussion on loops for partners to discuss further. Students will review commands with their partners.

Activity (30 min)

Introduction and Modeling

Set Up: Cupid's Challenge - Paper tiles out and sphero

Valentine's day math true or false task cards

Display: Display Symbol Key or write the allowed actions on the board - make sure these are in a place where they can be seen for the whole activity. Explain to the class that these will be the only four actions that they can use for this exercise.

Model/Whole Class: Teachers will model what will be done in small groups: will model how to add a loop into the code. Students will be divided in groups of four for each tile set up. Students will be given one sphero and one compatible device to program with. *Each group should also be given 3 white boards to simulate code and fix any bugs, and solve addition and subtraction equations to determine if they are equal or not.

Guided/Partner: Class will be divided in half. Half of them will be with the teacher creating a code for the floor game while identifying bugs and adding loops in their code to determine as many true/ false equations as they can.

This will lead into them creating a program with at least one loop for their Sphero. For each valentine image they go past they will also collect a true or false task card. At the end of the lesson each student will circle **T or F** on their response sheet for every task card they collected along the way.

Instructional Technology: The other half of the class will work on code.org: lesson 8 independently

Wrap Up (10 min)

Journaling - Reflection

I will ask the students to write about what they just did. I will provide some questions to help them reflect on the lesson. What did you like about it? What didn't you like about it? Did you learn anything new? Draw a picture to go with your writing.

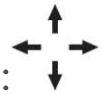
A. Lopez

Unplugged Activity

Lesson Name: Debugging the Maze

Code.org/Sphero Topic: Debugging

Standard Alignment: Algorithms and Programming 1A-AP-09: Model the way programs store and manipulate data by using numbers or other symbols to represent information.

Overview: Students will create an algorithm using these symbols:  . Students will create an algorithm with bug(s) based on a maze. When students are paired up, one of the students must identify what the bug is and debug it.

Learning Objectives:

- Identify and address bugs in instructions.
- Attend to precision when creating instructions.

Estimated time: 1 hour

Materials:

- Paper and pencil so students can write their algorithms using the arrows.
- One premade maze with16 empty white squares for each student.
- Tiles that will cover the squares. Tiles will either be white tiles, tnt tiles, start tile, and star or goal tile.

- Baggies for the tiles.    
  
- One blue round chip.

  Lesson Guide:
  Warm Up: Review arrows. You can choose a youtube video that reviews directionality. Also review the term debugging

  Activity: 50 minutes

  Students will use the tiles to make a maze and purposefully add a bug or bugs to the maze. Next, students will write down their algorithm exactly as seen on the maze.

  Students will then be paired up with only one maze and one set of tiles and blue chip.

  20 Minutes (10 minutes each student)

  Student A will give the maze with the tiles on it along with the algorithm paper. The job of Student B is use the chip and move the chip exactly as the algorithm states. Student B must identify the error by circling the arrow that either needs to be deleted or added an arrow. Student B must then rewrite the algorithm so that the blue chip successfully moves from the start chip to the star chip.

Teacher is circulating the classroom and asking question and observing.

Now it's Student B's turn to do the same thing.
20 Minutes (10 minutes each student)

Teacher will ring the bell after 30 minutes and clean up. Then teacher will ask student A to pair up with another student from student B and do the same thing.

If time permits, a students can share what they learned from their experiences.

Extension: Are there any loops in your algorithm? How can you include them next time?

Plugged Activity

Lesson Name: Sphero Art with Jackson Pollack

Sphero Topic: Algorithms, Loops, Sphero Variables

Overview: Students will create art inspired by Jackson Pollack using the spheros. They will program the sphero using different colored paints on a white canvas or paper to create their masterpiece. Will their paintings be simple or complex? Does adding more algorithms make your art more complex or less?

Learning Objectives:

- Attend to precision when creating an algorithm.
- Create art by programming the sphero.

Estimated time: 1 hour

Materials:

- Tempera paint of any color.
- Large aluminum tray (depth 3 3/8 inches, top length 20 ¾ inches, top width 12 ¾ inches)
- White paper that fits inside the tray with ½ inch border left.
- Spheros – one for two students if possible
- Ipad or phone to program spheros.
- Journal and pencil
- Paper plates

Lesson Guide:

Warm Up 7:42 minutes
Students watch youtube video *Art with Mati and Dada*. Students will be exposed to Jackson Pollack's art to get them thinking about how they can create art similar to Pollack's using spheros .
Also review the angles such as 0°, 90°, 180°, and 270°

Activity: 20 Minutes (each student)

Teacher: Make sure you have set up materials before you begin the lesson. Each pair of students should have tray, white paper, and 3-4 different colored paints on paper plates. Also remind students to write their name on the bottom of the canvas before starting.

Pair students up. Student A will program the sphero (roll, speed, seconds). The student should write down the commands. Student B can help student A. Don't forget about the delay button when changing directions. Make sure that the sphero uses the same specifications in terms of measurement with the size of the canvas. Also make sure you properly aim the sphero. Next student A will decide which color he/she will choose next and dip the sphero into the paint and place on the paper. Continue until all the colors have been used.

20 Minutes

Student B programs the sphero and begins their masterpiece just like student A.

Wrap Up: 5 Minutes
Journal: Choose one
How were you inspired by Jackson Pollack?
How is your art same/different than one from Jackson Pollack?
What would you name your masterpiece?

After the the paint dries, add the title onto a flashcard and attach it to the bottom of your art and display in the classroom.

Extension:

To be done at a different date.

Since students wrote down their commands for the spheros, have students exchange them with another student. Using the same materials (tray, paint, canvas paper) will a student be able to replicate their partners painting by following the same commands? Attending to precision will be important.

Desiree Luna
St. Odilia School
2nd Grade

BOTS Capstone Project 2019–2020

Unplugged Activity

Lesson Name: Looping for Cheeseburgers.

Code.org/Sphero Topic: Loops

Objectives:

Students will be able to:

- Identify repeated patterns in code that could be replaced with a loop
- Write instructions that use loops to repeat patterns

Standards:

CSTA K-12 Computer Science Standards (2017)

- AP - Algorithms & Programming

Cross-curricular Opportunities

Common Core English Language Arts Standards

- L – Language
- SL - Speaking & Listening

Common Core Math Standards

- MD - Measurement And Data
- MP - Math Practices

Next Generation Science Standards

- ETS - Engineering in the Sciences

Overview: This lesson will help the students prepare for coding loops over the length of this course. In small groups, the students will work together in a physical activity where they will create a code for a program.

Purpose: Through the use of physical activity, the students will come across issues and emotions that will be similar to those they will encounter when programing in coding. By using hands-on approach, the students will alleviate the tensions, build confidence, and acquire necessary skills to solve problems on their own.

Estimated Time: 40 minutes

Desiree Luna
St. Odilia School
2nd Grade

Materials:

- Paper
- Pencil
- Cut out of buns, hamburger meat, and cheese

Lesson Guide:

Warm-Up: Review vocabulary. Give an example of a loop

- Loop - The action of doing something over and over again.
- Repeat - Do something again

Goal: For the warm-up, I want the students to know the meaning of the words given. It is important for me that they use these words in their vocabulary when discussing patterns that are repeating. I also would like for them to recognize when something is being repeated and identify where the loop should be placed.

Discuss: Define the words and give an example of a loop. Have a student come up to the class and create a loop, and action that he is repeating, and have the rest of the class identify what the loop is.

Activity: Looping for cheeseburgers

- Place the students into groups of 6.
- Distribute the materials to each group.
- Assign each student a task.

Directions to class:

1. Explain that each person has a job, stick to your job unless another team member asks for help. (2 navigators, 1 conductor, 2 writers, and 1 team manager.)
2. As a class, discuss the steps (code) for making a cheeseburger.
3. In their teams, the navigators will instruct the conductor on how to make the cheeseburger, giving 1 step at a time.
4. The writer will write down the steps (code) for making a cheeseburger.
5. The team manager will tell the team "we need to make 10 cheeseburgers, how can we do this using the same steps?"
6. The writer will write the code that the team decides would be best for repeating the steps.
7. Lastly, ask the students how these steps will change if they were to add extra ingredients: lettuce, tomatoes, pickles, etc.

Desiree Luna
St. Odilia School
2nd Grade

Repeat ____ times.

Bun
Meat
Cheese
Bun

Cross-curricular Activity

Lesson Name: Adding with my robotic friend.

Code.org/Sphero Topic: Addition with regrouping, angles, and degrees.

Objectives:

Students will be able to:

- Add the equation using regrouping.
- Program the sphero to reach the finish line.

Standards:

CSTA K-12 Computer Science Standards (2017)

- AP - Algorithms & Programming

Cross-curricular Opportunities

Common Core English Language Arts Standards

- L – Language
- SL - Speaking & Listening

Common Core Math Standards

- CCSS.MATH.CONTENT.2.NBT.B.7
- CCSS.MATH.CONTENT.2.NBT.B.7

Overview: With the given maze that has addition equations, the students will program their sphero to follow the path and solve the addition equations found along the path. Students will take turns programming the sphero and solving an equation. This reinforces the skill of adding with regrouping, while also reinforcing the topic of angles and degrees.

Desiree Luna
St. Odilia School
2nd Grade

Purpose: In this lesson, the students are expected to complete the maze while solving the addition problems that they come across.

Estimated Time: 40 minutes

Materials:

- Paper
- Pencil
- Paper tiles
- Sphero app on IPad
- Poster displaying degrees and angles.

Lesson Guide:

Warm-Up: Review degrees song, and addition with regrouping.

- Where I face is 0 degrees, shuffle to right its 90, I go back 180, shuffle to my left 270.

Set-Up:

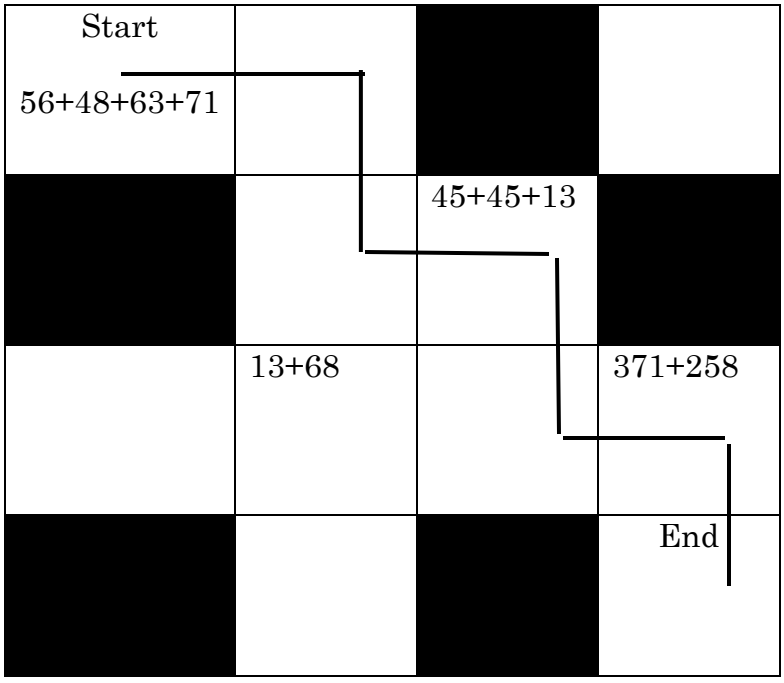
Using the paper tiles, set up a maze for each group. Match the difficulty of the maze to the abilities of each group (this will be done during rotations). An example of a set up will look like the figure below.

Start 56+48+63+71			
		45+45+13	
	13+68		371+258
			End

Desiree Luna
St. Odilia School
2nd Grade

Math maze with sphero: Once in their groups, the students will be able to take turns solving and programming with sphero. The goal of this activity is to get the sphero to a tile that has an equation. Once a student has arrived at an equation, they will solve it and then give another student a chance to move on.

Example path:



Adriana Montijo
Murchison St. School
1st first grade

Lesson Name: Real Life Algorithms / Life Science- plant life

Code.org: Algorithms

Standards Alignment: CSTA K-12 Computer Science Standards (2017).
Algorithms & Programming **-**

Overview: In preparation of this lesson we watched a video, **Plant a Seed.** I then introduced the new vocabulary word **Algorithm.** I explained that algorithms are steps to do something and algorithms are part of our daily life. We can write down the steps of an event or activity we do.

Learning Objectives - Students will be able to:

- Decompose large activities into a series of smaller events
- Arrange sequential events into their logical order

Estimated Time: 60 minutes I broke up the lesson in two days

Materials:

- cups
- Vocabulary words
- Soil
- Seeds
- Real-life algorithms: Plant a Seed Worksheet
- Assessment worksheet

Name: _____ Date: _____

U

Unplugged

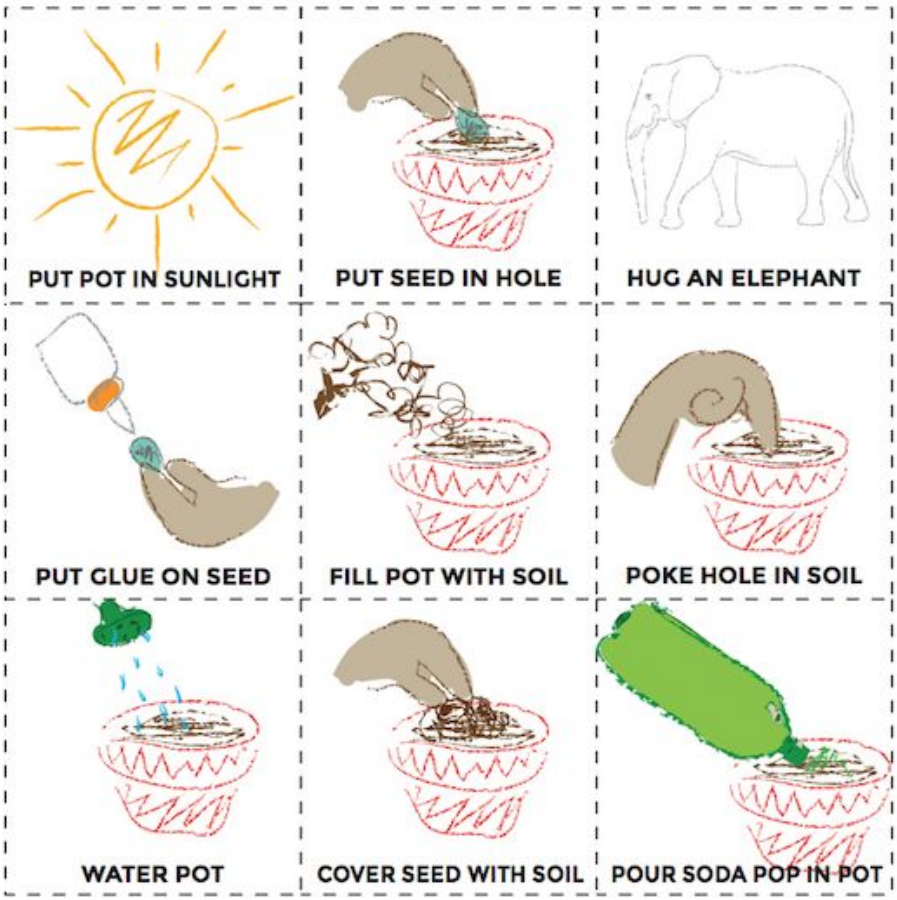
Real-Life Algorithms

Plant a Seed Worksheet

C O D E

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 plant a seed.

Cut out the steps of planting a seed below, then work together to glue the six 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 plant their seed!



Warm-up 5 min

Discussed the new vocabulary with students

Algorithm-break up the word into syllables and maybe the sounds they hear.A list of steps that you can follow to finish a task

Ask your students what they did to get ready for school this morning.

Wrote their answers on the board and discussed the order things were done. I did this with a few other activities and focused on the steps it took to complete the activity

Introduced students to the concept that it is possible to create algorithms for the things that we do daily.

Activity 20 minutes

Students and I discussed how algorithms help describe things that people do every day then we did the worksheet. Cut out the steps for planting a seed from the worksheet.Students worked together to choose the six correct steps from the nine total options.Then they glued the six correct steps, in order, onto a separate piece of paper.

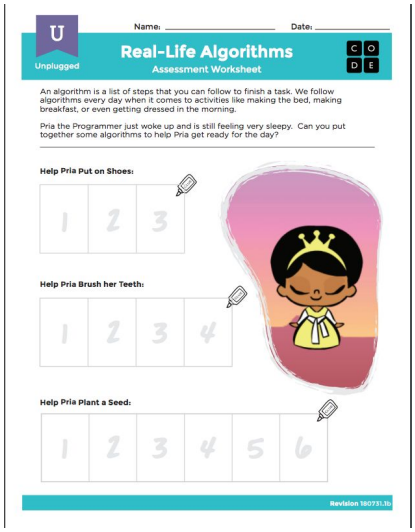
Next Day 20 minutes

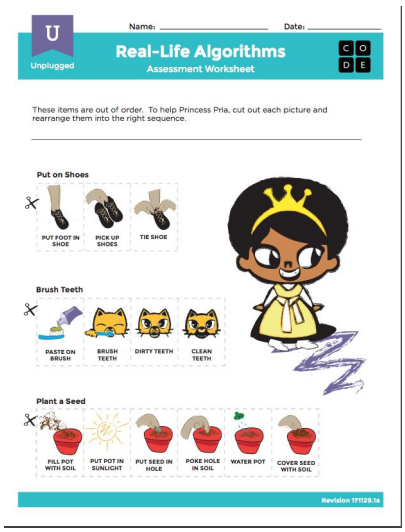
We traded our glued worksheet with another group and we talked about the step of planting a seed. Then they planted their seed according to the step on the sheet.

Warm up 15 minutes

We were able to discuss the steps of planting a seed. I had students use their journal to write what they learned. Those that had difficulty writing drew a picture. They all had to write the new vocabulary word in their journal.

Then I gave them an assessment they had to do on their own.





Lesson Name: My Robotic Friend
Code.org/Sphero Topic: Algorithms
Standards Alignment: CSTA K-12 Computer Science Standards (2017).
Algorithms and Programming
Overview Students will pretend to take turns being robots and have a set of symbols for instructions. Each student will take turns guiding each other on completing a task. They have to guide their robot by using symbols just like in programming. They will learn a new vocabulary word **Debugging** and that would mean fixing a mistake. This is in Course C but since I did it last year I figured it would be fun and I also cut it short and made them do simply stacking.
Learning Objectives - Students will be able to:
● Learn to make a connection between activities and verbal instructions
● Identify and address bugs or errors in sequenced instructions

Time: 60 minutes broke it up into 2 days

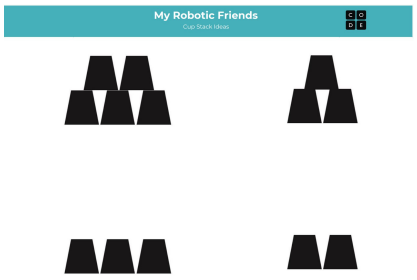
Materials:
●Stack of 20 paper cups (or paper trapezoids) for each group of 2-3 students.
●Handouts of the symbols from My Robotic Friends Symbol Key students used the sheet to reference throughout the lesson as the pictures indicate.
● One My Robotic Friends - Cup Stacking Ideas handout per group of 2 students.
● Journal or notebook for each student.

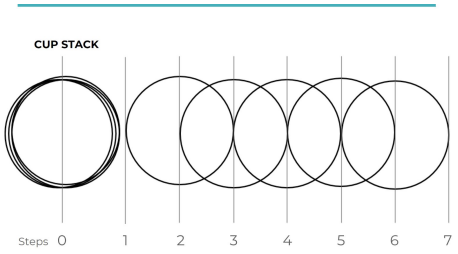
Warm Up (5 min)
Talking to Robots Display: Watch the video, My Robotic Friend below to give students context for the types of things that robots can do

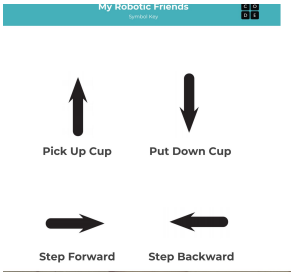
Lesson- 10 minutes
Day 1 -10-15 minutes
I presented the lesson first as a whole class. I picked one student to give me directions on how to stack the cups using the symbols. He wrote the symbols on the board. We discussed the new term debugging and talked about the mistakes I made or the programmer made. It was challenging with first graders. I did this with one more student so I could model what I wanted them to do. We discussed some mistakes and referred to that as a bug and how we needed to debug.
Day 2 -20 minutes
We discussed the symbols and the goals of the lesson. I reminded them one was the robot and the other was a programmer writing down the symbols/directions. I passed out the materials and some of the students still had a difficult time understanding that the arrows meant one step. Having one adult in the room makes it very difficult.

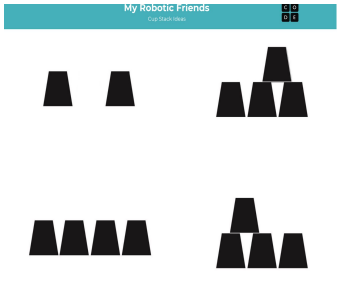
Wrap-Up 10

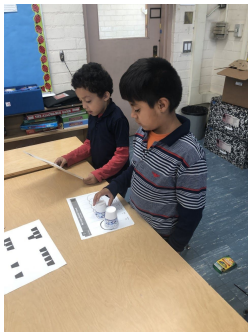
Journaling - Students wrote in their journals of the activity. Some drew pictures of themselves as robots with their friends. They drew the symbols and the cups. I asked them to copy Debugging and what they understood about the new word. They also drew/ wrote how they felt about the lesson.

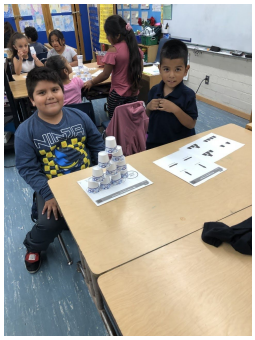












Palmira Reyes
Second Grade
Sheridan Street Elementary
May 30, 2020

Lesson Name: "Find the Buildings"
Standards Alignment: CSTA K-12 Computer Science Standards AP- Algorithms & Programming 1A-AP-09: Model the way programs store and manipulate data by using numbers or other symbols to represent information.
Social Studies Standard: 2SS2: Geography Students demonstrate map skills by describing the absolute and relative locations of people, places, and environments.

Overview: This unplugged lesson will integrate our Social Studies Alive! Lesson 3 to use as an unplugged activity to prepare students for the coding exercises they will be participating in during the course of our Code.org lessons. Students will work in whole group as well as with their seat partner to navigate their way through a map.

Learning Objectives- Students will be able to:

- Cooperatively work together to help each other physically navigate their way through a map
- Listen to and follow a list of defined steps to perform a series of physical actions to reach their goal
- Define a list of steps, (algorithm), to get their partner from their starting position to a selected destination
- Identify and fix errors in the execution of an algorithm

Estimated time: 45 minutes

Materials:

- Social Studies Workbook
- Butcher Paper to draw large scale map from workbook p.12
- Blind- fold
- Pencils
- Journal

Warm-Up, (12 minutes): Have students sit in a circle on the carpet around a large scale map of p.12 Lesson 3 map from our Social Studies Alive! workbook. State the objective of the lesson: In this lesson, we will work with a partner to navigate our way

through a map. We will use a compass rose and the words North, South, East and West to move around the map. If we make any mistakes, we will "debug" or fix our program or route. Discuss the rules first:

1. Start at the designated starting points
2. Follow the instructions/directions step by step
3. End at the designated destination
4. Fix or "debug" your mistakes

Teacher will model by creating a program, or list of instructions that contains a "bug" or mistake. She will ask students to help her figure out what went wrong on her way to the Grocery Store. Discussion: What happened? Where did Ms. Reyes go wrong? How does she "debug" or fix her mistake? Teacher will then choose two volunteers to select and model reaching a selected destination on the map. The person navigating the map will use the blind-fold.

Activity: "Find the Building" (20 minutes)

Have students take out their Social Studies workbooks. One partner will be the "programer" and the other partner will be the "Walker", they will alternate roles after each destination. The "Walker" will close his/her eyes and keep them closed until he/she reaches their destination. The "Walker" will start at either Main Street or Third Street. The "Programmer" will guide the "Walker" to the destination of his choice using step-by-step instructions using the compass rose vocabulary of North, South, East, West. If the Walker made a "mistake", the programmer needs to help him/her "debug" and try again.

Wrap-Up, (12 minutes): Journaling: Prompt: What did you like about the activity? What did you find frustrating? Do you think it's important to be able to follow directions? Why or why not?

Palmira Reyes
Second Grade
Sheridan Street Elementary
May 31, 2020

Lesson Name: "Painting Like Jackson"
Code.org/Sphero Topic: Relationship Between Time and Distance
Standards Alignment: CSTA K-12 Computer Science Standards: CS.1 Select and operate computing devices that perform a variety of tasks accurately and quickly based on user needs and preferences.
CSTA K-12 1A -AP-09: Model the way programs store and manipulate data by using numbers or other symbols to represent information.
Visual Arts Standards 2nd Grade: CR2.12a Experiment with various materials and tools to explore personal interest in work of art or design.
Visual Arts Standards 2nd Grade: Cr2.2.2a Demonstrate safe procedures for using and cleaning art tools and equipment.
Speaking & Listening 2.6 Produce complete sentences when appropriate to task or situation in order to provide requested detail or clarification.

Overview: Teacher will read aloud the book "Action Jackson" by Jan Greenburg and Sandra Jordan to the class and discuss lines, composition, design, and artistic technique. Students will take turns using the Sphero Robots to create their own work of art inspired by Jackson Pollock using the Sphero Draw program within the Sphero Edu App.

Objective: Students will be introduced to the Sphero Robot and the Sphero App and will be able to take turns and calibrate the speed and directionality of the Sphero Robot to create a work of art inspired by Jackson Pollock.

Estimated Time: 60 minutes

Materials:

- 23 9x14 white, art paper
- 6 file box/copy paper ream box tops
- Tempra paint in multiple colors; red, yellow, black, blue etc.
- 6 paintbrushes
- 6 cups of water
- 6 kitchen towels
- 6 Sphero Robots
- 6 iPads with Sphero Edu App
- Photograph of Jackson Pollock
- Photographs of Pollock's "Lavender Mist" and "Convergence"
- Book Action Jackson by Jan Greenburg and Sandra Jordan
- 23 Chromebooks

Lesson Guide:

Warm Up (5 min)

Show students the photograph of Jackson Pollock and explain that he was a famous American painter who started a new way of painting. Show students the pictures of two of his works, "Lavender Mist" and "Convergence". Tell students that we will be reading a book about him. Show the book to the class and read the title and authors. Show students the Sphero Robot. Tell them they will be using Sphero Robots just like it to create their own works of arts inspired by Jackson Pollock.

Activity, (15-17 minutes): Teacher will read aloud the book Action Jackson to students, pausing to discuss the artist's technique and use of lines and angles.

Activity, (3 minutes): Watch the video "Painting with Robots: Sphero 2.0 Street Art" (52 seconds) Explain to students that we will be breaking up into groups. 6 students at a time will get to work with the Sphero Robots at the kidney table with the teacher, while the rest of the class works on

Code.org on their Chromebooks and wait for their turn to use a Sphero Robot to create their work of art.

Small group activity, (5-7minutes per group): Teacher gives each student a box top with the art paper in it, a Sphero Robot, a cup of water, and a paintbrush. Students are told that they will be able to choose up to 4 paint colors to create their Jackson Pollock inspired work of art. Teacher models for students how to cover their Sphero Robot with paint using a paint brush. Teacher models how to use the Sphero Edu Draw program to "move" the Sphero Robot using directionality words like North, South and 60 degrees and 90 degrees. Students are asked to try it themselves. Teacher also models how to change speed levels to move the Sphero Robot. Students are asked to follow. Teacher Models how to clean the Sphero Robot and how to clean the paint brush so that students can choose another paint color.

Students are released to work independently using the Sphero Draw App and Sphero Robot to create their artwork. Teacher supervises and assists as needed. Upon completion, students go to their desks to work on Code.org and waiting students take their turn with the Sphero Robots.

Wrap-Up, (8 minutes): Have students share their artwork using the following sentence stems:

"I was inspired by Jackson Pollock to create this work of art. I created _____. I used the colors _____, _____, _____, and _____ because _____."

Clean up, (5 minutes): Clean and put away Spheros, paint brushes, paint. Turn off and put away Chromebooks.

Joseph A. Umana-Walker

Murchison Street Elementary

Binary Bracelets

Create a bracelet that spells your name using the following code that computers understand!

Unplugged Lesson: Binary Bracelets

Lesson Name: Binary Bracelets

Standards Alignments: CSTA K-12 Computer Science Standards (2017). AP - Algorithms Programming-1A-AP-09: Model the way programs store and manipulate data by using numbers or symbols to represent information

Overview: By using binary codes, students will understand how something from real life can translate into a series of on's and off's or zeros and ones.

Learning Objectives: Students will be able to:

1. Encode letters into binary
2. Decode binary back to letters
3. Relate the idea of storing letters on paper to the idea of storing information in a computer.

Estimated Time: 50 minutes

Materials:

Notebook for annotation

Binary Bracelets –Worksheets

Sentence strips

Lesson Guide:

Warm-Up: Teacher writes on the whiteboard a message using binary codes:

How Computes Read Programs

Teacher asks, "What do you notice?"

Students respond, "Shapes, white squares, black squares".

Teacher shows students pictures of the inside of a computer. This creates a lot of dialogue. Students notice chips, wires, cables, fans. Teacher explains that wires carry information through the machine in the form of electricity. Then, a computer uses electrical information like the classroom lights turned on or off. This is called "Binary." Computers store information using information in this binary format. So today we are going to learn to write our name using binary, just like a computer.

Activity :

Provide students with a Binary Bracelets –Worksheet, sentence strips

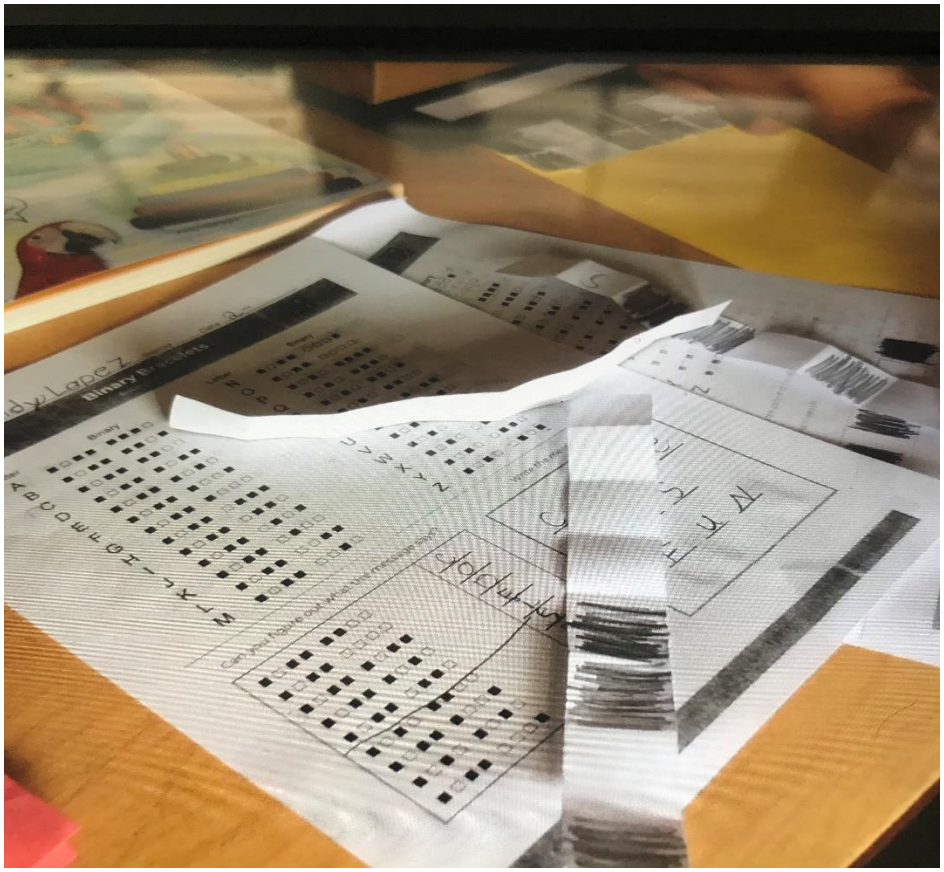
Directions:

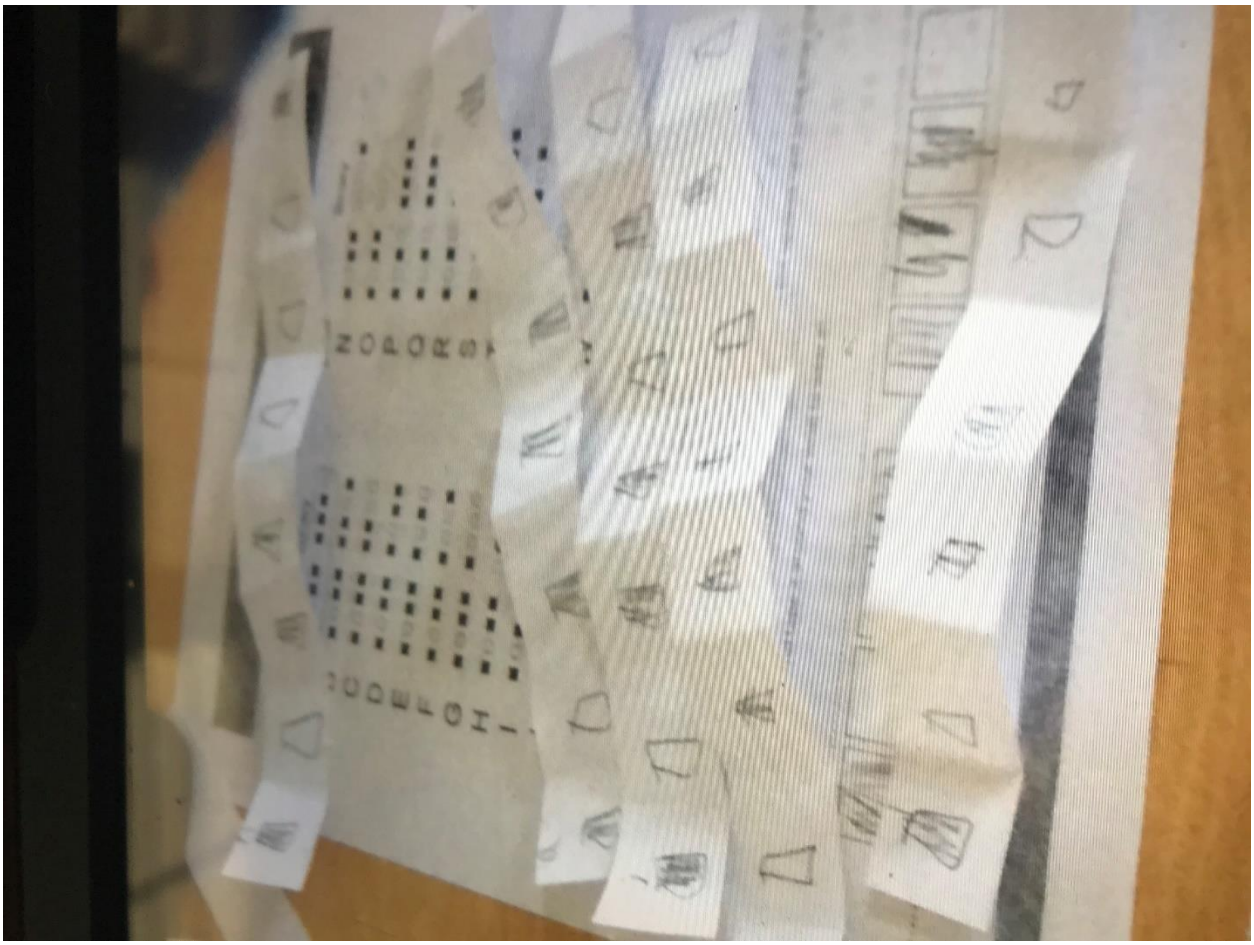
Find the first letter of your first name on the activity sheet. Fill in the squares of a bracelet to match the pattern of the squares next to the letter that you selected. Provide feedback as all students do their first letter of their name. Next, students continue with their second letter, and so on. Teacher continues providing feedback to all students. Last, teacher tapes the bracelets around their wrists for students to wear them.

After the activity of the binary bracelet, revisit the message that was on the board and see if the students can decipher it:

Wrap Up: What did we learn today?

Journaling: Write about what you learned, why is it useful?





Joseph A. Umana-Walker

Murchison Street Elementary

Plugged Lesson: Sphero

Lesson Name: Visiting Continents with Sphero

Standards Alignments: CSTA K-12 Computer Science Standards (2017). AP - Algorithms Programming-1A-AP-09: Model the way programs store and manipulate data by using numbers or symbols to represent information.

Students use map and globe skills to determine the absolute locations of places and interpret information available through a map's or globe's legend, scale, and symbolic representations.

Overview: Students will be given a maze with the names of the seven continents. Every time the Sphero SPRK stops at a continent, it will speak the name of the continent and play a sound.

Learning Objectives: Students will be able to:

1. Program the Sphero SPRK by working cooperatively
2. Recite the names of the seven continents
3. Learn the continents locations

Estimated Time: 50 minutes

Materials:

Paper tiles

Vocabulary words

Notebook for annotation

Sphero(s)

Sphero Edu app on devices

Lesson Guide

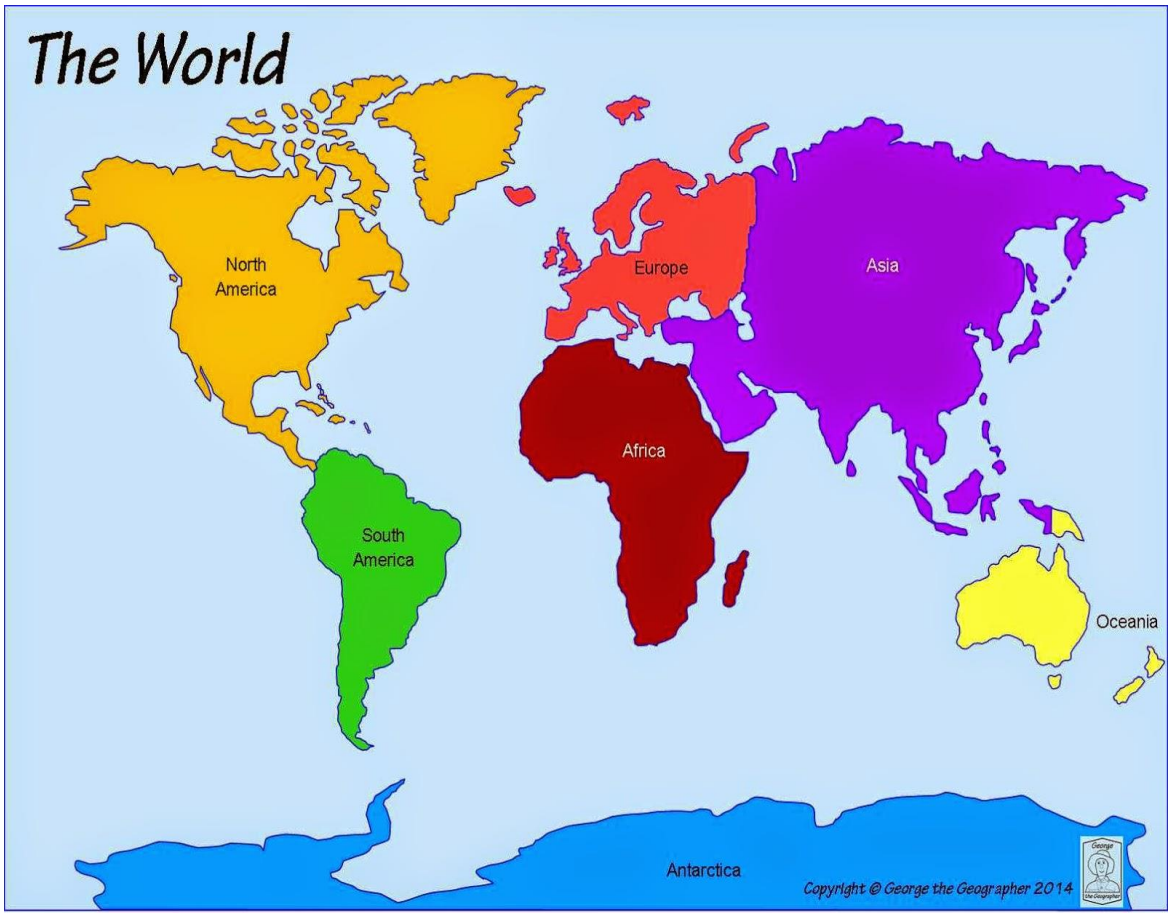
1. **Warm-Up:**
   Watch two videos about the Sphero SPRK.
   https://video.search.yahoo.com/yhs/search?fr=yhs-mnet-001&hsimp=yhs-001&hspart=mnet&p=how+to+program+sphero+to+roll#id=35&vid=93782f43cf26b407e0744add48dbb185&action=view

https://video.search.yahoo.com/yhs/search?fr=yhs-mnet-001&hsimp=yhs-001&hspart=mnet&p=how+to+program+sphero+to+roll#id=57&vid=6f2ae026b170a714886602e6fb7b6d84&action=view

Activity:

Students program their Sphero to roll through all the continents. All groups create a maze with the names of the seven continents with the paper tiles: **North America, South America, Africa, Europe, Asia, Antarctica** and **Australia.**

North America		Europe	Asia	
				Australia
South America		Africa	Antarctica	



Programming Steps:

Step 1: Each group works collectively to begin their program. Teacher guides them with the first step.

Teacher says, "Everyone clicks on the word **Movement** (the light blue menu at the bottom of the screen). Next, select **"roll ____ at ____speed for _____".**

Teacher asks, "Which direction should Sphero roll?"

Students reply, "It should roll 0 degrees at 30 speed for 1 second."

Teacher responds, "Very good". "Now, let's go ahead and enter the information". (teacher waits for students to ask what to do next)

One student says, "We need to add 'delay' before we continue to the next continent".

Since students are already familiar with this type of coding, they know that they need to add the "delay" operation from the Control menu, so the Sphero can pause.

From this moment on, students are on their own coding their program. Teacher is only monitoring and offering feedback.

Wrap Up: What did we learn today?

Journaling: Write about what you learned, why is it useful?