

Computational Thinking Boosters: Algorithms in K-2

Date: Jan. 19, 2021

[Link to recording](#)



This work was supported by the National Science Foundation under grant award #1923314. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

Introductions



Agenda

What: 20min CT Boosters(K-2)

Plan for today:

- 5 minutes: **Introduction & Vocabulary**
- 15 minutes: **Lesson Ideas**
- 5 minutes: **Q & A**



Guiding Questions:

- Why do we need directions?
- Have you ever tried to follow directions that are not clear? What happened?
- What kinds of directions do you think computers need?



I can statements:

- I can follow instructions to complete a task.
- I can create a sequence of instructions for people to follow (an algorithm).
- I understand that the order of instructions is important.



Technology & Computer Science in KY



7 Big Ideas of Technology

- Global Collaborator
- Computational Thinker
- Creative Communicator
- Empowered Learner
- Digital Citizen
- Knowledge Constructor
- Innovative Designer

5 Key Concepts of CS

- Networks & the Internet
- Using Algorithms & Programming
- Data Analysis
- Computing Systems
- Impacts of Computing

Technology Standards for Primary (K-2)



Concept: Computational Thinker (CT)

Competency: Students understand sequences and use them to develop solutions to problems.

Standard:

CT1. Develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test solutions.

Learning Priority:

C. Break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.

Indicator(s) for grades K-2:

1. Break a problem into smaller parts, identify key information, and use age-appropriate digital tools to help with problem solving
ex.: online whiteboard, online mindmapping tools, digital outline , with guidance and support.

Learning Priority:

D. Understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.

Indicator(s) for grades K-2:

1. Define and give examples of automation ex.: thermostat controls temperature, buttons pressed on toys make various sounds . 2. Complete a simple coding task with at least 3-5 coded actions ex.: html, block-based coding, python , with guidance and support.

Vocabulary: Algorithms

Algorithms are precise step-by-step plans or procedures to meet an end goal or to solve a problem; algorithmic thinking is the skill involved in developing an algorithm.

- Shuchi Grover in [Computer Science in K-12: An A-To-Z Handbook on Teaching Programming](#)

An algorithm must be seen to be believed, and the best way to learn what an algorithm is all about is to try it.

-Donald Knuth in [The Art of Computer Programming, Volume 1](#)

Examples:

- Maps and Directions
- Cooking/Baking
- [Sorting Algorithms ← Computer Science Unplugged](#)



Algorithms

Definition: A process or set of rules to be followed in calculations or other problem solving operations especially by a computer.

Characteristics:

- **Finite:** They must always end after a finite number of steps. Every algorithm needs an instruction that tells it to stop.
- **Has Inputs and Outputs:** an algorithm can have inputs and produces an output (or a result).
- **Effective:** An algorithm is generally expected to produce an answer (or a result).
- **Definite:** Each step must be precisely defined. Actions are carried out in a rigorous and unambiguous way. The instructions need to be so clear that they do not need to be simplified.

PBS Kids Scratch JR

An App for iPads or Android tablets where students can build their own interactive stories or games.

[ScratchJR Example Activities](#)

[Introductory Video](#)



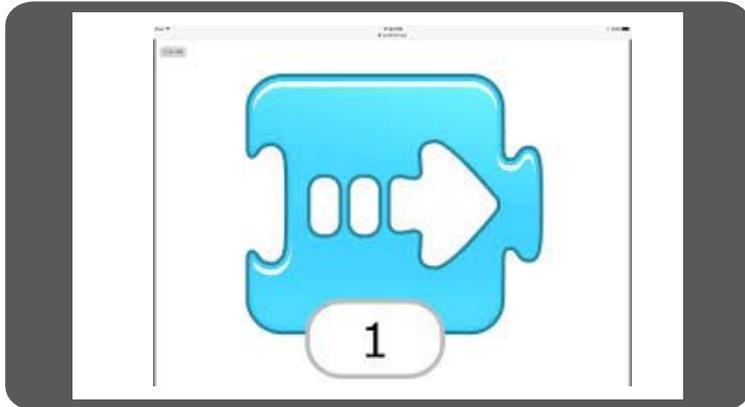
Unplugged Coding

What: [Printable Block Images](#)

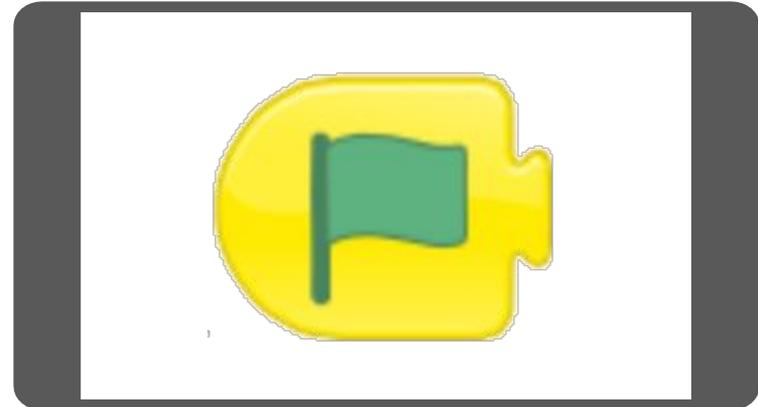
Why: Young Children may benefit from manipulating the blocks as an introduction to online coding.

Use the motion and trigger blocks to let students create dance routines, maps, or directions around the classroom by writing their own code through the use of these printable blocks. This would be a great introduction to coding while using physical movement.

Motion Blocks



Trigger Blocks



Music Algorithm

What: Music Lessons with Percussion Instruments

Why: Students learn the importance of completing a sequence in order. There is a specific process to achieve the end result.



 Twinkle Twinkle Little Star


Twin-kle, twin-kle litt- le star.


How I won- der what you


are. Up a-bove the world


so high. Like a dia- mond in


the sky. Twink- le, twink- le


litt- le star. How I won- der what

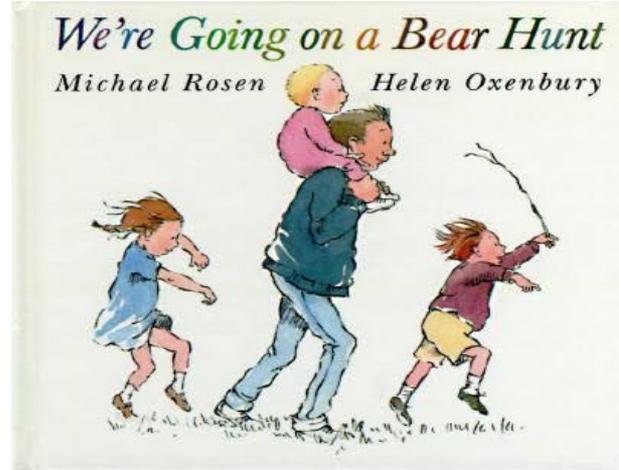

you are.

Going on a Bear Hunt

Put a story in the correct order by using the worksheet enclosed in the online lesson.

Good Stories:

- 3 Pigs
- 3 Bears
- If You Give a Mouse a Cookie
- The Napping House
- There Was an Old Woman who Swallowed a Fly



[YouTube Retelling](#)

Sequencing a Task

Ideas:

- Getting dressed
- Making a pizza
- School procedures (Hallway, restroom, lining up)
- Drawing a picture
- Brushing your teeth



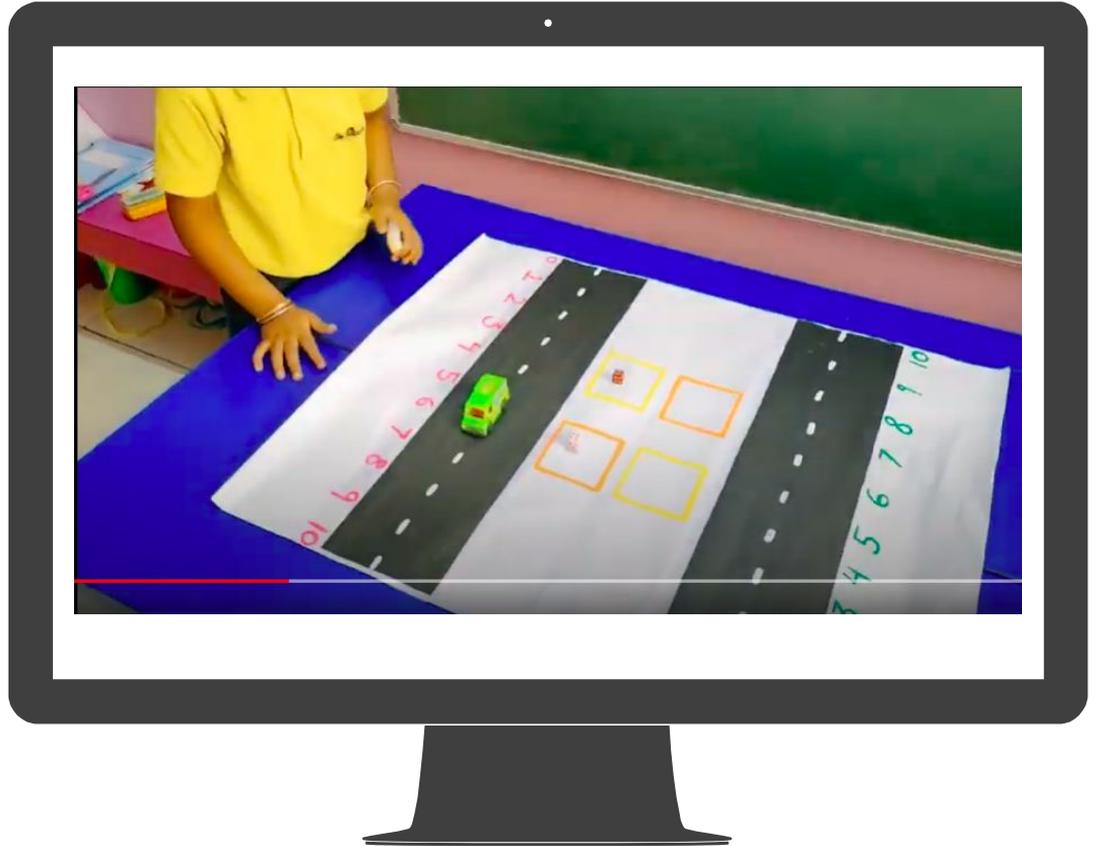
Code and Go Mouse

- Allow students to create a “Noun Town” for the mouse to travel through. This can reinforce map skills as well.
- Recreate a story by allowing students to draw different portions of the story. Then code the mouse to travel around the map to order the story correctly.



Use a Number Line

- Create a number line with a road above.
- Create two boxes above the road.
- Allow the students to roll two dice and place them in the two boxes.
- Make a small card with + or - to place between the two boxes.
- Allow the students to add or subtract the numbers and move the toy car along the road.



Code a Dance

- Allow students to identify a health concern that exercise could help improve. (Problem based learning)
- Create a class dance that could provide such exercise.
- Allow students to create their own dance for the class or a partner to test.

Other teaching ideas:

- Brain Breaks
- Virtual Learning
(Teachers Pay Teachers)



Where Am I Likely Already Integrating These Skills?

- Many skills for K-2 students are algorithmic in nature.
- Thought processes - such as decomposition, pattern recognition, drawing a conclusion, comparing and contrasting
- Following a daily schedule (picture or word)
- Student/teacher collaborate to create classroom procedures through testing and analyzing outcomes
- Conversations about how order of sequence affects outcomes
- Others?

