

@CSTANJ

@CS4NJ

#CSPDWeek2026

#CompSciNJ

@csteachersorg



New Jersey CSPDWeek

This material is based upon work supported by the National Science Foundation under Award No. 2607763.



Meet the Presenters



Alicia Somers

Classroom Teacher, Prek-2
Spotswood Public Schools

Dedicated to empowering
early learners opportunities
and exposure to CS and
access to inclusive CS tools.



Jahaira Ortiz

Classroom Teacher, K
Union City Public School

Enthusiastic, & passionate STEM &
Coding Facilitator dedicated in
developing creative experiences, &
facilitating project-based learning
activities for young learners.



Joann Case

Technology & PLTW Teacher, K-4
North Hanover TWP Public School

Dedicated to empowering the
next generation of innovators and
problem solvers through coding,
robotics, and physical computing.



Unplugged Computer Science Plan

Educators will:

- **Understand** key early CS concepts (sequencing, patterns, logic, problem-solving)
- **Experience** unplugged activities as learners
- **Practice** facilitating activities for young children
- **Connect** activities to early learning standards (play-based, social-emotional, literacy)



What is Unplugged Coding?

Hands-On CS

Unplugged coding is a way to teach computer science concepts **without using computers or digital devices**.

Students learn core programming mindsets through physical activities, games, movement, puzzles, and real-world collaborative problem-solving.



Algorithms

Creating step-by-step instructions



Sequencing

Putting instructions in the correct order



Loops

Repeating a set of instructions



Conditionals

Making decisions based on a condition



Debugging

Finding and fixing errors

Classroom Games

Mapping physical play and classroom activities to core computer science concepts.



Simon Says

Algorithms and conditionals



Red Light, Green Light

Events



Follow-the-Leader

Sequencing



Scavenger Hunt

Search algorithms



Guess My Number

Binary search



Telephone Game

Data transmission



Escape Room Puzzles

Computational thinking



Unplugged Learning

Building intuitive mental models
before introducing complex
coding syntax.

★ Follow the Leader — Visualizing the Game

A classic movement game mapping physical play to sequences and instructions.

Leader Demonstrates Action

One person acts as the leader, performing sequences of physical actions. The rest of the group must observe and copy them exactly.

↺ Sequencing Actions

- Walk 🚶
- Jump 🦘
- Clap 🖐️
- Spin ↺
- March 🥁



Simon Says — Visualizing the Game

A classic movement game mapping physical play to algorithms and conditional instructions.

The Leader Giving Commands

One player acts as the leader (Simon), issuing directions. Others must only move when the conditional trigger is present.

Valid Commands (Execute)

- "Simon says touch your nose!"
- "Simon says jump!"

Trick Commands (Ignore)

- "Clap your hands!" — Fails conditional check (No "Simon says")



Red Light, Green Light — Visualizing the Game

A classic movement game mapping physical play to event triggers and conditional actions.

The Leader & Event Triggers

One player acts as the leader, standing at the front and facing away from the group to issue event signals.

Green Light Event (Execute)

- “Green light!” — Players are allowed to move forward.

Red Light Event (Stop)

- “Red light!” — Players must freeze instantly.
- **Leader turns around** to catch any moving players!



*This material is based upon work supported by
the National Science Foundation under Award No.
2607763.*

Grade 3 Unplugged CS Activities



CONCEPT



FUN ACTIVITY



OBJECTIVES



TIME



Algorithms

Human Robot – Give step-by-step directions to a "robot."

Computers need clear, ordered instructions.

**30
min**



Debugging

Fix the Algorithm – Find and correct mistakes in directions.

Errors are called bugs, and fixing them is debugging.

**30
min**

Grade 3 Unplugged CS Activities



CONCEPT



FUN ACTIVITY



OBJECTIVES



TIME

 **Binary**

Binary Bracelets – Use black and white beads to make secret codes.

Computers use only 0s and 1s.

40 min

 **Data Rep.**

Pixel Art Mystery – Recreate pictures using color coordinates.

Pictures are made of tiny pixels.

35 min

 **Sorting**

Human Number Sort – Work together to order numbers.

Computers sort information quickly and efficiently.

30 min



Unplugged - Code a Friend

Voice Helper Pretend Play

Voice Helper Pretend Play

Goal: Connect to real-life AI (like voice assistants)



The Smart Helper Role

- Kids pretend to be "smart helpers" (like Alexa/Siri).



The User Role

- Others ask questions: "What's the weather?" "Play music!"



Learning Outcome

- Encourage polite speaking, and listening skills and following directions.



This material is based upon work supported by the National Science Foundation under Award No. 2607763.



Unplugged - I am a Robot

Unplugged Activity — Understanding how instructions guide technology



"I Am a Robot" Game

Goal: Understand how AI follows instructions



Give Directions

- One child is the "robot," others give simple directions (walk, clap, turn).



Step-by-Step

- Teacher emphasizes: "Robots/AI follow instructions step-by-step!"



Silly & Fun

- Make it silly—wrong directions = funny results.



This material is based upon work supported by the National Science Foundation under Award No. 2607763.

★ Obstacle Course Coding



With permission from an adult, or your teacher, create the most epic obstacle course you can with things around the house/class. Take a video of yourself or someone else completing the course and share it with your teacher.

Everyday Classroom Materials

You can build many unplugged CS lessons using these inexpensive, highly reusable items.



Index cards



Sticky notes



Grid paper



Dice



Playing cards



Plastic cups



String or yarn



LEGO bricks or building blocks



Pattern blocks



Colored paper



Clothespins



Dominoes



Counters or buttons



Whiteboards & dry-erase markers

This material is based upon work supported by the National Science Foundation under Award No. 2607763.

★ Everyday Classroom Materials Examples



This material is based upon work supported by the National Science Foundation under Award No. 2607763.

☀️ Everyday Classroom Materials Examples



This material is based upon work supported by the National Science Foundation under Award No. 2607763.

★ Unplugged CS Materials Chart-1

Quick reference for mapping common, low-cost classroom items to computational concepts and activities.

 Material	 Computer Science Concept	 Example Activity
 Index Cards	Algorithms • Sequencing	Write step-by-step instructions or coding commands.
 LEGO Bricks / Blocks	Algorithms • Engineering • Debugging	Build a model by following a partner's algorithm.
 Sticky Notes	Variables • Sorting • Data	Create flowcharts, organize information, or build simple programs.

★ Unplugged CS Materials Chart-1

Quick reference for mapping common, low-cost classroom items to computational concepts and activities.

 Material	 Computer Science Concept	 Example Activity
 Pattern Blocks	Patterns • Computational Thinking	Create repeating patterns and identify rules.
 Grid Paper	Pixel Art • Coordinates • Data	Design pixel pictures or map coding paths.
 Colored Paper	Binary • Pixel Art • Representation	Build binary messages or colorful pixel mosaics.
 Whiteboards & Markers	Algorithms • Debugging • Flowcharts	Write, test, erase, and improve algorithms together.

This material is based upon work supported by the National Science Foundation under Award No. 2607763.

★ Unplugged CS Materials Chart-2

Quick reference for mapping common, low-cost classroom items to computational concepts and activities (Continued).

 Material	 Computer Science Concept	 Example Activity
 Dice	Randomness • Probability • Algorithms	Roll dice to make coding decisions or create random programs.
 Clothespins	Binary • Counting	Clip on/off to represent 1s and 0s or build binary numbers.
 Playing Cards	Sorting • Searching • Classification	Practice sorting algorithms or search challenges.
 String or Yarn	Networks • Paths • Graphs	Build classroom networks or create maze pathways.

This material is based upon work supported by the National Science Foundation under Award No. 2607763.

Unplugged CS Materials Chart-2

Quick reference for mapping common, low-cost classroom items to computational concepts and activities (Continued).

 Material	 Computer Science Concept	 Example Activity
 Dominoes	Patterns • Matching • Logic	Match patterns, sequences, or create algorithm paths.
 Plastic Cups	Stacking Algorithms • Engineering	Follow directions to build towers and debug mistakes.
 Counters or Buttons	Counting • Data • Binary	Represent data, create graphs, or model binary numbers.

This material is based upon work supported by the National Science Foundation under Award No. 2607763.

The Mats

Maintenance & Setup

Bee Bots mats work best when **stored upright**.

If it has a bubble, the bee bot will have trouble moving correctly on the mat surface.

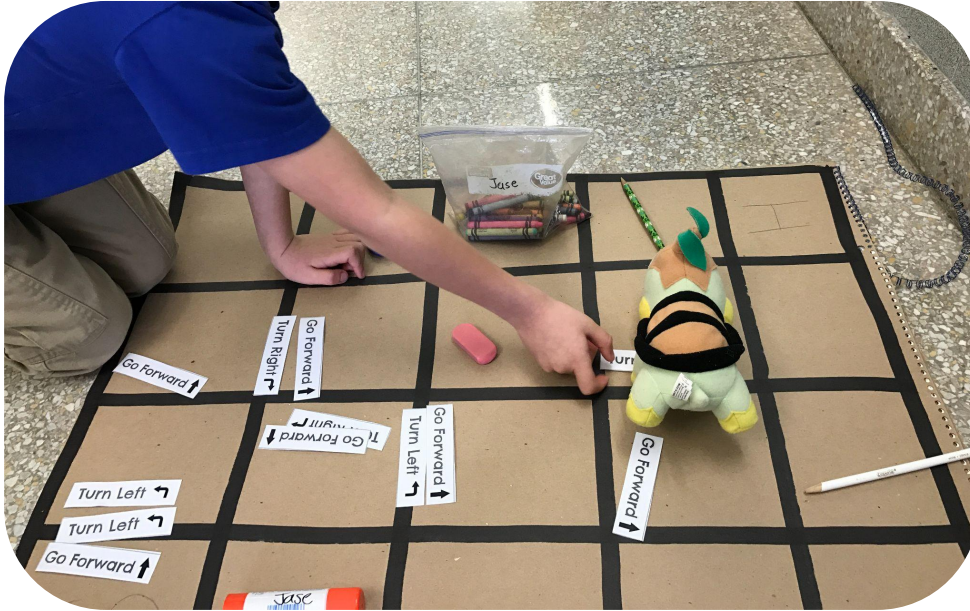
DIY Solutions:

- Use a shower curtain (tape it down for stability).
- Create a grid on poster board and laminate it.





More Unplugged Coding Ideas



Classroom Connection:

- Giving and Following directions
- Story retelling

Teacher Activity:

Students code their stuffy through a maze. Label one block Start and the other block end. Then place everyday classroom materials (eraser, glue stick, crayons, pencil, etc.) on the grid and have the students program their stuffed animal around them.



Create Your Own Coding Mats



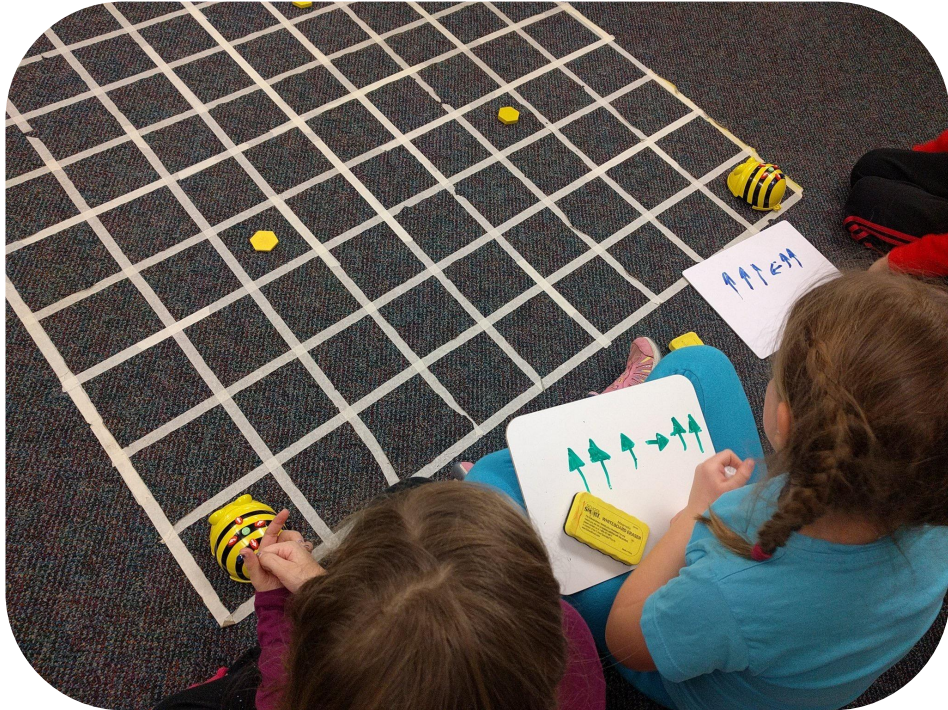
Classroom Connection

- Giving and Following directions
- Story retelling

Teacher Activity

Using a tarp and masking tape, create a versatile grid mat for your students. Label one block "Start/Home" and the another block "End." Add pictures to the grid and have students take turns coding their friends through the maze. Pictures could represent places in a neighborhood, content area vocabulary words, math problems, letters, sight words, etc.

Create Your Own Coding Mats



Classroom Connection

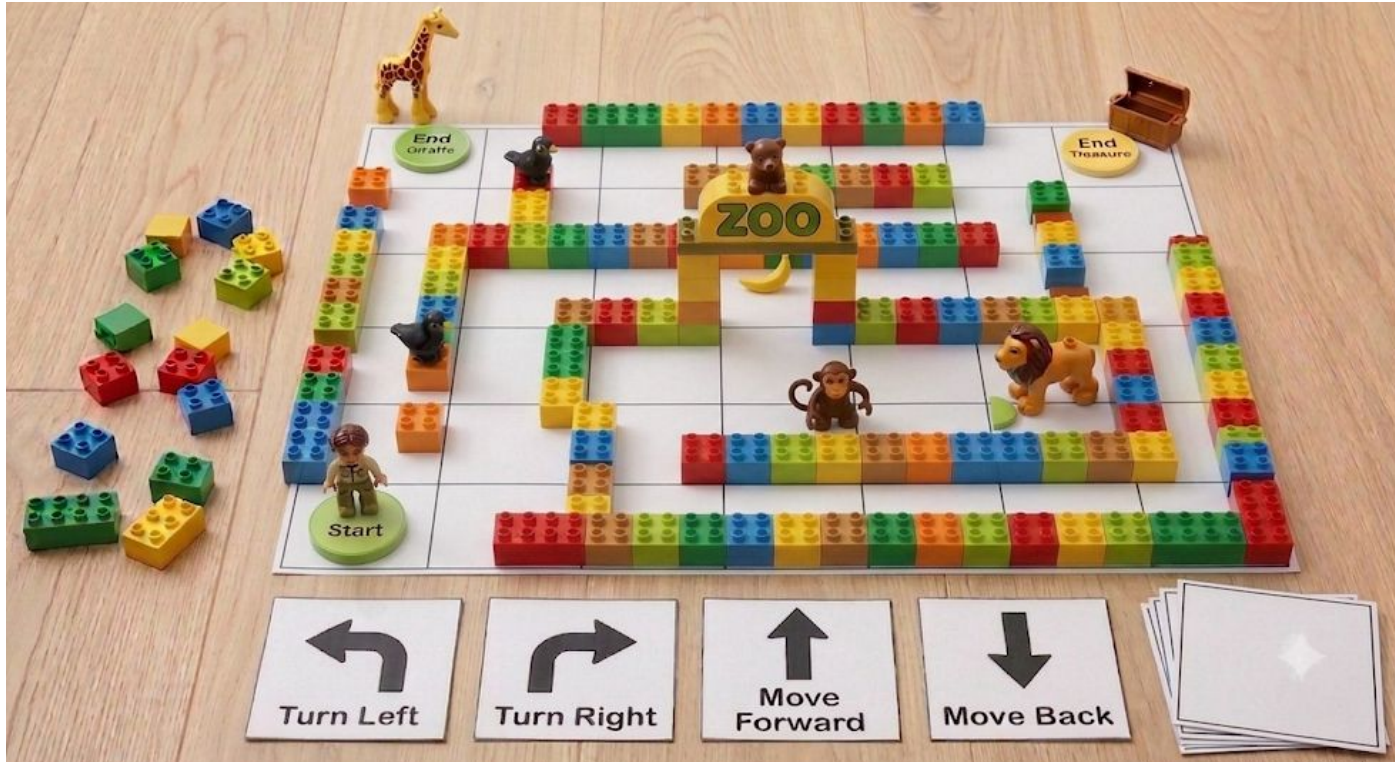
- Giving and Following directions
- Story retelling

Teacher Activity

Using a masking tape, create a grid on the floor for your students. Place objects on the grid and have students code their Bee-Bots or other object/robot through the maze.

Other ideas: add content area vocabulary words, math problems, letters, sight words, etc.

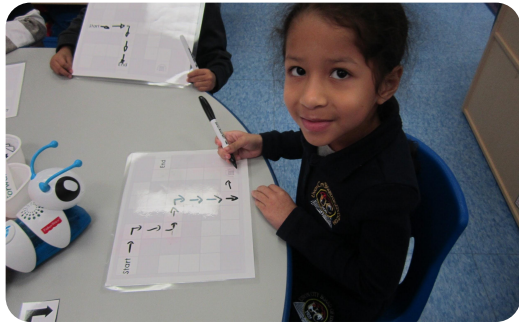
Coding a Lego Maze



This material is based upon work supported by the National Science Foundation under Award No. 2607763.



Create Your Own Coding Board Games



Classroom Connection

- Giving and Following directions
- Story retelling

Teacher Activity

Using poster board, file folders, or cardstock, students can draw their own tabletop grid games.

Label a "Start" and "End" space. Children can draw obstacles or themed pictures on spaces, and create custom direction cards (forward, turn left/right) to navigate a small toy or game piece through their custom board game.

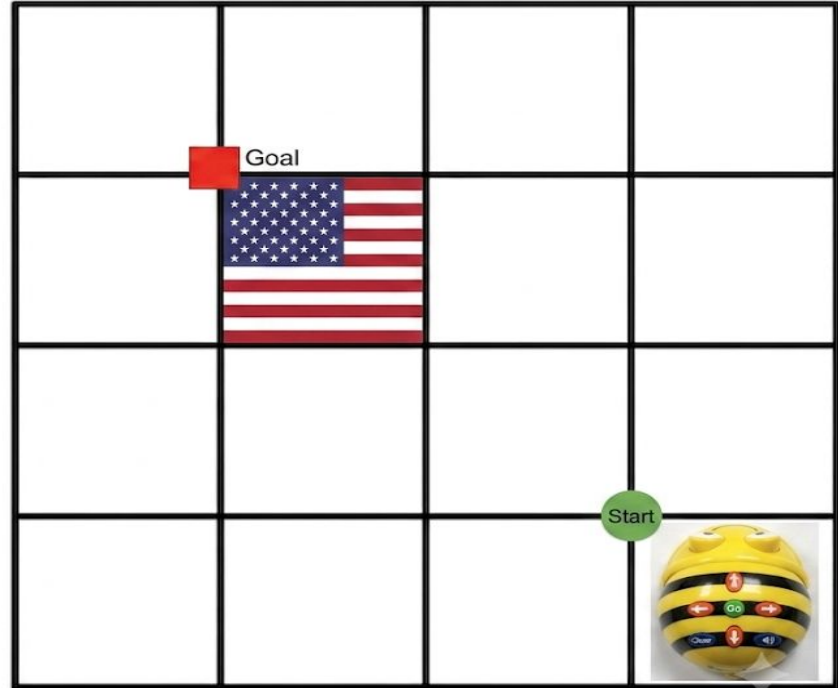


Create Your Own Paper Board Games

Get your
robot from
one point to
another

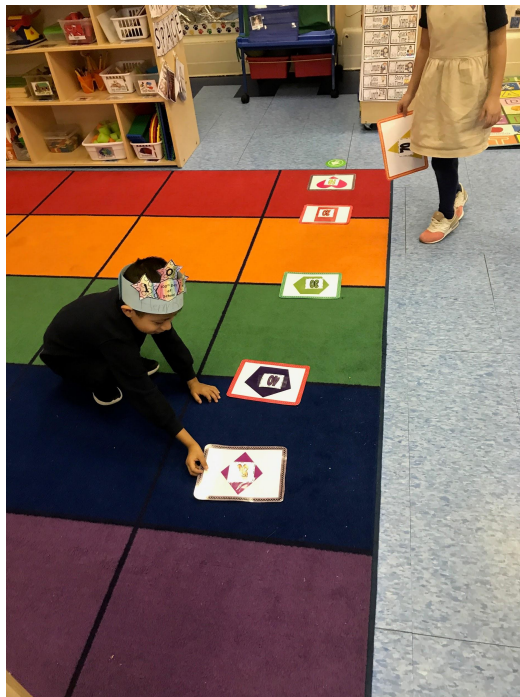
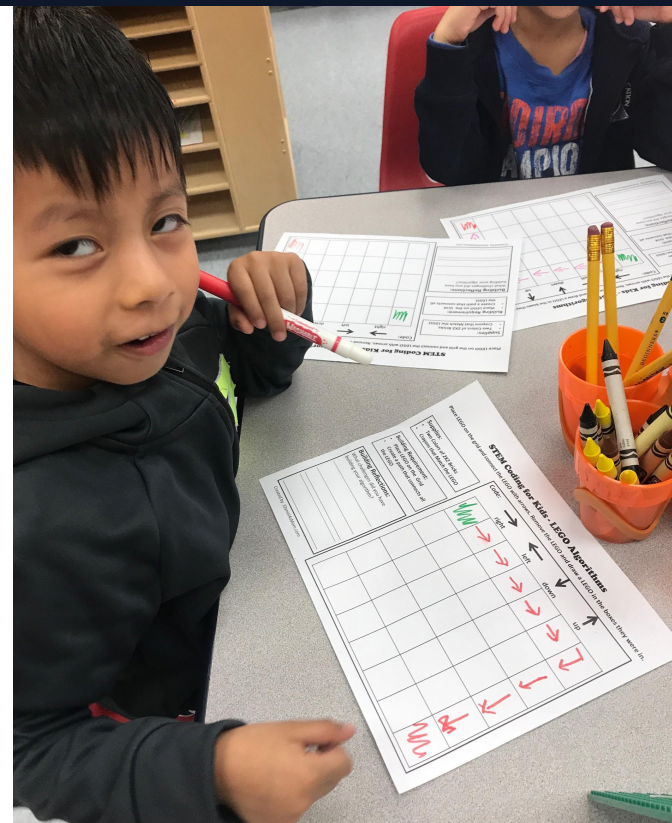


1 your path:
Draw program arrows here:





More Unplugged Activities



This material is based upon work supported by the National Science Foundation under Award No. 2607763.



Make Your Own Mats



This material is based upon work supported by the National Science Foundation under Award No. 2607763.



Logic & Problem Solving



Classroom Connection:

- Centers
- Cooperative play

Teacher Activity:

- Solve simple mazes
- "If/Then" movement games



Debugging (Finding and Fixing Mistakes)



Classroom Connection:

- Growth mindset
- Problem-solving discussions



Teacher Activity:

- Identify what went wrong in a sequence
- Fix incorrect instructions

Unplugged Computer Science Games

Active, social, screen-free computational thinking through play for PreK-2

Robot Freeze Dance (PreK-2)



Objective: Follow step-by-step instructions.

How to Play:

- One player is the “robot.”
- Others give commands (e.g., step forward, turn left, jump).
- Robot moves or freezes until new command.

Builds:  Algorithms  Listening

Human Board Game (PreK-2)



Objective: Understand sequences and rules.

How to Play:

- Create a board on the floor with spaces.
- Players take turns rolling a die and moving.
- Follow the rules to win!

Builds:  Sequencing  Rules

Pattern Path (K-2)



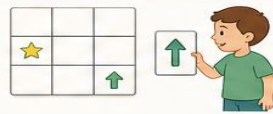
Objective: Create and extend patterns.

How to Play:

- Use tiles or cards to make a pattern.
- Partner continues the pattern or creates a new one.

Builds:  Patterns  Abstraction

Code the Classroom (1-2)



Objective: Write and follow algorithms.

How to Play:

- Choose a start and goal.
- Write arrow commands (→ ↑ ↓ ←) to reach the goal.
- Partner follows the code.

Builds:  Algorithms  Debugging

Obstacle Course Challenge (PreK-2)



Objective: Plan steps to solve a problem.

How to Play:

- Set up an obstacle course.
- Plan the path from start to finish.
- Test it out and improve it!

Builds:  Decomposition  Problem Solving

Sorting Showdown (PreK-1)



Objective: Sort and classify data.

How to Play:

- Spread out different objects.
- Sort by color, shape, size, or another attribute.
- Count and compare groups.

Builds:  Data & Analysis

Secret Code Messages (K-2)



Objective: Encode and decode information.

How to Play:

- Create a code using pictures, numbers, or symbols.
- Write a secret message.
- Partner decodes it!

Builds:  Encoding  Abstraction

Simon Says: Computer Edition (PreK-2)



Objective: Practice listening and following commands.

How to Play:

- Teacher says, “Simon says” jump” or “Simon says turn around.”
- Follow only if “Simon says” comes first!

Builds:  Algorithms  Self-Control

Build It! (K-2)



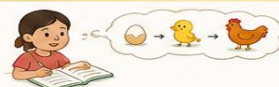
Objective: Follow directions and build.

How to Play:

- One player has a model.
- They give step-by-step directions.
- Partner builds the same structure.

Builds:  Decomposition  Algorithms

What Comes Next? Story Challenge (1-2)



Objective: Sequence events in a story.

How to Play:

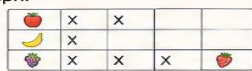
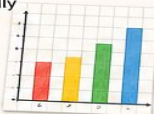
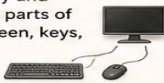
- Read or act out a story.
- Put the events in order.
- Predict what might happen next!

Builds:  Sequencing  Prediction

This material is based upon work supported by the National Science Foundation under Award No. 2607763.

Unplugged Computer Science Games

Active, social, screen-free computational thinking through play for PreK-2

CS Concept (Standards)	Preschool (3–4 years)	Kindergarten (5–6 years)	1st Grade (6–7 years)	2nd Grade (7–8 years)
Algorithms & Programming (CA 1A, 1B, 1C) 	Follow the Path Students follow a partner's directions to move around the room. 	Build a Sequence Students arrange picture cards to put familiar events in order. 	Paper Airplane Algorithm Students write step-by-step instructions for how to make a paper airplane. 	Obstacle Course Design Students create step-by-step instructions (an algorithm) for a partner to navigate an obstacle course. 
Data & Analysis (CA 2A, 2B, 2C) 	Classroom Tally Students tally their favorite color, snack, or animal and compare results. 	Sort It! Students sort objects by one or two attributes (color, shape, size). 	Picture Graphs Students collect data (e.g., favorite fruit) and create a picture graph. 	Tally & Bar Graphs Students collect data, tally results, and create a bar graph to represent it. 
Decomposition (CA 1D) 	Break It Up Students break simple tasks (e.g., getting dressed) into smaller steps with picture cards. 	Story Sequencing Students break a story into beginning, middle, and end. 	Task Breakdown Students break down a classroom task (e.g., making a peanut butter sandwich) into smaller steps. 	Project Planning Students break a larger project into smaller tasks and put them in order. 
Patterns & Abstraction (CA 3A, 3B) 	Finish the Pattern Students identify and extend simple patterns using colors, shapes, or movements. 	What Comes Next? Students continue patterns using objects, colors, or shapes. 	Create & Extend Patterns Students create repeating patterns and ask classmates to continue them. 	Find the Rule Students identify the rule in a pattern and use it to extend or create their own patterns. 
Computing Systems (CA 4A, 4B) 	What Is a Computer? Students explore different devices and identify that they help us do things. 	Parts of a Computer Students identify and match the basic parts of a computer (screen, keys, mouse). 	Human Computer Students act out how people (users) give instructions and computers (helpful classmate) follow them. 	All Around Computers Students find examples of computers in our daily lives and explain how they help us. 

Standards referenced from CSTA K–12 Computer Science Standards (2017). Check your state for specific grade-level expectations and code references.

This material is based upon work supported by the National Science Foundation under Award No. 2607763.



Unplugged Computer Science Games

Active, social, screen-free computational thinking through play for 2nd - 3rd Grade



FILE FOLDER GAME

Unplugged Coding Activity



The programmer gives the computer directions to recreate and draw an image that is **hidden** in the open file folder!

PROGRAMMER

Looks at the picture inside the folder and gives clear directions.

Draw 4 up.
Draw 4 right.
Draw 4 down.
Draw 4 left.
Draw a roof.
Add a door, window and chimney.

COMPUTER

I will follow the directions step by step!

Listens carefully and follows the directions to draw the image.

HOW IT WORKS:

- 1 The programmer looks at the picture inside the folder.
- 2 The programmer gives clear, step-by-step directions.
- 3 The computer listens carefully and follows the directions.
- 4 The computer draws the image on their grid.
- 5 Reveal and compare! Did the computer recreate the image?

EXAMPLE:

1. Hidden Picture (Programmer sees this)



2. Programmer Gives Directions (Example)

↑ Draw 4 up.
→ Draw 4 right.
↓ Draw 4 down.
← Draw 4 left.
Draw a roof.
Add a door, window and chimney.

3. Computer's Drawing (Result)



4. Compare
Do they match?

=
YES!

5. Great Job!

You did it!



WHY IT'S FUN & HELPFUL:

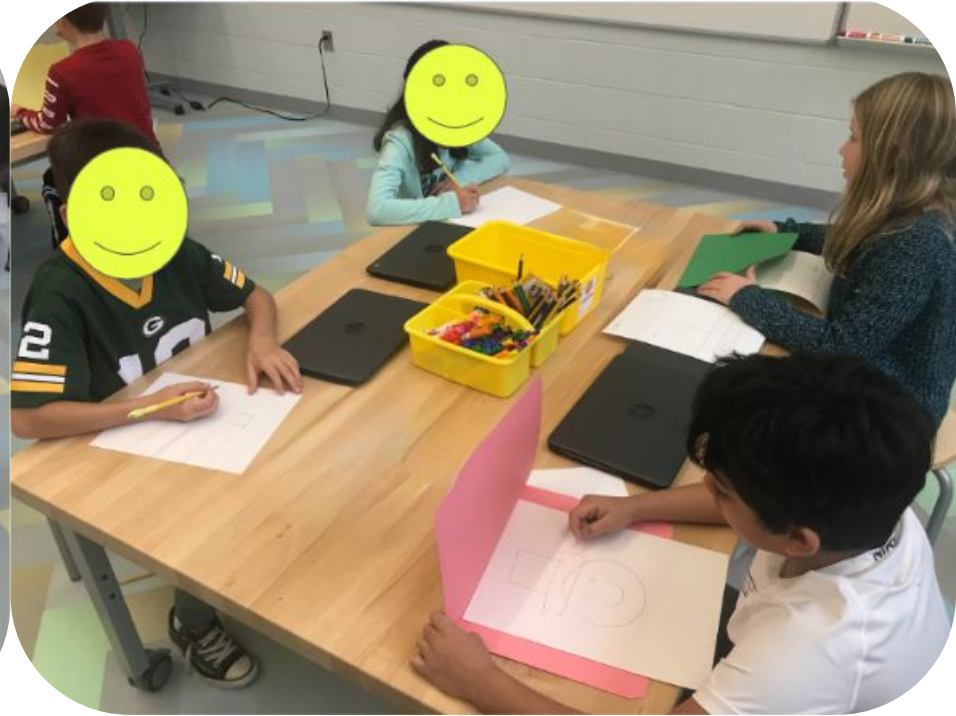
- ✓ Builds listening and communication skills
- ✓ Practices direction following
- ✓ Introduces coding concepts
- ✓ Encourages patience and problem solving



This material is based upon work supported by the National Science Foundation under Award No. 2607763.

Unplugged Computer Science Games

Active, social, screen-free computational thinking through play for 2nd - 3rd Grade



This material is based upon work supported by the National Science Foundation under Award No. 2607763.



Awesome Algorithms



An algorithm is a set of instructions that are done step by step. Coders tell the computer what to do.

You do this every day such as washing hands, brushing teeth or making cookies. You follow things step by step.

Objective

- Students will understand that an algorithm is a sequence of steps and practice creating and following one.

Materials

- Bread (real or pretend)
- Peanut butter & jelly (or paper cutouts)
- Plastic knife
- Chart paper or whiteboard
- Picture cards of steps (optional)



Algorithm - Example Unplugged

Peanut Butter & Jelly Recipe

Ingredients:
2 pieces of bread
peanut butter
jelly
1 knife
(and an adult's help)

1		Open the jars
2		Spread peanut butter on bread
3		Spread jelly on bread
4		Put pieces of bread together
5		Cut in half and eat!

© 2010 National Science Foundation

Get a slice of bread 	Get a knife
Cut the bread in half 	Get peanut butter
Spread peanut butter on one piece of bread 	Get jelly
Spread jelly on the other piece of bread 	Put one slice of bread on top of the other

Name: _____

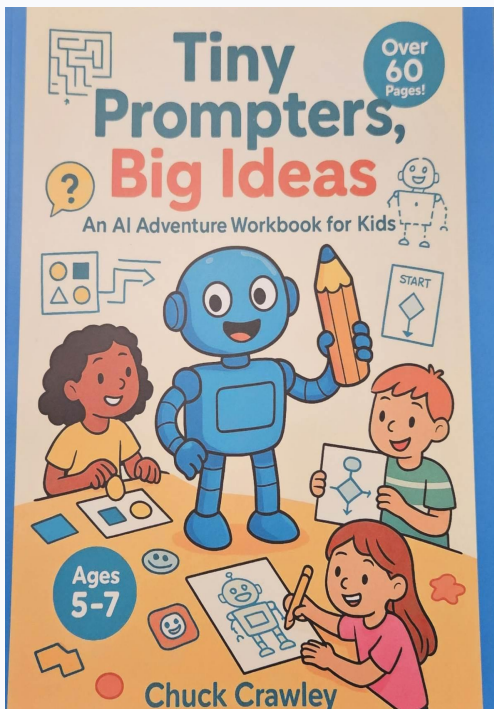
Making a peanut butter and jelly sandwich.

1		Get a plate.
2		Get loaf of bread.
3		Take out 2 slices of bread and set them on the plate.
4		Get jar of peanut butter.
5		Get jar of jelly.
6		Spread peanut butter on one slice of bread.
7		Spread jelly on one slice of bread.
8		Put slices together with dry sides facing out and set on the plate.
9		Eat your sandwich. Enjoy!



Tiny Prompters, Big Ideas!

An AI Adventure Workbook for Kids (Ages 5-7)



Hi there!

You're holding something special—a book that turns kids into creative thinkers, clear communicators, and tiny prompt engineers!

This book helps you **talk to computers, robots, and smart helpers** the way they understand best: with **AI**.



What Kids Will

Do:



Sort shapes like a robot



Give step-by-step directions



Explore how computers learn



Build their own AI helper



Table of Contents

Tiny Prompts, Big Ideas!

Welcome

- Welcome 1
- Personalize This Book 2
- Table of Contents 5

Unit 1: Smart Helper

- Design Your Avatar! 6
- Hello, Smart Helper! 7
- Pattern Play 8
- Robot Simon Says 9
- Draw-What-I-Say 10
- Design Your AI Helper 11



Table of Contents

Tiny Prompts, Big Ideas!

Unit 2: Prompt Power!

- Prompt Dice 14
- Spot the Pattern 15
- Feelings & Fairness 16

Unit 3: Talk Like a Bot

- Choose-Your-Own Intro 19
- Arrow Flow Fun 20
- Kindness Bot Prompter 21
- Sticker Tracker 22
- Robot Museum Poster 23

Unit 4: Debug & Reflect

- Bug Buster! 25
- Draw the Shape 26
- Feelin' Meter 27
- Prompt Puzzle 28
- Draw Yourself as Prompter 29
- Trace Another Prompt! 30

Coloring & Puzzles

- Prompt Garden 33
- Robot Playground 36
- Word Search 39



Unit-by-Unit Teaching Tips

Tiny Prompts, Big Ideas!

UNIT 1

What Is AI?



Learning Goal Learn about AI helpers and how they work.



Teaching Prompts What tasks can a helper do? How do they use patterns?



Pro Tip: Kids pair up & design an AI helper together.

UNIT 2

Prompt Power



Learning Goal Experiment with & refine different prompt types.



Teaching Prompts Which words are most helpful? What changed with edits?



Pro Tip: Offer an extra chance to play with dice & prompts.



Unit-by-Unit Teaching Tips

Tiny Prompts, Big Ideas!

UNIT 3

Talk Like a Bot

 **Learning Goal** Explore coding concepts like algorithms.

 **Teaching Prompts** Discuss how bots speak and follow basic instructions.

 **Pro Tip:** Let kids practice speaking 'like a bot' in pairs.

UNIT 4

Debug & Reflect

 **Learning Goal** Identify problems and brainstorm solutions.

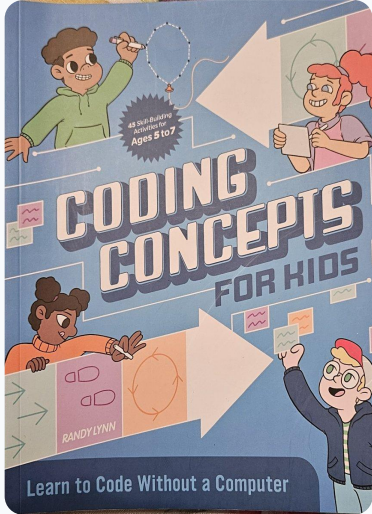
 **Teaching Prompts** Ask about times they fixed something or had to reflect.

 **Pro Tip:** Treat debugging as a fun, collaborative puzzle!



Recommended Books

Tiny Prompts, Big Ideas!

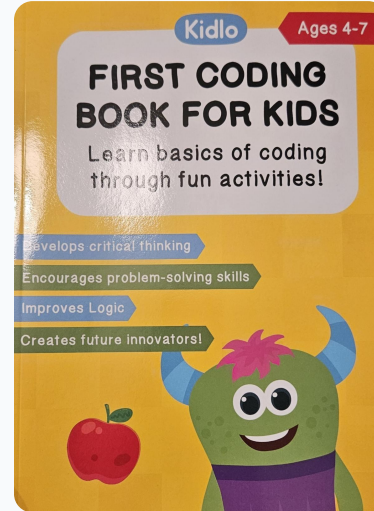


Coding Concepts for Kids

Learn to Code Without a Computer

Highlights:

- Ages 5 to 7
- 45 Skill-Building Activities
- By Randy Lynn



First Coding Book

Learn basics of coding through fun activities!

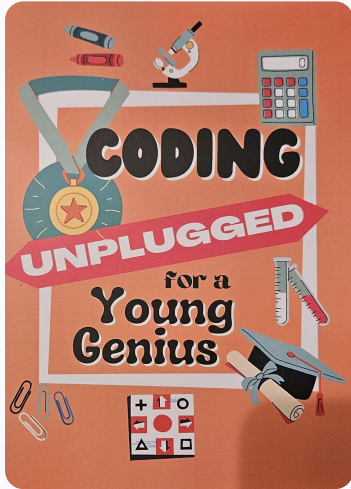
Highlights:

- Ages 4 to 7
- Kidlo Series
- Develops critical thinking & logic



Recommended Books

Tiny Prompters, Big Ideas!



Coding Unplugged

For a Young Genius

Highlights:

- Early learners guide
- Offline activities & puzzles
- Foundational logic



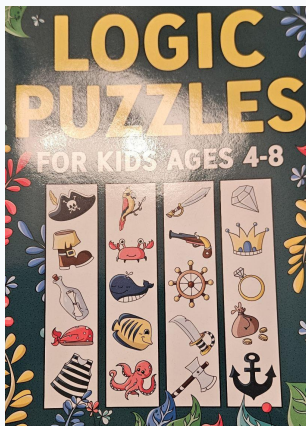
Coding for Kids

Unplugged Algorithms Workbook

Highlights:

- For Kids Ages 5-7
- Screen-free learning
- Critical thinking
- By Kate Kallgren

Logic Puzzles Ages 4-8 | By Activity Wizo

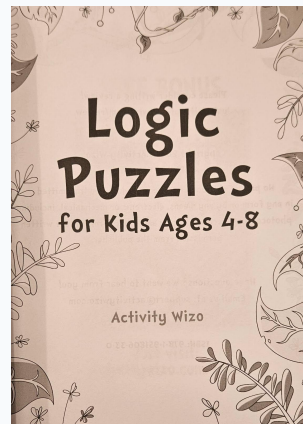


Logic Puzzles Cover

Interactive
brain-building
activities for kids

Features:

- Age-appropriate challenges
- Colorful, fun characters
- Builds early confidence



Inside Preview

A look inside the
introductory activities

Highlights:

- Clear, simple instructions
- Develops spatial reasoning
- Perfect for classroom use



Smart Start: Beginning Coding

Pre-K Stories & Activities

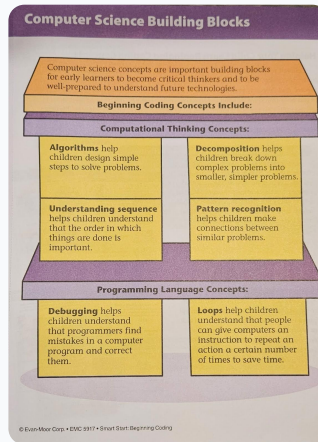


Workbook Cover

Evan-Moor Smart Start

Highlights:

- Critical thinking
- Computational logic
- Stickers & audio
- Basic coding skills



Inside Preview

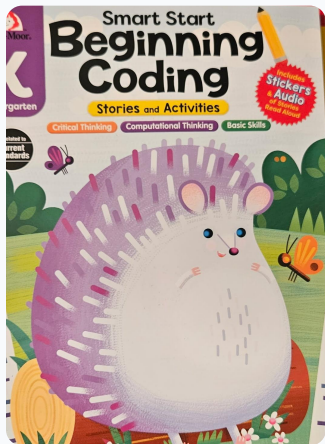
Building Blocks

Key Concepts:

- Algorithms & sequence
- Pattern recognition
- Decomposition
- Debugging & loops

Smart Start: Beginning Coding

Kindergarten & Grade 1 Stories & Activities



Kindergarten Stories & Activities

Highlights:

- Critical thinking
- Computational thinking
- Stickers & audio
- Basic coding skills



Grade 1 Stories & Activities

Highlights:

- Critical thinking
- Computational thinking
- Stickers & audio
- Basic coding skills

Summary & Implementation

Differentiation Tips

Preschool:

- Focus on play, movement, and visuals
- Use simple 2–3 step directions

Kindergarten–1st:

- Add more steps and complexity
- Introduce problem-solving challenge

Teacher Takeaways

By the end of training, teachers will:

- Feel confident teaching CS without technology
- Have ready-to-use activities
- Understand how CS supports early learning skills

CSTA Standards: PreK–3rd

Explore the New Guidelines

The Computer Science Teachers Association (CSTA) has updated its standards for 2026. Access the official interactive portal to browse, search, and implement these new computer science education benchmarks in your PreK-12 curriculum.

Grade-Level Standards

 [PreK / Kindergarten](#)

 [Grade 1 Standards](#)

 [Grade 2 Standards](#)

Including Grade 3+12 You can find those additional standards at the bottom of this [webpage](#).

CSTA- New 2026 Standards

Explore the New Guidelines

The Computer Science Teachers Association (CSTA) has updated its standards for 2026. Access the official interactive portal to browse, search, and implement these new computer science education benchmarks in your PreK-12 curriculum.



Official CSTA Portal

2026 CSTA PK-12
Standards

Additional Resources

- CS Unplugged
- CodeAI
- Computer Science Unplugged - National 4-H Council
- Unplugged Programming Activities
- Kodable: Unplugged Coding Activities
- Unplugged Coding Worksheets
- For Free CS Materials -
 - Join the Civil Air Patrol and become an Aerospace Education Member (AEM)



Finding Playful Coding Foundations

Curated Resources for PreK-3

Explore playful unplugged coding activities from trusted early childhood education sources.

Resources

- [Preschool STEAM](#)
- [Playful Coding Lessons for PreK-3](#)

Thank You!



Thank you for your time and participation. We appreciate your dedication to advancing Computer Science and Unplugged Activities in the classroom.

For further resources and collaborations, please reach out to us and visit [CSTANJ](#) to attend chapter meeting that are held every month during the school year. They offer a great deal of resources all year long!

You must complete the survey each day for your stipend!



**Please complete the survey after
each session and complete one at the
end of the day for your STIPEND.**

**Thank you for choosing CS in your
classrooms!**

@CSTANJ

@CS4NJ

#CSPDWeek2026

#CompSciNJ

@csteachersorg



New Jersey CSPDWeek

This material is based upon work supported by the National Science Foundation under Award No. 2607763.