



CoderZ at Home Curriculum

CoderZ is designed to feel like a fun game, blending maximum engagement with a sense of accomplishment! Each course emphasizes a complementary set of hard and soft skills, with real-life applications so all kids can relate.

Codyyssey World with LEGO® Education SPIKE™Prime

The road to code starts here! Codyyssey World introduces young learners to coding and robotics while developing core STEM skills.

Ages 8-11

Courses: 10

Level: Novice

Missions: 113

Hours: 30+

Adventure Peak

- Drive a robot using code
- Practice problem solving, ordering, precision
- Explain how things work

Frozen Island

- Plan and document a program
- Use mathematical, data-based models
- Learn why planning is important

The Lost City

- Use repeat loops, write readable code
- Use measurement, rulers, patterns
- Discover how patterns make problems easier to solve

Crystal Crater

- Use parameters and arguments
- Use angle measures, parallelism, types of angles
- Practice persistence and resilience

Candy Town

- Drive a robot in 'smooth' turns
- Use circles, arcs, radius, diameter and circumference
- Compare different methods for solving the same task

Sketch It

- Draw letters and pictures by programming a robot
- Practice engineering design
- Break down big problems so they're easier to solve

The Milky Way

- Improve code efficiency
- Understand criteria, constraints for design and testing
- Consider different needs in solving a problem

Weebo's Manor

- Use a distance sensor
- Understand greater than and less than
- Solve a problem in a changing or unpredictable environment

Cyber World

- Program a robot to change its environment
- Use data gathering, hypothesis testing
- Learn how to incorporate data into a solution

Arctic Lab

- Use coordinates to instruct a robot
- Use coordinate planes
- Understand how different technologies work together

Codabunga World

Set in a virtual-farm environment, this beginner-level curriculum for coding, STEM, and computer science builds a strong interdisciplinary foundation that's as fun as it is functional!

Ages: 9-12

Courses: 7

Level: Beginner

Missions: 87

Hours: 40+

Code Control

- Use arguments to change the way a program runs
- Use negative and positive numbers
- Differentiate between hardware and software



Moove It!

- Use a structured debugging process; sequencing
- Understand relationships between speed, time, distance
- Practice persistence and resilience



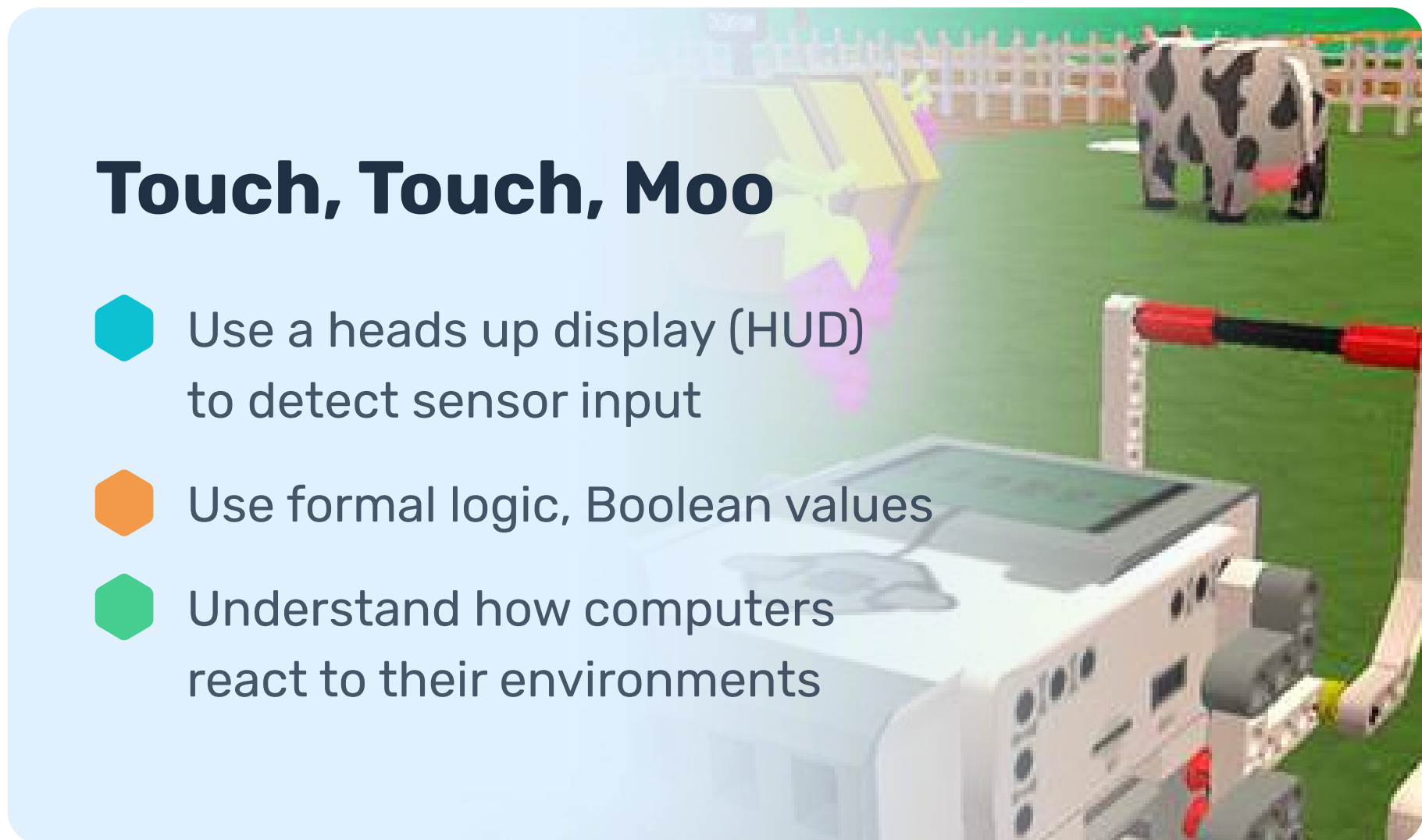
Spinning Around

- Use a structured process to plan a program
- Understand rotational motion versus linear motion
- Break down big problems so they're easier to solve



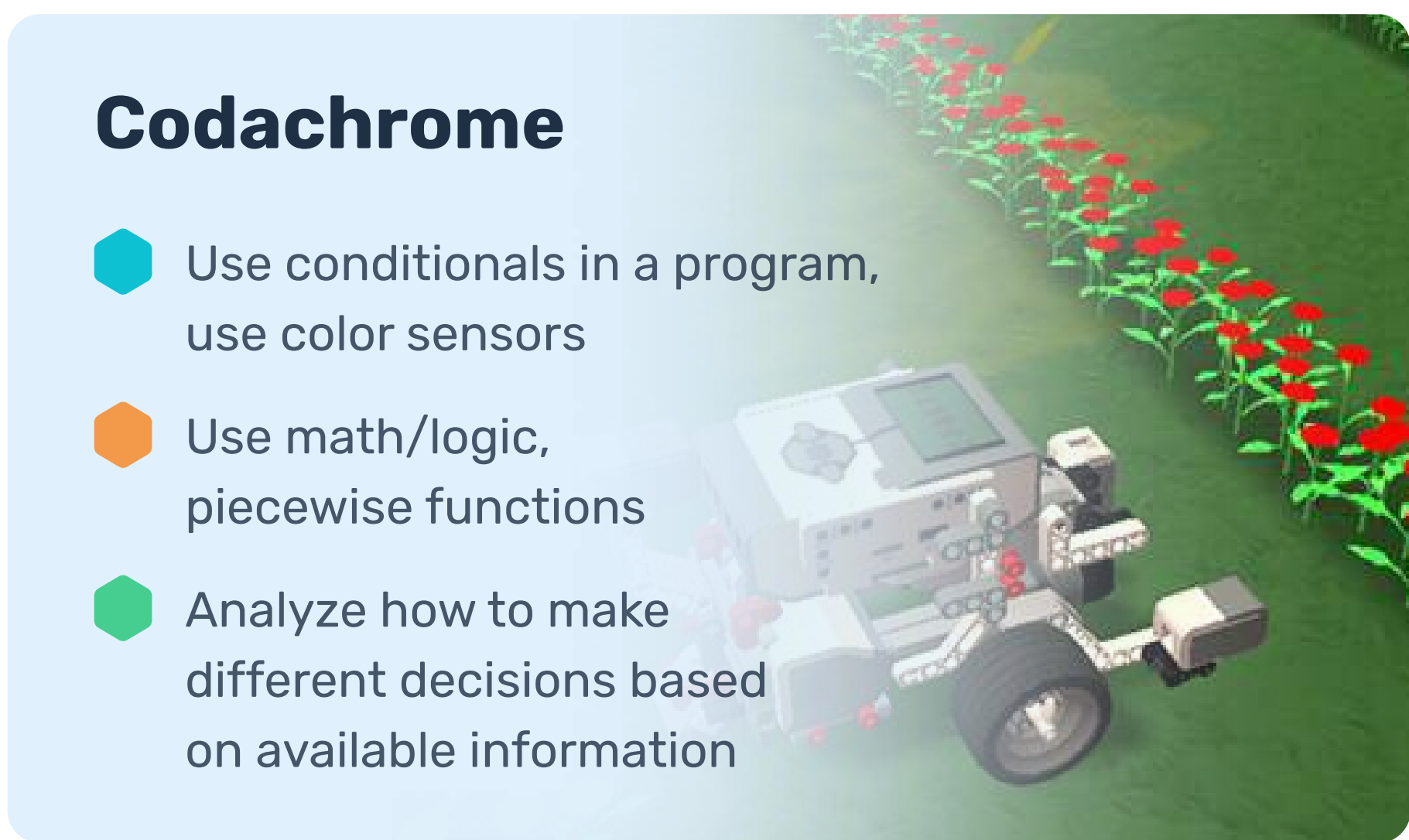
Touch, Touch, Moo

- Use a heads up display (HUD) to detect sensor input
- Use formal logic, Boolean values
- Understand how computers react to their environments



Codachrome

- Use conditionals in a program, use color sensors
- Use math/logic, piecewise functions
- Analyze how to make different decisions based on available information



Gyro's Turn

- Use the gyroscope to gather information about a robot
- Use comparison operators, relationships between angles
- Learn how to incorporate data into a solution



Loop de Loop

- Use a repeat loop to make code more manageable
- Understand patterns, criteria for design solutions
- Generate/choose between multiple solutions to a problem



Robo Rover World

Robots are the future, and this next-level coding and robotics curriculum covers the skills needed for full and active participation!

 Ages: 11-15




 Courses: 15

 Level: Intermediate



 Missions: 102

 Hours: 30+




Ahead of the Curve

-  Use parameters to change the specifics of a command
-  Understand relationships between speed, time, distance
-  Understand how computers react to their environments




STEM Is a Snap

-  Learn how input, output, and code are used in robotics
-  Explain how things work




Circtacular

-  Drive the robot in a curve
-  Use arcs, radius and circumference
-  Break down big problems so they're easier to solve




Sense It

-  Describe and use a touch sensor
-  Use Boolean values; Current and voltage
-  Learn how robots interact with the world through sensors




Repeat After Me

-  Use a repeat loop to make code more manageable
-  Practice identifying patterns
-  Discover how patterns make problems easier to solve




Heads Up!

-  Use heads up display (HUD) to detect sensor input
-  Use rotational speed, momentum
-  Consider how physical limitations impact software



Gyro Returns

-  Use the gyro reset to make it easier to program turns
-  Practice polygons and angle measures
-  Compare different methods for solving the same task




Domino Creations

-  Use a planning process to design and create
-  Use design engineering
-  Learn why planning is important




Mission Unleashed

-  Integrate all previous learnings in this first of three challenge missions. Use a formal problem-solving process: collect data, create a plan, then put it to the test!
-  Use a formal problem-solving process



Supersonic

-  Use an ultrasonic sensor to detect and avoid obstacles
-  Use inequality operators, angle measures, distance
-  Combine multiple strategies to solve complex problems




Color Me Smart

-  Program the robot to color changes in its environment
-  Understand light, color, intensity
-  Understand how different technologies work together




Mission Rising

-  Integrate all previous learnings in this second of three challenge missions. Use navigation, loops, and sensors to program your robot at a basic level!
-  Consider the appropriate tool(s) for solving a problem



A Call to Arms

-  Control the robotic arm using code
-  Use angle measures, force, motion
-  Discover how robots can change their environments

What If?!

-  Set conditions based on data from sensors
-  Use math/logic, piecewise functions
-  Learn how to help robots show more complex behavior

Mission Master

-  In this final challenge mission, you'll combine all the new concepts learned in Robo Rover World to program your robot at an intermediate level
-  Practice persistence and resilience

 Coding

 STEM

 Soft Skills