Celebrate the Hour of Code with //CODE Games Unplugged

Robot Repair: Can You Fix the Robot Brain?

Lesson Overview and Preparation

Overview
Unplugged lessons are a great way to introduce computer science topics. The //CODE Programming Game Series teaches the logic and problem solving skills coders need. For this lesson, students will use a condensed version of the //CODE Robot Repair game to learn about Boolean Logic.

We'll take you through the process of fixing a Robot's brain, step by step! Along the way you'll solve challenges based on the famous 'Boolean Satisfiability Problem,' and within an hour you'll be a Boolean Master and have an understanding of advanced programming concepts! No computers required – everything you'll need can be printed out from our supplied PDF.

Before playing, spend a short amount of time preparing students by introducing a few computer science vocabulary terms. Students will then read the instructions and play the Robot Repair: Can You Fix the Robot Brain? challenge. At the end of the lesson, students will reflect and you will guide a wrap-up discussion.

Time:
60 - 90 minutes

Materials:
- Printed copy/copies of the Robot Repair: Can You Fix the Robot Brain?

Suggested Grade Levels
This lesson is appropriate for grades 3 and up. The early challenges will be very easy for secondary students and the later challenges will be very difficult for elementary students. No prior Math skills are needed for playing. English language learners may need support understanding the instructions especially when jargon is used (e.g. node, power cell, sensor…). The challenges will expand students' logical thinking and problem solving abilities.

Managing Game Play:
Ideally, students should work in groups of 2-3. Each group should have their own game to play. Feel free to structure the student game play time however works best for your classroom. We suggest using one of these options:

Option 1: Provide each group of students with a printed copy of the Robot Repair: Can You Fix the Robot Brain? and have everyone play simultaneously.

Option 2: Print a single copy of the Robot Repair: Can You Fix the Robot Brain? and provide it as a small-group center-based activity. After completing the lesson introduction, have students rotate through the game center while the rest of the class works in other centers or on another project.

Lesson Plan

Topic

Today, we have a very special lesson. We will be learning about computer science by playing a game!

Objective

Students will be able to explain and use Boolean logic to solve challenges.

CSTA National Computer Science Standards

Grades 3-5

AP-11 Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.

AP-15 Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.

Grades 6-8

AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

Grades 9-12

AP-17 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.

Vocabulary

- **Boolean Logic**: problems that evaluate to True or False
- **Boolean Expression**: a statement that is either True or False
- **Logic Gate**: an operation on inputs that are True or False

Agenda

- Vocabulary - 5 minutes
- Stand If… - 10 minutes
- Play //CODE game - 35+ minutes
- Reflect on what we learned - 10 minutes

Stand If...

A big part of coding and computer science is called **Boolean Logic**. Boolean Logic includes problems that evaluate to True or False. In Boolean Logic, we use Boolean expressions and logic gates.

A **Boolean Expression** is a statement that is either True or False. For example, "The lights are on" is a Boolean expression, it's either true or false (ask the students if it is true or false, then turn the lights off and ask again). Ask the students if the following statements are Boolean expressions:

- "The lights are NOT off." *(yes, it is, because it evaluates to True or False)*
- "Your dog's name." *(no, it is not, because it does not evaluate to True or False)*
- "How old are you?" *(no, it is not, because it evaluates to your age, not True or False)*

Let's play the Stand If game to demonstrate **Boolean Logic**. To play, we will take turns telling the class to "Stand up if you…" and add a Boolean expression. Then, anyone that the Boolean expression is true for will stand up. For example, you could say, "Stand up if... **you are wearing red**." Anyone wearing read would then stand up.

Have the students play for a few rounds and then tell them that you want to add a twist. You are going to add a logic gate, or an operation on inputs that are True or False. In other words, we are going to add operations to our Boolean Expressions. In algebra, we use + - * / and other operators to manipulate numbers. In Boolean Logic we use AND OR XOR NOT NAND NOR and IFF to manipulate Boolean Expressions. If it sounds complicated, don't worry, we will start simple.

Let's start with the AND operator by asking two students to suggest different Boolean expressions. Students will only stand if BOTH Boolean expressions are true for them. For example: "Stand up if you are **left handed** AND **if you like to run**" would result in only students who are left-handed and like to run standing up.

After play for a few rounds, tell the students that you are going to change the rules again. This time, two students will say Stand Up phrases but this time they will stand if one OR the other is true about them. This is an OR logic gate. For example: "Stand up if you are wearing shoes or you like kittens" would result in all students who are wearing shoes or who like kittens to stand up.

Extension: with older students you could keep complicating the game by using two Boolean expressions with other logic gates found in the game. Here are the logic gates and how they could be used in the Stand if game:

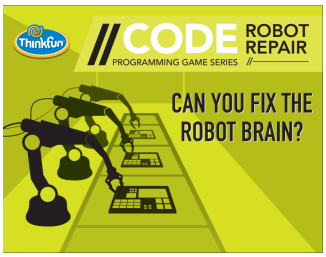
Logic Gate	Stand if...
AND	All Boolean Expressions are TRUE
OR	At least one Boolean Expression is TRUE
IFF	Either both Boolean Expressions are TRUE or both Boolean Expressions are FALSE
XOR	One Boolean Expression is True and one Boolean Expression is False
NOR	All of the Boolean Expressions are False
NAND	At least one of the Boolean Expressions is False

Next, students will play the Robot Repair: Can You Fix the Robot Brain? Challenge where they will be using Boolean Logic just like they did in the Stand Up activity.

Play Robot Repair: Can You Fix the Robot Brain? Challenge

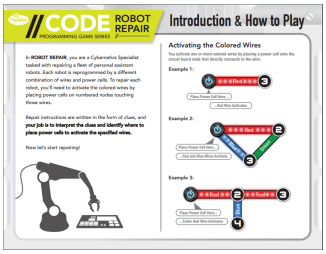
The Robot Repair: Can You Fix the Robot Brain? challenge is non-competitive and collaborative. Players work together to solve challenges.

Before letting students jump in on their own, explain how the rules, demonstrate how to play, then complete a challenge as a class. You can do this by using the Robot Repair: Can You Fix the Robot Brain? PDF file as a guide. Display the PDF on a projector and work through pages 1-6 using this guide:



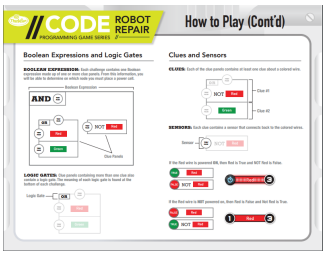
Page 1: Title Page

Today we are going to play a robot game using the Boolean Logic we just used in our Stand If game.

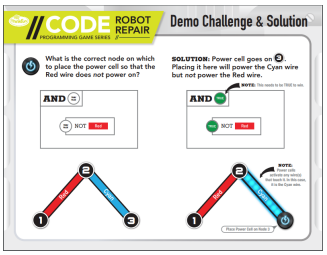


Page 2: Introduction & How to Play

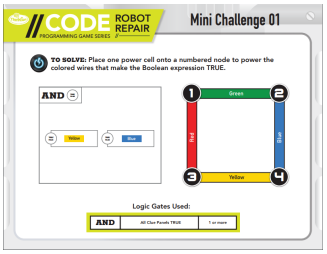
Have students read along, or ask students to take turns reading aloud, the introduction and instructions on this page. Point out in the examples which wires are powered and which ones are not.



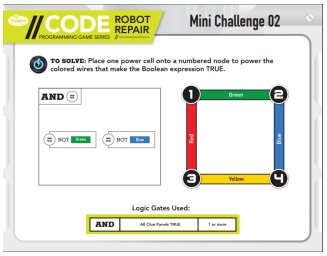
Page 3: How to Play (Cont'd)
Have students read along, or ask students to take turns reading aloud, the instructions on Boolean expressions, logic gates, clue, and sensors. Make the connection that you just learned about Boolean expressions and logic gates in your previous Stand If activity. Also compare the clues and sensors to your Stand If activity (the clues are like the "you are wearing read" statements and the sensors are like the students position: standing/sitting).



Page 4: Demo Challenge & Solution
Walk through the demo challenge with your students, demonstrating your thought process as you look at the Boolean expressions, logic gate, and clues. Read aloud the text on the page as you work through the solution.



Page 5: Mini Challenge 01
Demonstrate how you would solve this challenge. If you have a document camera, use it to demonstrate placing the power cells and sensors. Think aloud as you solve the challenge. When you complete the challenge, show how you use the sensors to make sure you have the correct answer. (Answer: power cell placed on node 4).



Page 6: Mini Challenge 02
Work with your students to complete this challenge together. Ask a different student to help you with each step as you complete the challenge. Ask the students to explain their thoughts and reasons behind their suggestion. Again, if you have a document camera, use it to demonstrate placing the game pieces. After completing the puzzle, demonstrate how to use the sensors to check if your answer is correct (Answer: power cell placed on node 3).

Page 15 contains the answers to each challenge. The answer indicates which node should be covered by a power cell. DO NOT DISTRIBUTE THIS PAGE TO YOUR STUDENTS. Use this for your records to check student answers as you circulate the room. Consider setting up a procedure whereby students may only proceed to the next challenge after they check their current answer with you. Or, have groups peer check their answers without using the answer sheet.

Students should work in their groups to solve as many challenges in 30 minutes as they can.

As students are completing the game challenges look for these skills and strategies:

- How students decompose the problem (break it into smaller pieces to solve).
- How students communicate with each other about their logic and reasoning.
- Aspects of Boolean logic where they are more emergent in their thinking.

- Aspects of Boolean logic where they are most fluent and flexible in their thinking.
- Strategies they use for testing and refining their solutions.

While your students are playing share these tips with them:

- Determine what kind of logic statements are presented.
- Once you have solved a clue, or a part of a clue, you will want to see what you can learn about the other clues based on what you already know.
- Work through statements one at a time and then go back and check each statement.

Reflection and Discussion:

- Wrap up the lesson by asking students to reflect and discuss what they learned.
- Ask students to share the most difficult time they had with Boolean Logic and what steps they used to get through the difficulty or challenge.
- How did they go about decomposing the problems or breaking the problem into smaller pieces to solve? When else do you use decomposition to solve a problem outside this game?
- Encourage students to use the vocabulary from the lesson in the discussion.

EXTENSION

This lesson was a great introduction to Boolean Logic. But don't let one hour of code be the only time your students study this important 21st century skill! Take your students' learning deeper with one (or all) of the ThinkFun //CODE Programming Series and other board games, sold online:

//CODE On the Brink

//CODE Robot Repair

//CODE Rover Control

Code Master

Robot Turtles

ThinkFun //CODE PROGRAMMING GAME SERIES

These board games are a perfect way to introduce your students to complex computer science topics without devices! Each game has challenges and are labeled by difficulty making it easy for you to differentiate instruction in your classroom.

The ThinkFun //CODE Programming Game Series is quickly becoming the preferred unplugged activity for K-12 teachers everywhere.