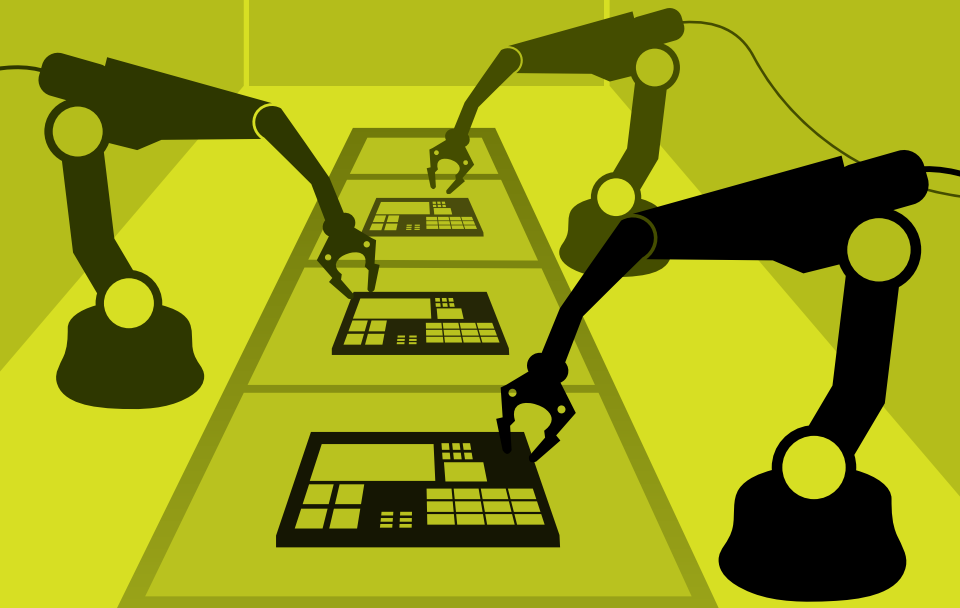
ThinkFun

//CODE

PROGRAMMING GAME SERIES

ROBOT REPAIR



Instruction Manual

AGES 8 to Adult

//CODE ROBOT REPAIR

PROGRAMMING GAME SERIES

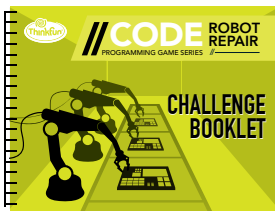
Coding is like a superpower. It might be the reason you can make a robot work. It might even help you do something really big for the world, like send a rocket to Mars. One thing is for sure, if you learn to code, you will never be out of things to do with your new super skill.

ROBOT REPAIR is the third in a series of games designed to build the mental skills needed to fully grasp the concept of coding. All of the games are screen-free for a unique, unplugged play experience. Take them anywhere you go, and before you know it, you'll be ready to CODE!

Visit **www.thinkfun.com/learn-coding** for video instructions, solution strategies, and more great resources from the inventor!

Includes

4 Circuit Boards



40 Challenge Booklet

Solution Booklet

20 TRUE/FALSE Tokens

9 ON/OFF Tokens

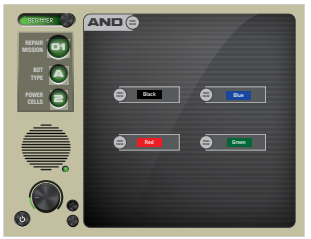
10 Power Cell Tokens

Game Boards

The four Game Boards each contain a Circuit Board consisting of colored wires and numbered Nodes. Place Power Cell Tokens on the numbered Nodes to repair the Robots.

Challenge Booklet

Each of the 40 Challenge Cards contains a series of clues about where to place Power Cells on the Circuit Boards.



Challenge Card

TRUE/FALSE Tokens

These double-sided Tokens represent sensors which report on the state of the colored wires on a Circuit Board.

ON/OFF Tokens

Place the double-sided ON/OFF Tokens on the designated spots on a Circuit Board to track which wires must be powered ON, and which must remain OFF.

Power Cell Tokens

The double-sided Power Cell Tokens have two states. When a Power Cell is placed face up on a Node, it serves as a power source to all wires that it directly touches.

Face Up
(Power Source)

Face Down
(NOT a Power Source)

How to Play

You are at a shop full of Robots that need to be repaired and reactivated. Their Power Cells have been removed from their Circuit Boards and your mission is to reprogram each of them!

For each repair, you'll work on one of four bot types, placing 1-5 Power Cells on the corresponding Circuit Board. Repair instructions come in the form of clues, and for each Challenge, you'll need to use the information given in a large Clue Panel to deduce where to place the Power Cells.

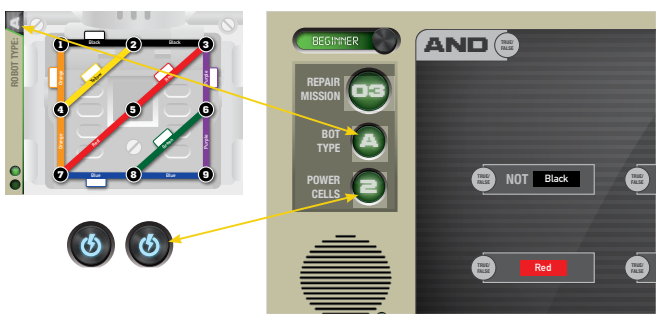
Your Goal

Place Power Cells onto Circuit Board Nodes to create power sources that will turn on specific wires.

When your Power Cells power the combination of wires that make a Clue Panel TRUE, thereby activating the Robot – **YOU WIN!**

Set-Up

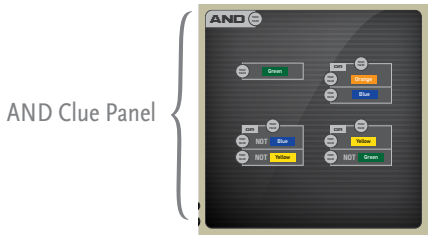
1. Select a Challenge Card and corresponding Circuit Board, listed under "BOT TYPE."
2. Select the number of Power Cells listed, and place them face up next to you. Place any remaining Power Cells face down. Keep all other Tokens accessible.



Game Overview

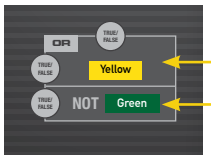
Each Challenge contains one large AND Clue Panel. Each AND Clue Panel contains a series of smaller panels.

Example:

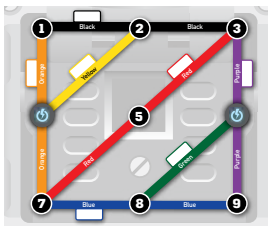


Each of the smaller panels contains at least one clue statement about a colored wire.

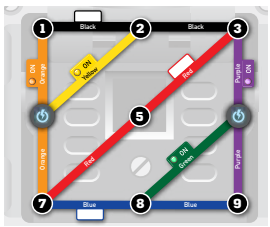
Example:



The placement of face up Power Cells on the Nodes of a Circuit Board determines which wires are ON and OFF (see "Wires and Nodes", page 7).



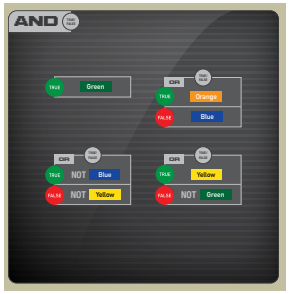
Face up Power Cells are placed on Nodes 4 and 6.



This turns ON the orange and yellow wires,and the purple and green wires, which can now be marked with their corresponding ON Tokens.

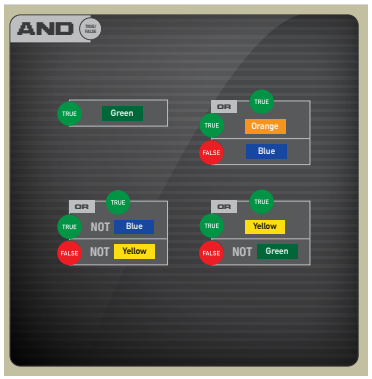
The wires that are ON and OFF then determine which clue statements are TRUE and which are FALSE (See “Wires and Clue Statements”, page 7).

These TRUE/FALSE Tokens mark which clue statements are TRUE and which are FALSE when the green, orange, and yellow wires are ON, and the blue wire is OFF.



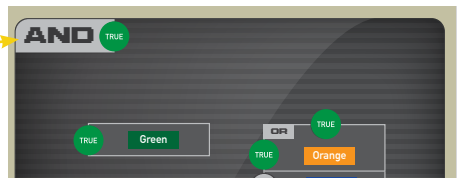
Each panel with more than one clue statement has the name of a logic operation in the upper-left corner. These tell you how to combine the TRUE/FALSE values of the clue statements to determine whether the panel is TRUE or FALSE (see “Interpreting a Clue Panel”, page 8).

When you have confirmed that the placement of TRUE and FALSE Tokens makes a clue panel TRUE, place a TRUE Token in the center of that panel.



When you have placed the Power Cells in such a way that the outer AND Panel evaluates to TRUE - **YOU WIN!**

This outer AND Panel becomes TRUE when all the smaller panels within it are TRUE.



There is only one placement of Power Cells that will cause the AND Panel to be TRUE, and we’ll show you deductive reasoning tricks you can use to find the correct placement of Power Cells (see “Solving Process”, page 11).

Wires and Nodes

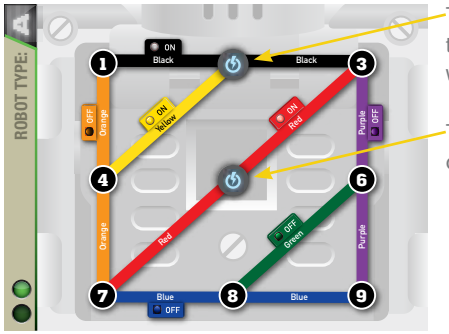
A Circuit Board consists of a set of colored wires and numbered Nodes. The placement of Power Cells on Nodes determines which wires are ON and OFF. Placing a Power Cell face up on a Node turns on ALL the wires intersecting that Node. This is the only way wires are turned on. Only one Power Cell is needed to turn on a wire. A wire with no Power Cells on its Nodes will remain off.

NOTE: You may place more than one Power Cell on the same wire.

Wires and Clue Statements

Each clue statement contains information about whether a wire is ON or OFF.

Example:



This Power Cell turns the black and yellow wires ON

This Power Cell turns only the red wire ON

Some clue statements are just the name of a color. Such statements are TRUE if that colored wire is ON, FALSE if that colored wire is OFF.

This is TRUE because the yellow wire is ON.

This is FALSE because the green wire is OFF.

If the clue statement has a NOT in front of the color, this reverses the meaning.

Continuing with the example on the previous page:

This is FALSE because the yellow wire is ON.

This is TRUE because the green wire is OFF.

IMPORTANT: Not all wires turned on a by a Power Cell will have corresponding clue statements. As long as you have met all the conditions of an AND Panel, additional wires may be powered on.

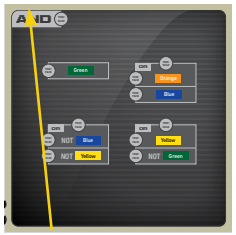
Interpreting a Clue Panel

The AND Clue Panel and each of the smaller panels that contain more than one clue statement are labeled with an AND, OR, IFF, XOR, NOR, or NAND logic gate.

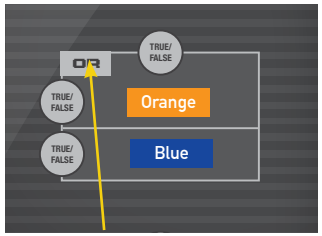
Review the logic gate definitions and examples that follow.

Understanding these logic gates is the key to figuring out which wires must be turned ON in order to repair a Robot, and thus, key to determining which Nodes must contain Power Cells.

Example:



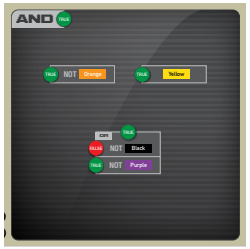
AND Logic Gate



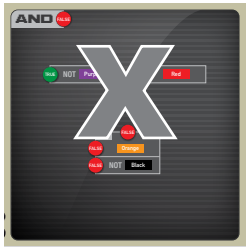
OR Logic Gate

AND

An **AND** Panel is TRUE when **all** of the panels inside it are TRUE



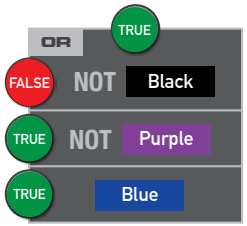
True OR Panel



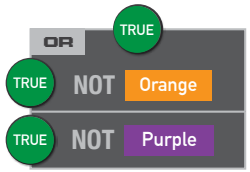
False OR Panel

OR

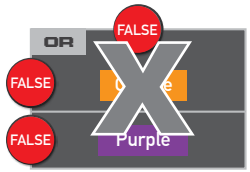
An **OR** Panel is TRUE when **at least one** clue statement on the panel is TRUE.



True OR Panel



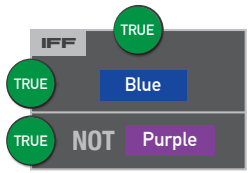
True OR Panel



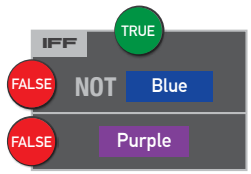
False OR Panel

IFF

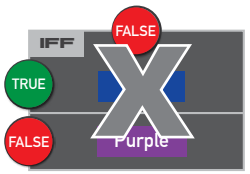
IFF means "if and only if." An **IFF** Panel always has two clue statements. An IFF Panel is TRUE when the clue statements are **both** TRUE or both FALSE.



True IFF Panel



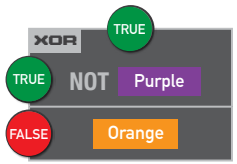
True IFF Panel



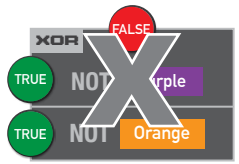
False IFF Panel

XOR

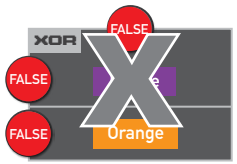
XOR stands for "Exclusive Or." An **XOR** Panel always has two clue statements. An **XOR** Panel is TRUE when **one** statement is TRUE and the other is FALSE.



True XOR Panel



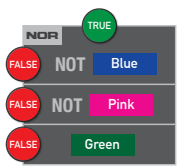
False XOR Panel



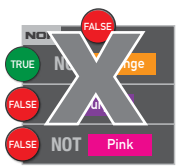
False XOR Panel

NOR

NOR means "NOT OR." A **NOR** Panel is TRUE when **all** the clue statements are **FALSE**.



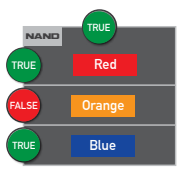
True NOR Panel



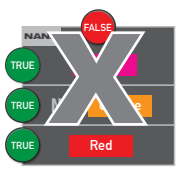
False NOR Panel

NAND

NAND means "NOT AND." A **NAND** Panel is TRUE when **at least one** clue statement is **FALSE**.



True NAND Panel



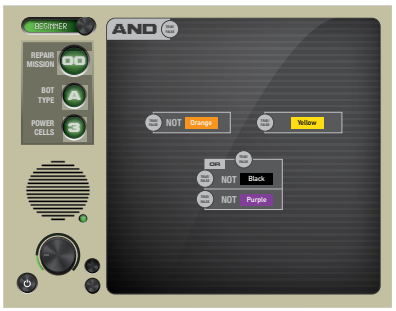
False NAND Panel

You're now ready to repair the Robots! Start playing now, or first, review the Sample Challenge on page 11 and the Deduction Tips on Page 14.

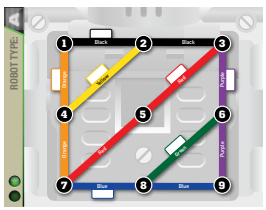
Solving Process - SAMPLE CHALLENGE

As you make deductions about which wires are ON and OFF, the included Tokens will help you keep track of all the information you know. Follow the steps below to learn how to use the Tokens to your advantage!

Sample Challenge Card:



Step 1: Gather the Circuit Board and the number of Power Cells listed on the Challenge Card.



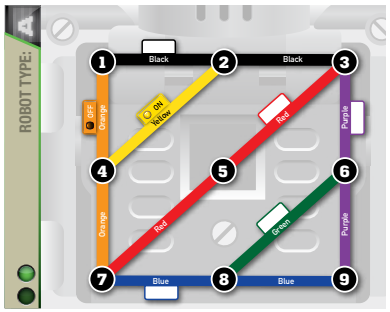
Circuit Board for BOT TYPE A

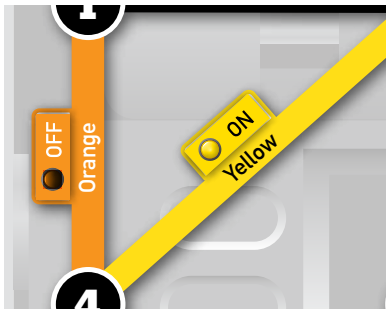
Three Power Cells (face up)

Step 2: Examine the Clue Panel. To make the AND Panel TRUE, all the smaller clue panels inside it must be TRUE. **Start with any clue panels that contain only one clue statement. You know these must be TRUE.**

In this example, we know the top two clue panels must be TRUE.

Step 3: Once you have definitive information about at least one wire, you can use the double-sided ON/OFF Tokens to mark this data on the Circuit Board.



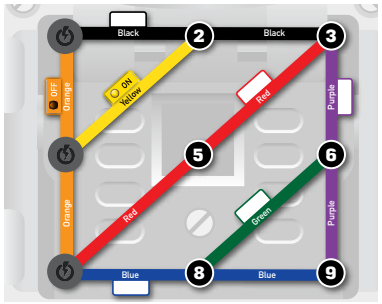


From the clue panels on the previous page, you know that the orange wire is OFF, and the yellow wire is ON.

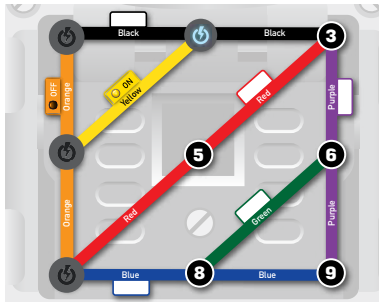
Step 4: Examine the Circuit Board to see if this wire data can help you place any Power Cells on the board.

Anytime you mark a wire as OFF, you can begin to place face down Power Cells along that wire's Nodes to mark where a power source can't go.

Face Down Power Cell

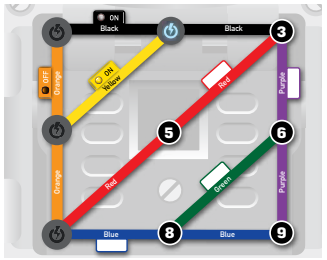


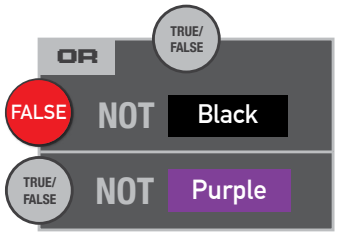
Because you know the orange wire is OFF, you can place face down Power Cells on Nodes 1, 4, and 7.



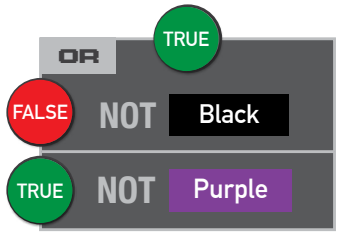
You also know that the yellow wire is ON, so the only Node left to turn on the yellow wire is Node 2. Place one of your three face up Power Cells here.

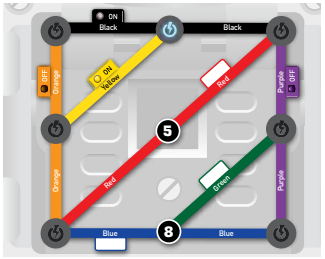
Step 5: Examine the Circuit Board to see if you have any new info about other wires and, in turn, clue statements.





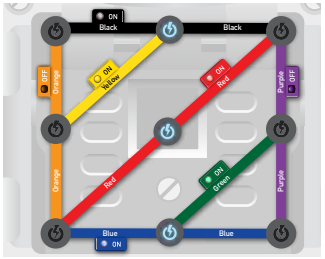
You placed a Power Cell on Node 2, which also activated the black wire. Thus, the clue statement "NOT Black" must be FALSE.





To make the OR Panel TRUE, at least one clue statement must be TRUE. Thus, it must be TRUE that the purple wire is OFF. Place an OFF Token on the purple wire and Power Cells face down on Nodes 3, 6, and 9.

Step 6: Use any new info to place additional Power Cells.



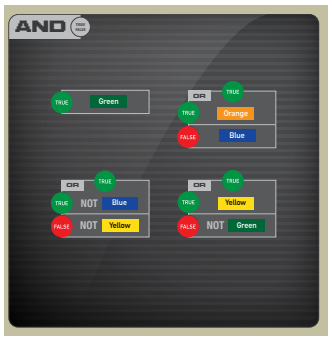
There are only two Nodes available to place Power Cells. Place your two remaining Power Cells face up on Nodes 5 and 8, and – **YOU WIN!**

Deduction Tips

Tip #1

For the AND Panel to be TRUE, all the panels inside it must be TRUE.

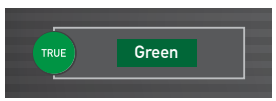
Example:



Tip #2

All the smaller panels within a large AND Panel must be TRUE. This means a panel with only one clue statement must be TRUE.

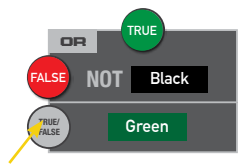
Example:



Tip #3

When you are sure that all but one clue statement in an OR Panel is FALSE, the last clue statement must be TRUE to make the OR Panel TRUE.

Example:

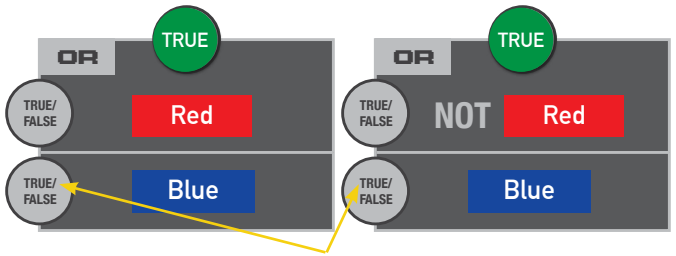


This statement must be TRUE

Tip #4

When you see two True OR Panels, each with two clue statements, where one statement is the opposite between panels and the other statement is the same between panels, the matching statement must be TRUE.

Example:



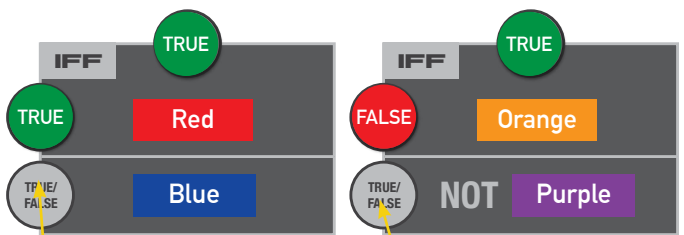
These statements must be TRUE

In this example, the "Red" and "NOT Red" clue statements are opposites. This means one is TRUE, and one is FALSE. If one of these is FALSE, you can deduce that the "Blue" clue statement is TRUE because at least one clue statement in an OR Panel must be TRUE.

Tip #5

When you discover that one clue statement in a True IFF Panel is TRUE, the other statement must be TRUE. When you discover that one statement in a True IFF Panel is FALSE, the other must be FALSE.

Example:



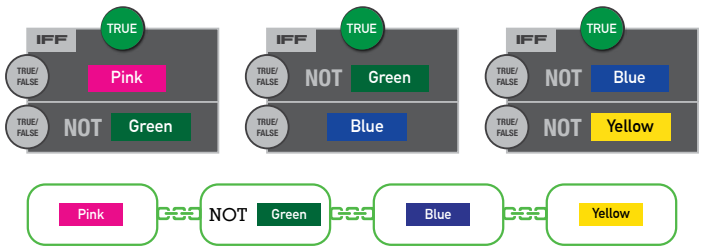
This statement must be TRUE

This statement must be FALSE

Tip #6

Chain together IFF Panels to discover several clue statements that must be ALL TRUE or ALL FALSE together. Then try both possibilities to see which one works.

Example:



These are ALL TRUE or ALL FALSE

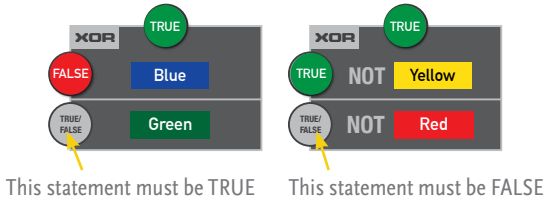
In this example, looking at the first two panels, you can see that Pink is equivalent to NOT Green, and NOT Green is equivalent to Blue. Now looking at the third panel, you see that NOT Blue is equivalent to NOT Yellow. NOT Blue is the opposite of Blue, which means the opposite of NOT Yellow is equivalent to Blue.

Tip #7

When you discover that one clue statement in a True XOR Panel is FALSE, the other must be TRUE.

When you discover that one clue statement in a True XOR Panel is TRUE, the other must be FALSE.

Example:



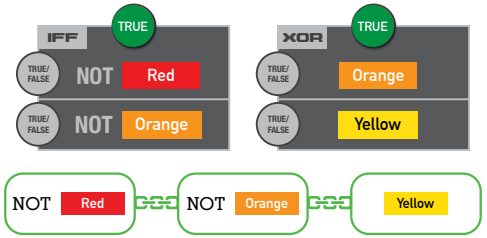
This statement must be TRUE

This statement must be FALSE

Tip #8

Any two IFF or XOR Panels that share a color can be chained together.

Example:



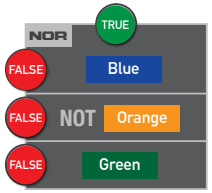
These are ALL TRUE or ALL FALSE

In this example, the IFF Panel tells us that NOT Red and NOT Orange are equivalent. The XOR Panel tells us that Orange and Yellow are opposites. This means that NOT Orange and Yellow are equivalent. Thus, NOT Red, NOT Orange, and Yellow are all equivalent and are ALL TRUE or ALL FALSE.

Tip #9

In the Expert Challenges, start by making the NOR Panel TRUE. All clue statements in a NOR Panel must be FALSE.

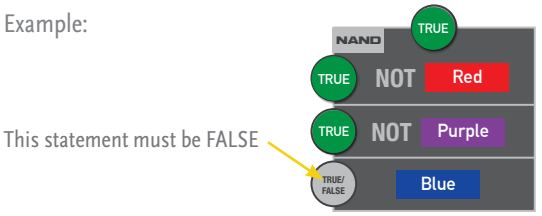
Example:



Tip #10

When you are sure that all but one clue statement in a True NAND Panel are TRUE, the last statement must be FALSE.

Example:



Message for Parents and Educators

The idea that all kids should learn coding skills is one of today's fastest growing educational movements. ROBOT REPAIR is a fun, hands-on way to develop computational thinking abilities and build the mental skills necessary to understand the concept of coding.

The ThinkFun //Code Programming Game Series

This Programming Game Series is designed to take players on a grand tour of the big ideas in computer science. If you play all the games in our series, you will develop a love for concepts such as procedures, if/then statements, while loops, for loops, state machines, logic gates, and Boolean satisfiability, while learning how to mentally evaluate logical expressions and control structures the way computers do. Most importantly, these games will start you on a fun and thrilling journey into the realm of computational thinking. Enjoy the ride!

A Closer Look at ROBOT REPAIR

The true beauty of coding lies in its simplicity. It's pretty remarkable that a binary code of 0's and 1's can generate the many complex outputs that you get from a computer. Or that two basic answers can control much of a computer's output. This is the magic behind Boolean logic—a form of mathematics commonly used in computer programming in which the only values used are true and false. ROBOT REPAIR familiarizes players with this form of logical thinking and has them deciphering clues based on the three basic operations in Boolean logic – AND, OR, and NOT statements.

ROBOT REPAIR is based on a famous NP-complete problem known as the Boolean Satisfiability Problem (SAT for short). NP-complete problems are some of the most interesting problems in computer science, because no one knows whether it is possible to write a program that can solve large instances quickly. As players journey through this collection of challenges, they learn the truth tables for all the standard logic gates, and discover a number of important deduction principles that are used by SAT solvers to tackle some of the world's most complex computational problems.

About the Inventor

Mark Engelberg is the inventor of ThinkFun's award-winning coding game Code Master as well as one of the challenge developers for ThinkFun's blockbuster game Rush Hour®. To create Code Master™ and the //CODE Programming Game Series he drew on his experience as a programmer of virtual reality simulations for NASA, as well as his many years of experience as a teacher of computer science and mathematical logic. Mark believes that kids of all ages can and should learn how a computer executes programs - entirely through play!

ThinkFun's Mission is to Ignite Your Mind!®

ThinkFun® is the world's leader in addictively fun games that stretch and sharpen your mind. From lighting up young minds to creating fun for the whole family, ThinkFun's innovative games and mobile apps make you think while they make you smile.

www.ThinkFun.com

MADE IN CHINA, 150. #1903. IN01.