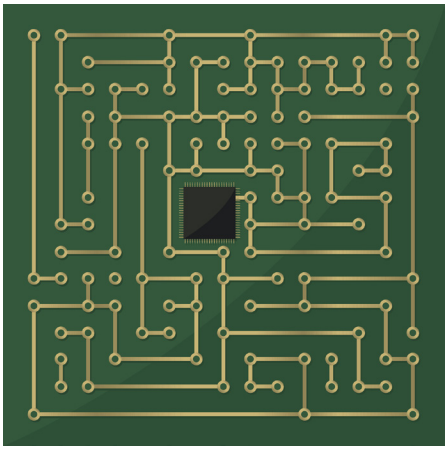
TATA CONSULTANCY SERVICES | Discovery EDUCATION

IGNITE MY FUTURE

FAMILY ACTIVITY

Secret Code for Mazes



ACTIVITY TIME
30-45 minutes

MATERIALS NEEDED

- Pencil
- Paper
- Secret Code Sample Maze worksheet
- A set of mazes (optional)

Background Information

It's a scene we see all the time in TV and movies: the villain is outsmarting the hero and causing chaos by messing with computer systems. Suddenly, a tech whiz comes in and we see a computer with a black screen and lots of green numbers rapidly blinking. The moment is tense—will the tech whiz save the day?

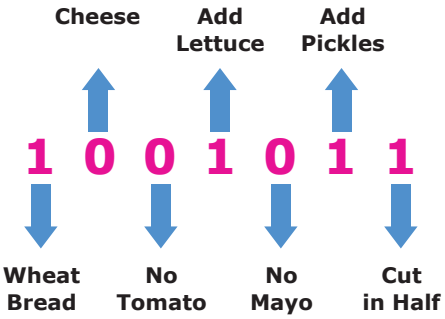
But, wait a minute—what ARE those green numbers flashing on the screen?

Those green numbers taking over the computer screen are a language called binary code. Binary code is the language computers use to complete tasks.

Here is an example of a binary sequence:

1 0 0 1 0 1 1

A sequence like this can provide a computer with a lot of information, but only if the computer knows how to read and understand it. For example, what if the sequence was meant to describe the type of sandwich a customer orders at a restaurant? The first digit in the sequence could represent the type of bread: 0 for white bread, 1 for wheat bread. The next digit might represent the type of cheese: Havarti (0) or Cheddar (1). The next four digits could represent whether or not the customer wants toppings like tomato, lettuce, mayonnaise, and pickles. The last digit represents whether the customer wants the sandwich cut in half or not. If the computer knows what each digit is supposed to mean, then the binary sequence can communicate exactly what type of sandwich the customer wants:



At the most basic level, all computer programs can be broken down to a binary sequence. Those sequences can be very complex. Many industries are looking for professionals who can think like computers in order to build programs. Understanding how computers think in binary sequences is a useful skill for all computer programmers. Many engineering fields, such as the fields of automation and robotics, also use these skills. The proper application of these skills can also make manufacturing and infrastructure design more efficient. Can we use this perspective to look at real-life scenarios and find a way to describe them using only 0s and 1s?

In this activity, you will work together as a family to solve a maze using binary code. Together, you'll learn how to analyze the maze the way a computer might and then communicate the solution to the maze using a binary sequence.

Blueprint for Discovery: BEGINNER

Phase 1: Complete the Secret Code Sample Maze

1. **Trace** out the solution on the Secret Code Sample Maze worksheet included in this activity.

Phase 2: Turn the solution into binary code

1. **As a family**, see if you can create a solution "code" that expresses the solution pathway using only 0s and 1s. To do this, you will first need to discuss and decide on a set of rules for understanding the "code" you are going to create. Locate the points in the maze at which you had to decide about which way to go (such as "left or right" and "up or down"). These are the decisions you can represent with 0s and 1s. Decide—what will a 0 mean? What will a 1 mean?

2. **Next**, express the solution to the sample maze using the binary code system you have created.

3. **There are many ways you could use binary code** to describe the solution. For an example of how this could be done, complete the Binary Maze Example Solution. Is the system of code represented on the solution handout the same as yours, is it similar, or is it different?

Phase 3: Test your code system

1. **Let each member of the family draw** a simple maze to use for testing your new system of code. For now, keep the mazes fairly simple by making sure each decision point has only two possible options.

2. **Let each member of the family provide** a binary code that represents the solution to the maze he or she created.

3. **Exchange mazes** with each other. Follow the binary sequence that was provided with the maze you received, and see if it brings you to your destination correctly. If so, you have mastered the skill of communicating in binary code. Congratulations!

Blueprint for Discovery: ADVANCED

Phase 1: Complete the Secret Code Sample Maze

1. **Trace out** the solution on the Secret Code Sample Maze worksheet included in this activity.

Phase 2: Turn the solution into binary code

1. **As a family**, see if you can create a solution "code" that expresses the solution pathway using only 0s and 1s. To do this, you will first need to discuss and decide on a set of rules for understanding the "code" you are going to create. Locate the points in the maze at which you had to make a decision about which way to go (such as "left or right" and "up or down"). These are the decisions you can represent with 0s and 1s. Decide—what will a 0 mean? What will a 1 mean?
2. **Next**, express the solution to the sample maze using the binary code system you have created.
3. **There are many ways you could use binary code** to describe the solution. For an example of how this could be done, complete the Binary Maze Example Solution. Is the system of code represented on the solution handout the same as yours, is it similar, or is it different?
4. **Create at least one new system of rules** that allow you to use binary code in a different way to create a solution for the maze. See how many different systems you can come up with. Do the various systems have advantages or disadvantages?

Phase 3: Test your code system

1. **Let each member of the family draw** a simple maze to use to test your new system of code. For now, keep the mazes simple by making sure each decision point has only two possible options.
2. **Let each member of the family provide** a binary code that represents the solution to the maze he or she created.
3. **Exchange mazes** with one another. Follow the binary sequence that was provided with the maze you received, and see if it brings you to your destination correctly. If so, you have mastered the skill of communicating in binary code. Congratulations!

Blueprint for Discovery: ADVANCED
Phase 4: More code challenges

1. **Find (or create)** a more complex maze that includes intersections with three options for directions.

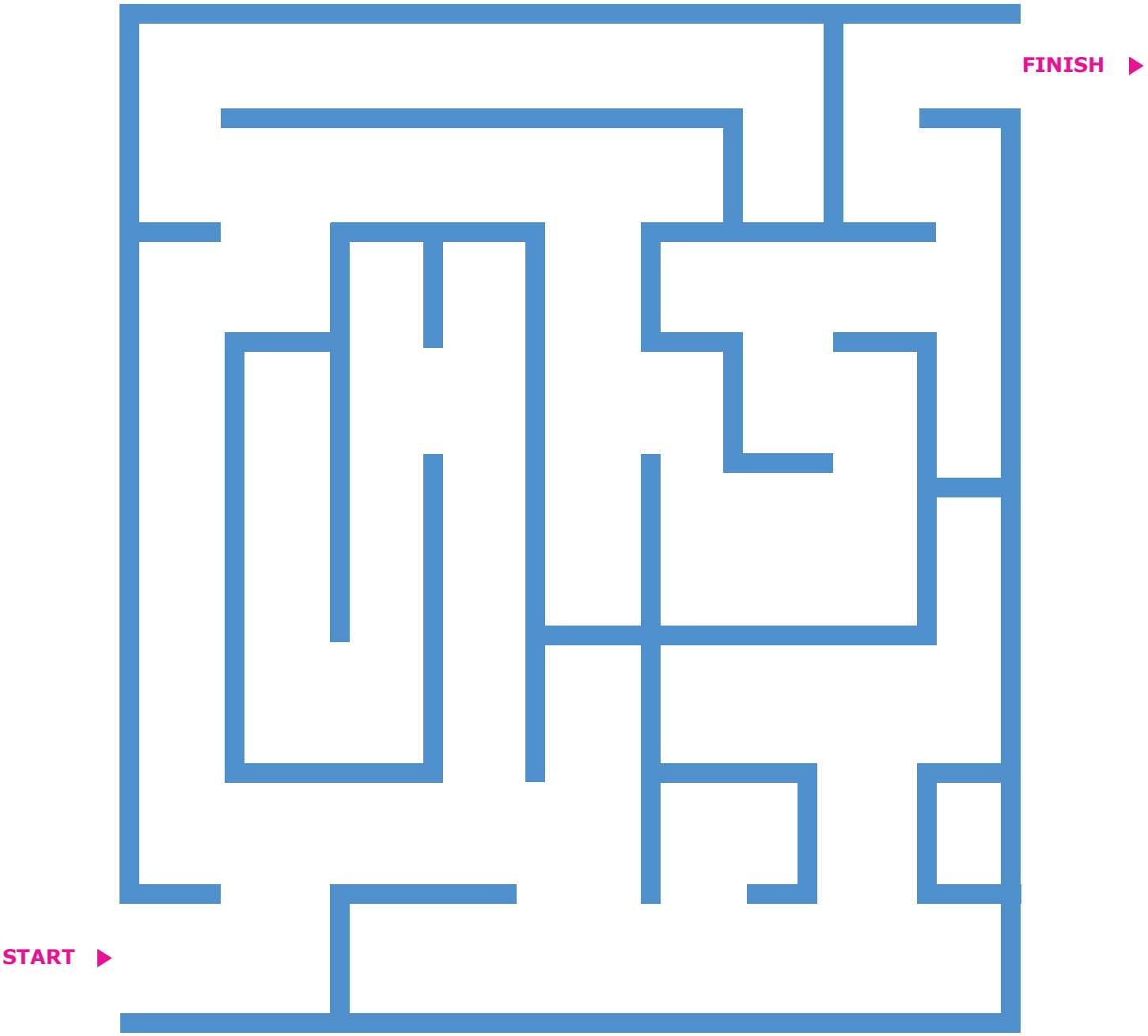
2. **Test your system of code** to see if it will work for the more complex maze. You can test it by letting one family member solve the maze and create a binary code solution, then letting another family member use the solution to solve the maze.

3. **If your existing binary system does not work** for the more complex situation, develop a new system that will work. Try using the following hints (these hints could lead you to multiple new systems):
   - Can you create a system where the digits represent rotation rather than a specific direction to turn?
   - Can you create a system where more than one digit is looked at for each decision?

4. **So far,** our examples have assumed that no special instructions are needed to continue in the maze when there is only one possible direction to go. We have only provided instructions when a decision must be made between two or more possible directions. However, this assumption doesn't always match reality. For example, what if we wanted to provide binary instructions to robotic equipment that couldn't analyze and determine that only one direction was possible? For a final maze challenge, create a binary code system that provides instructions for every step of the way along the maze, including steps when the maze only allows one possible direction.

   Discuss as a family: Can you apply the skill of binary analysis to a situation other than mazes? Think of another activity or situation in your life, and create a system of rules for analyzing it and communicating about it in binary code.

Secret Code Sample Maze



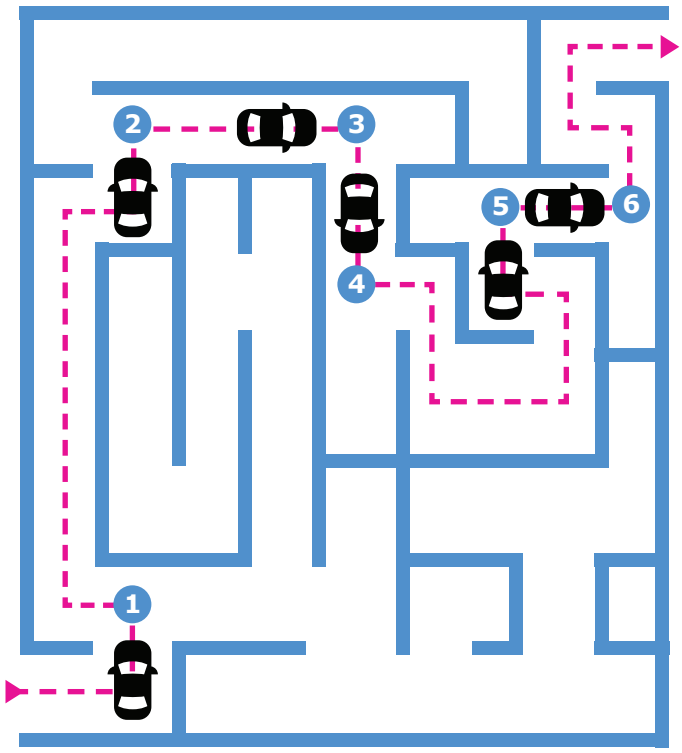
Binary Maze System Capture Sheet

Our rules for interpreting the binary sequence that communicates the solution to the maze:

The binary sequence that represents the solution to the sample maze:

Explain the meaning of each part of your binary sequence according to the rules of interpretation you created:

Binary Maze Example Solution



The solution pathway for the maze is shown above. There are exactly six points along the pathway where a decision must be made about the direction to take. You can use binary code here to make a number (either a 0 or a 1) to represent the decision at each turning point. That means that for this maze, the binary code solution would have six digits because there are six decision points in the maze.

At each of the turning points, we can see that the car has exactly two options for the directions to go. If we compare the two directions, we can assign one of them as toward the left and one of them as toward the right (this makes the binary code apply even when the maze includes an option to go straight). We will represent each "left" choice as a 0 and each "right" choice as a 1.

So, we can summarize the rules for understanding our code as follows: The car must always go in the only available direction until it reaches a point where a decision must be made between two directions:

- When a decision must be made, the current digit in the binary sequence indicates which direction the car should go.
    - If the current digit is a 0, the car should take the available direction that is farthest to the left.
    - If the digit is a 1, the car should take the available direction that is farthest to the right.

Let's apply our system to this maze. For each of the six decisions that are made, circle whether the leftmost or rightmost direction is followed and write the appropriate digit to represent that choice.

Choice 1: **L** or **R** **Binary digit (0 or 1):** _____

Choice 2: **L** or **R** **Binary digit (0 or 1):** _____

Choice 3: **L** or **R** **Binary digit (0 or 1):** _____

Choice 4: **L** or **R** **Binary digit (0 or 1):** _____

Choice 5: **L** or **R** **Binary digit (0 or 1):** _____

Choice 6: **L** or **R** **Binary digit (0 or 1):** _____

Now, put all six of your digits together in a sequence, and you will have your code for the solution to the maze!

____ ____ ____ ____ ____ ____