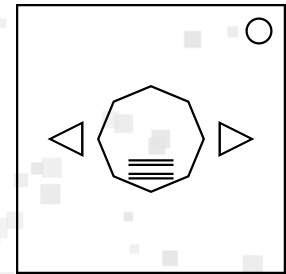


On the Subject of hexNull

*"Everyone likes a well-constructed maze, right? Here's our take on the genre!"
"...we did actually remember to construct a proper exit, right?"*

See Appendix BT in [Binary Tree](https://ktane.timwi.de/HTML/Binary%20Tree.html)
(<https://ktane.timwi.de/HTML/Binary%20Tree.html>) for
binary tree terminology.



This module has two directional buttons and a screen. To activate the module, interact with the display.

When recording any data from this module, note that all variables start at 0.

Upon generation, this module will create a depth-four complete binary tree containing 15 nodes. Pressing the left or right buttons will take you to one of the left or right children of your current node.

Each node on the bottom layer of the tree has a 3-digit binary code that matches with the button inputs needed to reach it, where left is 0, and right is 1. This will be referred to as the node's binary location. (ex: 110 represents index 6.)

Upon interaction with the display, your position will be set to the binary tree's root, and the module will give you an encoded message. This message identifies the binary location of an unnamed entity somewhere on the bottom layer.

	Digit 1 (#--)	Digit 2 (-#-)	Digit 3 (--#)
	Sound is...	Color is...	Line is...
0	Low-pitched	Black	Vertical
1	High-pitched	White	Horizontal

When you communicate with the entity, they will provide you with the binary locations of 4 indexes at the bottom of the tree, where black is a 0, and white is a 1. To find a solution, create an zeroed-out array of 8 binary digits, and for each location shown, flip the digit of that itself and adjacent locations, from 0 to 1, or 1 to 0. (ex: if the entity presented [Black, White, White] (aka: 011), the 8-digit array would become 00111000. **The first and last digit are adjacent.**)

Each node on the tree's bottom layer has a zeroed-out binary digit. Whenever you reach the bottom layer, the digits that represents your location **and the entity's location** are flipped. To solve the module, these binary digits (from left to right) must match those represented by the entity's solution. To submit your answer, interact with the entity after modifying at least one of these digits.

If you submit an incorrect array, the data contained within the bottom nodes will be set back to 0 - though the solution to the module won't change.