XOR Gate

3 min

Now, we are going to create what's called an *XOR gate*, an exclusive or gate. This gate receives two inputs, a and b, and only returns a 1 if one of the inputs is 1, but not if both of the inputs are 1.

To build your XOR_gate(), you should use any combination of the gates you've already made: NAND_gate(), NOT_gate(), AND_gate(), and OR_gate().

Here's the truth table:

a	b	output
0	0	0
0	1	1
1	0	1
1	1	0

Instructions

1. Checkpoint 1 Passed

1.

Define XOR_gate() which takes two inputs, a and b, and returns the outputs specified in the truth table.

Hint

XOR_gate() returns 1 if either a or b is 1, but not if they're both 1.

Push yourself to use the previous gates in creating your XOR_gate()!

One way to do this would be:

AND(NAND(a, b), OR(a, b))

Copy to Clipboard

Can you see why?

script.py

```
from nand import NAND_gate

from not_gate import NOT_gate

from and_gate import AND_gate

from or_gate import OR_gate


# TEST CASES

def XOR_gate(a, b):

  return AND_gate(NAND_gate(a, b), OR_gate(a, b))
```

```
print("A: 0, B: 0 | Output: {0}".format(XOR_gate(0, 0)))
print("A: 0, B: 1 | Output: {0}".format(XOR_gate(0, 1)))
print("A: 1, B: 0 | Output: {0}".format(XOR_gate(1, 0)))
print("A: 1, B: 1 | Output: {0}".format(XOR_gate(1, 1)))
```