

Boolean Circuits and Truth Tables

The following exercises are recommended to be completed with assistance from trutabgen.com and circuitverse.org/simulator

Boolean Algebra

Boolean variables a, b, c can have two values: True or False. They can be joined with operators like **and** (\wedge) and **or** (\vee) and **not** (\bar{a}) to create boolean statements. The value of a boolean statement is the result of performing boolean operations on the input. For example The sentence $a \wedge (b \vee \bar{c})$ can be evaluated when - $a = T, b = T, c = F$: result = T - $a = F, b = F, c = F$: result = ?

Write a full truth table for this problem (by hand, don't use computer). In a truth table you generate a list of all 2^n input possibilities for a boolean statement with n variables and compute the outputs.

a	b	c	$a \wedge (b \vee \bar{c})$
True	True	True	True
True	True	False	True
True	False	True	True
True	False	False	False
False	True	True	False
False	True	False	False
False	False	True	False
False	False	False	False

Manipulating boolean statements

Generate truth tables for each of the following in trutabgen.com. Analyze the output and try to find a different (usually shorter) boolean statement with the same result column. Write the new statement below.

1. $a \vee \bar{a}$
2. $b \wedge \bar{b}$
3. $\overline{a \wedge b}$
4. $\overline{\bar{a} \wedge \bar{b}}$
5. $\overline{a \vee b}$
6. $\bar{a} \vee \bar{b}$
7. $a \vee b \vee (\bar{c} \wedge a)$
8. $b \vee (c \wedge \bar{a}) \wedge \bar{c}$
9. $(a \wedge b) \vee (a \wedge c)$
10. $(a \vee b) \wedge (a \vee c)$

Boolean circuits

Pick 3 of the previous 10 problems. Using circuitverse.org/simulator, make a circuit for each and check that it's correct

The XOR gate

Make a truth table for $(a \wedge \bar{b}) \vee (\bar{a} \wedge b)$. This is called XOR and is written as a XOR b . In symbols, $a \oplus b$

Using the website above, create a digital circuit using the XOR gate that has two input switches and a light. The light should illuminate when exactly one of the switches is on.

Circuit Projects (optional challenge)

Do some or all of the following. In any order. Circle the ones you're trying. Save your answers or print them out when they work.

1. Create a circuit with 3 input switches. An output light should illuminate if and only if exactly one switch is activated.
2. Create a circuit with 4 input switches. An output light should illuminate if and only if more than one switch is activated
3. Create a circuit with 2 input switches and two boolean outputs. The output should represent the binary sum of the two inputs (this is called an *adder* circuit).
4. Create a circuit with at least 4 inputs. An output light should light if an even number of inputs are true.

Circuit-SAT

The circuit satisfiability problem is one of the most famous problems in computer science. It's very easy to state – given a boolean statement with n variables, what is the most efficient algorithm to determine if any setting of the variables results in the statement being **true**. (We call this *satisfying* the statement). How would you solve this problem?