Proofs, Alternate Notation

CSE 311 Autumn 2025
Lecture 4

Announcements

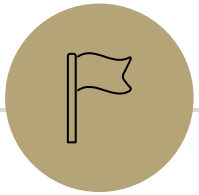
We're putting resources on the resources page:

The list of logical equivalences

A Translation Tips reading (for going from English to Logic)
Might be useful if you're still working on homework 1.

There's a lot of other resources on the page that will be useful as we get further along (e.g. reference sheets for particular topics).

We'll try to remind you about important ones, but it's good to keep in mind that it's there.



Our First Proof

Our First Proof

$$
\begin{aligned}
(a \wedge b) \vee (\neg a \wedge b) \vee (\neg a \wedge \neg b) &\equiv (a \wedge b) \vee [(\neg a \wedge b) \vee (\neg a \wedge \neg b)] && \text{Associative} \\
&\equiv (a \wedge b) \vee [\neg a \wedge (b \vee \neg b)] && \text{Distributive} \\
&\equiv (a \wedge b) \vee [\neg a \wedge \mathrm{T}] && \text{Negation} \\
&\equiv (a \wedge b) \vee [\neg a] && \text{Identity} \\
&\equiv [\neg a] \vee (a \wedge b) && \text{Commutative} \\
&\equiv (\neg a \vee a) \wedge (\neg a \vee b) && \text{Distributive} \\
&\equiv (a \vee \neg a) \wedge (\neg a \vee b) && \text{Commutative} \\
&\equiv \mathrm{T} \wedge (\neg a \vee b) && \text{Negation} \\
&\equiv (\neg a \vee b) \wedge \mathrm{T} && \text{Commutative} \\
&\equiv (\neg a \vee b) && \text{Identity}
\end{aligned}
$$

More on Our First Proof

With practice (and quite a bit of squinting) you can see not just the ironclad guarantee, but also the reason why something is true.

That's not easy with a truth table.

Proofs can also communicate intuition about why a statement is true.

We'll practice extracting intuition from proofs more this quarter.

Our First Proof

$$(a \wedge b) \vee (\neg a \wedge b) \vee (\neg a \wedge \neg b)$$

$\equiv (a \wedge b) \vee [(\neg a \wedge b) \vee (\neg a \wedge \neg b)]$	Associative
$\equiv (a \wedge b) \vee [\neg a \wedge (b \vee \neg b)]$	Distributive
$\equiv (a \wedge b) \vee [\neg a \wedge \mathrm{T}]$	Negation
$\equiv (a \wedge b) \vee [\neg a]$	Identity
$\equiv [\neg a] \vee (a \wedge b)$	Commutative
$\equiv (\neg a \vee a) \wedge (\neg a \vee b)$	Distributive
$\equiv (a \vee \neg a) \wedge (\neg a \vee b)$	Commutative
$\equiv \mathrm{T} \wedge (\neg a \vee b)$	Negation
$\equiv (\neg a \vee b) \wedge \mathrm{T}$	Commutative
$\equiv (\neg a \vee b)$	Identity

The last two terms are "vacuous truth" – they simplify to $\neg a$

a no longer matters in $a \wedge b$ if $\neg a$ automatically makes the expression true.

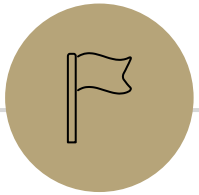
More on Our First Proof

With practice (and quite a bit of squinting) you can see not just the ironclad guarantee, but also the reason why something is true.

That's not easy with a truth table.

Proofs can also communicate intuition about why a statement is true.

We'll practice extracting intuition from proofs more this quarter.



Modifying Implications

Converse, Contrapositive

Implication:

If it's raining, then I have my umbrella.

$p \rightarrow q$

Converse:

If I have my umbrella, then it is raining.

$q \rightarrow p$

Contrapositive:

$\neg q \rightarrow \neg p$ If I don't have my umbrella, then it is not raining.

Inverse:

$\neg p \rightarrow \neg q$ If it is not raining, then I don't have my umbrella.

How do these relate to each other?

p	q	$p \rightarrow q$	$q \rightarrow p$	$\neg p$	$\neg q$	$\neg p \rightarrow \neg q$	$\neg q \rightarrow \neg p$
T	T						
T	F						
F	T						
F	F						

Converse, Contrapositive

Implication:

$p \rightarrow q$

Contrapositive:

$\neg q \rightarrow \neg p$

Converse:

$q \rightarrow p$

Inverse:

$\neg p \rightarrow \neg q$

An implication and its contrapositive have the same truth value!

p	q	$p \rightarrow q$	$q \rightarrow p$	$\neg p$	$\neg q$	$\neg p \rightarrow \neg q$	$\neg q \rightarrow \neg p$
T	T	T	T	F	F	T	T
T	F	F	T	F	T	T	F
F	T	T	F	T	F	F	T
F	F	T	T	T	T	T	T

Contrapositive

We showed $p \rightarrow q \equiv \neg q \rightarrow \neg p$ with a truth table. Let's do a proof.

Try this one on your own. Remember

1. Know what you're trying to show.
2. Stay on target – take steps to get closer to your goal.

Hint: think about your tools.

There are lots of rules with AND/OR/NOT,

but very few with implications…

Contrapositive

$$
\begin{aligned}
p \to q &\equiv \neg p \vee q && \text{Law of Implication} \\
&\equiv q \vee \neg p && \text{Commutativity} \\
&\equiv \neg\neg q \vee \neg p && \text{Double Negation} \\
&\equiv \neg q \to \neg p && \text{Law of Implication}
\end{aligned}
$$

All of our rules deal with ORs and ANDs, let's switch the implication to just use AND/NOT/OR.

And do the same with our target

It's ok to work from both ends. In fact it's a very common strategy!

Now how do we get the top to look like the bottom?

Just a few more rules and we're done!

Work from both ends, but…

…make sure at the end, if you read from top-to-bottom, every step makes sense.

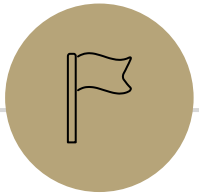
When proving an equivalence you must:
1. Start with the left side (or right side)

2. Modify what you had in the last step (using an equivalence)

3. Derive the right side (or left side if you started with the right)

You may **not** start with the equivalence you're trying to show, and simplify to something "obviously true."

More on why later in the quarter, but tl;dr for now is you can't use your goal as a starting assumption (it's what you're trying to show! If you knew it, no need to write a proof).



Digital Logic

Vocabulary!

A proposition is a....

Tautology **if it is always true.**
Contradiction **if it is always false.**
Contingency **if it can be both true and false.**

$p \vee \neg p$

Tautology

If p is true, $p \vee \neg p$ is true; if p is false, $p \vee \neg p$ is true.

$p \oplus p$

Contradiction

If p is true, $p \oplus p$ is false; if p is false, $p \oplus p$ is false.

$(p \rightarrow q) \wedge p$

Contingency If p is true and q is true, $(p \rightarrow q) \wedge p$ is true;
If p is true and q is false, $(p \rightarrow q) \wedge p$ is false.

On notation…

Logic is fundamental. Computer scientists use it in programs, mathematicians use it in proofs, engineers use it in hardware, philosophers use it in arguments,….

…so everyone uses different notation to represent the same ideas.

Since we don’t know exactly what you’re doing next, we’re going to show you a bunch of them; but don’t think one is “better” than the others!

Digital Circuits

Computing With Logic

T corresponds to 1 or "high" voltage

F corresponds to 0 or "low" voltage

Gates

Take inputs and produce outputs (functions)

Several kinds of gates

Correspond to propositional connectives (most of them)

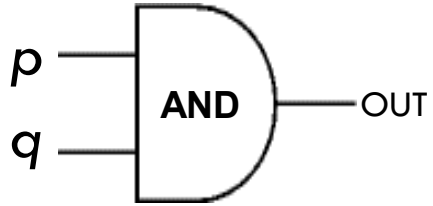
And Gate

AND Connective vs. **AND Gate**

$p \wedge q$

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

p	q	OUT
1	1	1
1	0	0
0	1	0
0	0	0



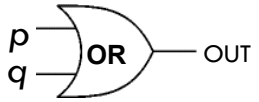
"block looks like D of AND"

Or Gate

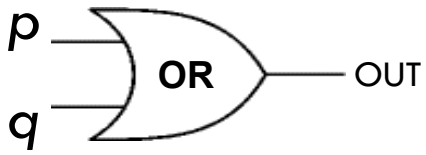
OR Connective vs. **OR Gate**

$p \vee q$

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F



p	q	OUT
1	1	1
1	0	1
0	1	1
0	0	0



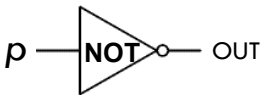
"arrowhead block looks like V"

Not Gates

NOT Connective vs. **NOT Gate**

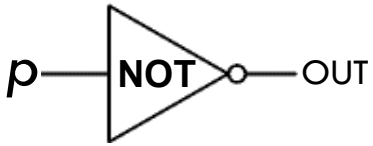
$\neg p$

p	$\neg p$
T	F
F	T



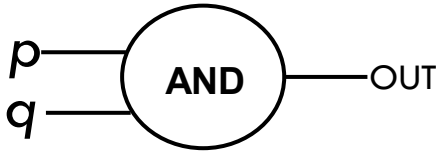
Also called *inverter*

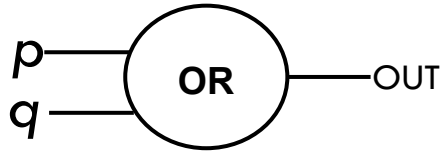
p	OUT
1	0
0	1

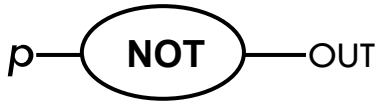


Blobs are Okay!

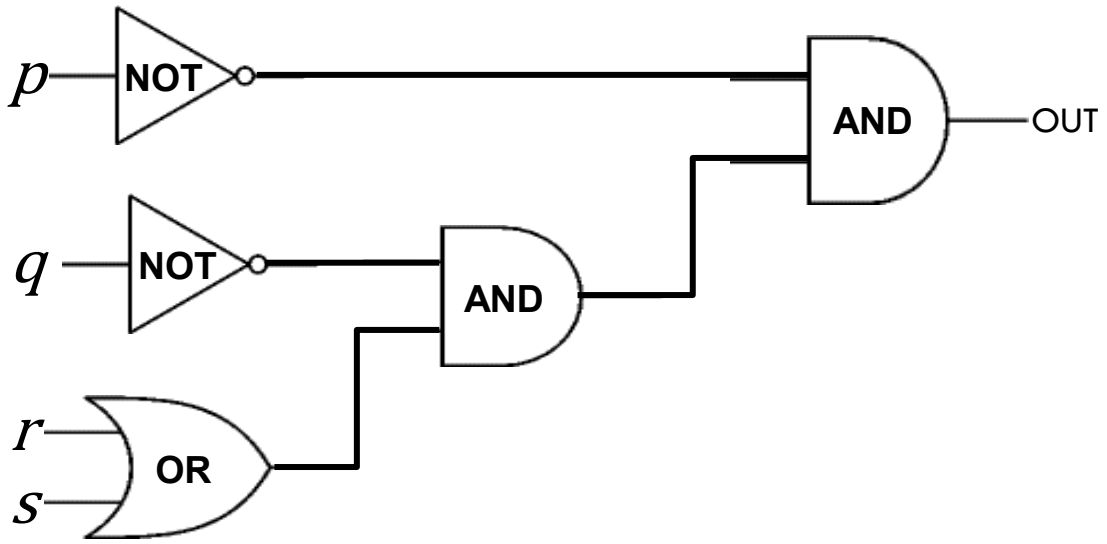
You may write gates using blobs instead of shapes!





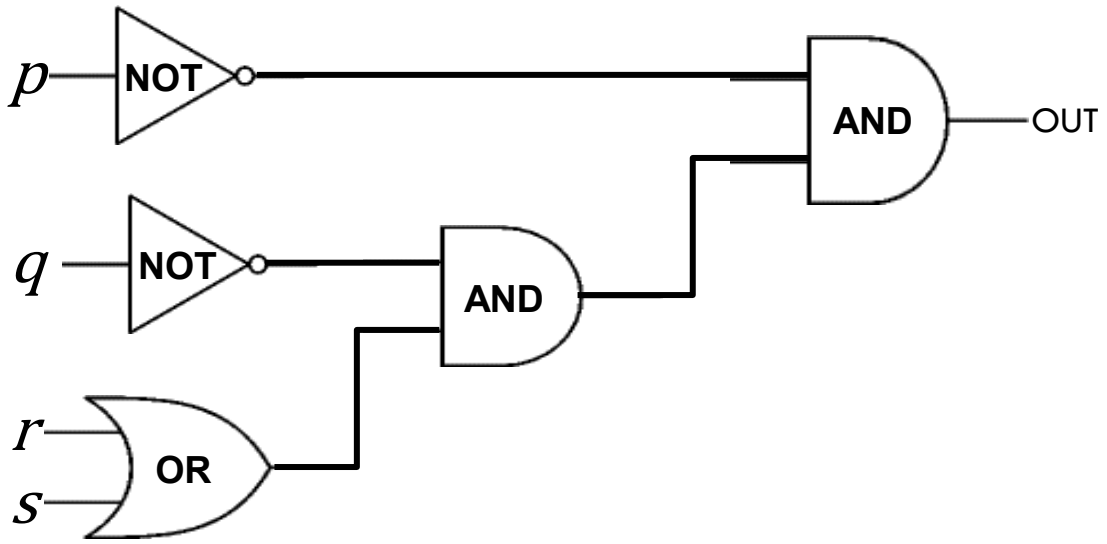


Combinational Logic Circuits



Values get sent along wires connecting gates

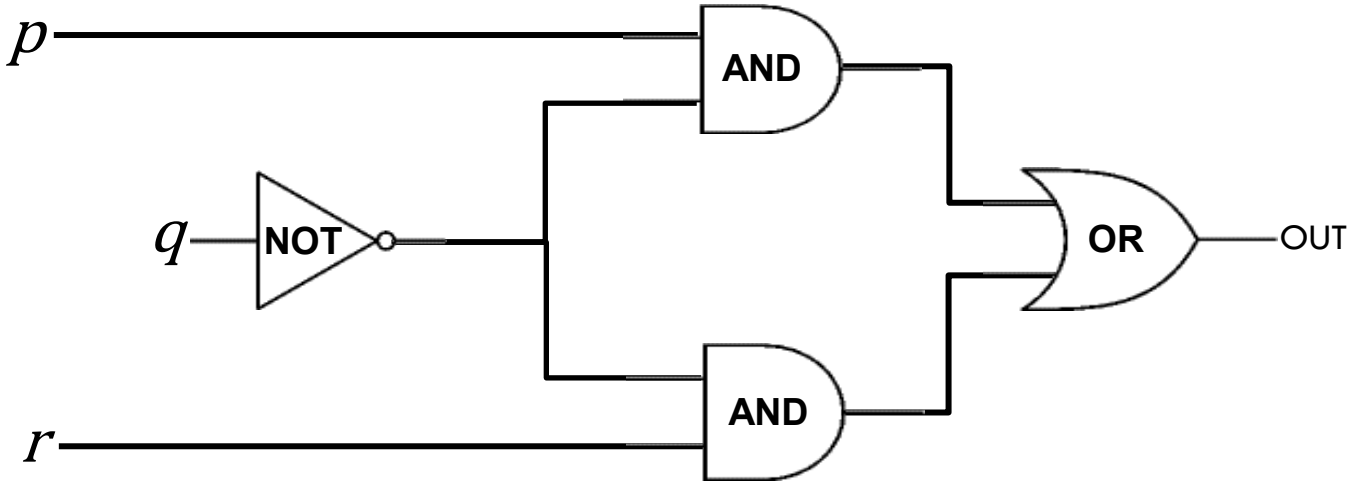
Combinational Logic Circuits



Values get sent along wires connecting gates

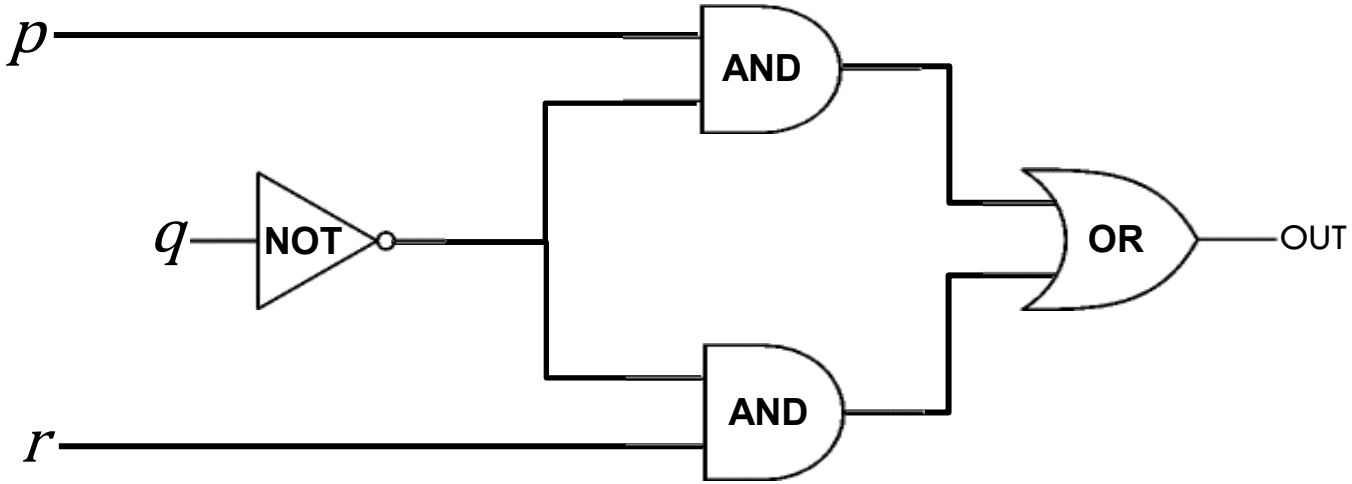
$$\neg p \wedge (\neg q \wedge (r \vee s))$$

Combinational Logic Circuits



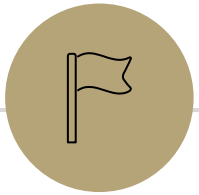
Wires can send one value to multiple gates!

Combinational Logic Circuits



Wires can send one value to multiple gates!

$$(p \wedge \neg q) \vee (\neg q \wedge r)$$



More Vocabulary

Meet Boolean Algebra

Preferred by some mathematicians and circuit designers.

"or" is +

"and" is · (i.e. "multiply")

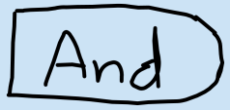
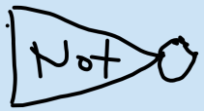
"not" is ' (an apostrophe after a variable)

Why?

Mathematicians like to study "operations that work kinda like 'plus' and 'times' on integers."

Circuit designers have a lot of variables, and this notation is more compact.

Meet Boolean Algebra

Name	Variables	"True/False"	"And"	"Or"	"Not"	Implication
Java Code	`boolean b`	`true, false`	`&&`	`||`	!	No special symbol
Propositional Logic	$"p, q, r"$	`T, F`	$\wedge$	$\vee$	$\neg$	$\rightarrow$
Circuits	Wires	1, 0				No special symbol
Boolean Algebra	a, b, c	1,0	$\cdot$ ("multiplication")	$+$ ("addition")	$'$ (apostrophe after variable)	No special symbol

Propositional logic

$(p \wedge q \wedge r) \vee s \vee \neg t$

Boolean Algebra

$pqr + s + t'$

Comparison

Propositional logic	Boolean Algebra
$(p \wedge q \wedge r) \vee s \vee \neg t$	$pqr + s + t'$

Remember this is just an alternate notation for the same underlying ideas.

So that big list of identities? Just change the notation and you get another big list of identities!

Sometimes names are different (“involution” instead of “double negation”), but the core ideas are the same.

Boolean Algebra

Axioms

Closure

$a + b$ is in $\mathbb{B}$

$a \bullet b$ is in $\mathbb{B}$

Commutativity

$$a + b = b + a$$

$$a \bullet b = b \bullet a$$

Associativity

$$a + (b + c) = (a + b) + c$$

$$a \bullet (b \bullet c) = (a \bullet b) \bullet c$$

Identity

$$a + 0 = a$$

$$a \bullet 1 = a$$

Distributivity

$$a + (b \bullet c) = (a + b) \bullet (a + c)$$

$$a \bullet (b + c) = (a \bullet b) + (a \bullet c)$$

Complementarity

$$a + a' = 1$$

$$a \bullet a' = 0$$

Boolean Algebra

Theorems

Null

$$X + 1 = 1$$
$$X \bullet 0 = 0$$

Idempotency

$$X + X = X$$
$$X \bullet X = X$$

Involution

$$(X')' = X$$

Uniting

$$X \bullet Y + X \bullet Y' = X$$
$$(X + Y) \bullet (X + Y') = X$$

Boolean Algebra

Absorbtion

$$X + X \bullet Y = X$$
$$(X + Y') \bullet Y = X \bullet Y$$
$$X \bullet (X + Y) = X$$
$$(X \bullet Y') + Y = X + Y$$

DeMorgan

$$(X + Y + \cdots)' = X' \bullet Y' \bullet \cdots$$
$$(X \bullet Y \bullet \cdots)' = X' + Y' + \cdots$$

Consensus

$$(X \bullet Y) + (Y \bullet Z) + (X' \bullet Z) = X \bullet Y + X' \bullet Z$$
$$(X + Y) \bullet (Y + Z) \bullet (X' + Z) = (X + Y) \bullet (X' + Z)$$

Factoring

$$(X + Y) \bullet (X' + Z) = X \bullet Z + X' \bullet Y$$
$$X \bullet Y + X' \bullet Z = (X + Z) \bullet (X' + Y)$$

A Few Fun Facts

That you're not responsible for:

The identities are divided into "axioms" and "theorems"

Mathematicians (and some computer scientists, like me ☺) will sometimes study what minimum starting points ("the axioms") will be enough to derive all the usual facts we rely on ("the theorems")

That's what I meant by "operations that work kinda like plus and times"

For our purposes, we won't make a distinction here, but we will use similar thinking later in the course.

Boolean algebra makes things like commutativity axioms (starting points, things we assume) with propositional logic, we start from the truth tables and can derive that commutativity is true. For this class, though, it's a fact you can use either way.

Why ANOTHER way of writing down logic?

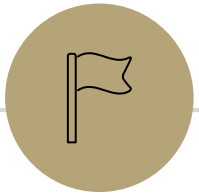
This is the third one!?

Because, in your future courses, you'll use any/all of them.

Remember there aren't new concepts here, just new representations.

We mostly use propositional notation ($\wedge$,$\vee$, $\neg$,etc.) but we'll use them all a bit so you're ready for any of them in your future courses.

Practice in section and on homework.



Canonical Forms

Back to the old notation.

Canonical Forms

A truth table is a unique representation of a Boolean Function.

If you describe a function, there's only one possible truth table for it.

Given a truth table you can find many circuits and many compound propositions to represent it.

Think back to when we were developing the law of implication…

It would be nice to have a "standard" proposition (or standard circuit) we could always write as a starting point.

So we have a (possibly) shorter way of telling if we have the same function.

Using Our Rules

WOW that was a lot of rules.

Why do we need them? Simplification!

Let's go back to the "law of implication" example.

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

When is the implication true? Just "or" each of the three "true" lines!

$$(p \wedge q) \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q)$$

Also seems pretty reasonable

So is $(p \wedge q) \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q) \equiv (\neg p \vee q)$

i.e. are these both alternative representations of $p \rightarrow q$?

Disjunctive Normal Form (DNF)

a.k.a. OR of ANDs

a.k.a Sum-of-Products Form

a.k.a. Minterm Expansion

1. Read the true rows of the truth table
2. AND together all the settings in a given (true) row.
3. OR together the true rows.

Disjunctive Normal Form

p	q	$G(p,q)$
T	T	T
T	F	F
F	T	T
F	F	F

$p \wedge q$

$\neg p \wedge q$

1. Read the true rows of the truth table
2. AND together all the settings in a given (true) row.
3. OR together the true rows.

$$G(p,q) \equiv (p \wedge q) \vee (\neg p \wedge q)$$

Another Canonical Form

DNF is a great way to represent functions that are usually false.

If there are only a few true rows, the representation is short.

What about functions that are usually true?

Well G is equivalent to $\neg\neg G$, and $\neg G$ is a function that is usually false.

Let's try taking the Sum-of-Products of $\neg G$ and negating it.

Another Canonical Form

p	q	$G(p,q)$	$\neg G(p,q)$	
T	T	T	F	
T	F	F	T	$p \wedge \neg q$
F	T	T	F	
F	F	F	T	$\neg p \wedge \neg q$

1. Read the true rows of the truth table
2. AND together all the settings in a given (true) row.
3. OR together the true rows.

$$\neg G(p,q) \equiv (p \wedge \neg q) \vee (\neg p \wedge \neg q)$$
$$G(p,q) \equiv \neg[(p \wedge \neg q) \vee (\neg p \wedge \neg q)]$$
$$G(p,q) \equiv [\neg(p \wedge \neg q) \wedge \neg(\neg p \wedge \neg q)]$$
$$G(p,q) \equiv [(\neg p \vee q) \wedge (p \vee q)]$$

This is not in Disjunctive Normal Form! It's something else, though…

Conjunctive Normal Form

a.k.a. AND of ORs

a.k.a. Product-of-Sums Form

a.k.a. Maxterm Expansion

1. Read the false rows of the truth table
2. OR together the negations of all the settings in the false rows.
3. AND together the false rows.

Or take the DNF of the negation of the function you care about, and distribute the negation.

Normal Forms

Don’t simplify any further! Don’t factor anything out (even if you can). The point of the canonical form is we know exactly what it looks like, you might simplify differently than someone else.

Why? Easier to understand for people.

Inside the parentheses are only ORs between the parentheses are only ANDs (or vice versa).

You’ll use these more in later courses.