Propositional Logic

Definition: A **proposition** is a sentence with a definite truth value.

Examples:

Sentence	**Declarative**	**True/False**	**Proposition?**
The GCU campus is in Phoenix.	Yes	T	Yes
2 + 3 = 5	Yes	T	Yes
2 + 3 = 6	Yes	F	Yes
Is GCU in New Mexico?	No	NA	No
Does 2 + 2 = 4?	No	NA	No
Do your homework!	No	NA	No
This sentence is false.	Yes	NA	No[1]

[1] If the sentence is true, then it must be false. If false it must be true. So it must be neither!

Propositional Variables

Definition: A **propositional variable** is a variable, usually, p, q, … These stand in place of propositions. The **value** of a propositional variable is either True (T) or False (F).

We use propositional variables to construct compound propositions with the logical connectives “or” (“$\lor$”), “and” (“$\land$”), “implies” (“$\rightarrow$”), and “not” (“$\neg$”).

Example: Let p = “You have COVID.” and q = “Your COVID test is positive.”

Compound Proposition	**Notation**	**Name**
You do not have COVID.	$\neg p$	negation
You have COVID or your COVID test is positive.	$p \lor q$	disjunction
You have COVID and your COVID test is positive.	$p \land q$	conjunction
You have COVID implies your COVID test is positive.	$p \rightarrow q$	conditional
You have COVID if and only if your COVID test is positive.	$p \leftrightarrow q$	bi-conditional

Variants

There are many variants that you must be aware of, for example, the following are all ways of expressing “$p \rightarrow q$”:

p implies q	p is sufficient for q	q provided that p
If p, then q	q is necessary for p	q whenever p
p only if q	q follows from p	q unless ¬p

Similar alternatives exist for the other connectives. Your homework will cover some of these.

Truth tables for the logical connectives

“**not** p” is true when p is false and false when p is true.

p	¬p
T	F
F	T

Truth tables for the logical connectives

“p **and** q” is true when p is true and q is true and otherwise false.

p	q	$p \land q$
T	T	T
T	F	F
F	T	F
F	F	F

Truth tables for the logical connectives

“p **or** q” is true when p is true or q is true or both. (**inclusive or**)

p	q	$p \lor q$
T	T	T
T	F	T
F	T	T
F	F	F

In natural languages, like English, context often determines if the current use of “or” should be *inclusive*. For example, “You might get back quarters **or** dimes for change.” You might very well get both quarters and dimes back, so this is an inclusive or.

Truth tables for the logical connectives

“p **xor** q” is true when p is true or q is true but not both. (**exclusive or**)

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

For example, “You may have fries or a salad with your meal.” This would usually be interpreted to mean “not both” and hence a use of exclusive or.

Truth tables for the logical connectives

"p **implies** q" is false only when the *hypothesis* p is true and the *conclusion* q is false.

p	q	$p \to q$
T	T	T
T	F	F
F	T	T
F	F	T

This can be a little strange at first when the hypothesis is false. Both of the following sentences are true, as in each case, the hypothesis is false:

- If the earth is flat, then the moon is round. ($F \to T$)
- If the earth is flat, then the moon is square. ($F \to F$)

More on conditionals

Conditionals are so important that they deserve some additional comments. Given a conditional $p \rightarrow q$ there are three related propositions

- (Contrapositive) $\neg q \rightarrow \neg p$
- (Converse) $q \rightarrow p$
- (Inverse) $\neg p \rightarrow \neg q$

The main thing to remember about these is that the contrapositive and the original conditional are *(logically) equivalent*[1], whereas the converse and inverse are different. This is easy to see in a truth table:

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

Assume below that a is a fixed integer:

▹If $a < 2$, then $a < 3$. (conditional - true for all a)
▹If $a \geq 3$, then $a \geq 2$. (contrapositive - true for all a)
▹If $a \geq 2$, then $a \geq 3$. (inverse - true for some a)
▹If $a < 3$, then $a < 2$. (converse - true for some a)

For which a are the third and fourth true?

[1]More on logical equivalence later.

Truth tables for the logical connectives

“p **if and only if** q” is true precisely when either both p and q are true or both p and q are false.

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

It is common to use the terminology “p is ***necessary and sufficient*** for q.”

Truth tables for compound propositions

p	q	$\neg p$	$\neg q$	$p \rightarrow q$	$\neg q \rightarrow \neg p$	$(p \rightarrow q) \leftrightarrow (\neg q \rightarrow \neg p)$
T	T	F	F	T	T	T
T	F	F	T	F	F	T
F	T	T	F	T	T	T
F	F	T	T	T	T	T

A truth table for a compound proposition with n propositional variables will have 2^n rows.

Definition: If the value of a compound proposition is always T, then the proposition is called a ***tautology.*** In the example above, $(p \rightarrow q) \leftrightarrow (\neg q \rightarrow \neg p)$ is a tautology. If the value is always F, then the proposition is a ***contradiction***. A proposition that is not a contradiction is ***satisfiable***. If a proposition is neither a tautology nor a contradiction, then it is called a ***contingency***.

Satisfiability

From the previous slide it is clear that checking if a proposition is satisfiable might require 2^n steps if there are n propositional variables. This is exponential in the size of the problem. Sometime filling out the entire table can be avoided. For example, show that $((p \land q) \to r) \land \neg((p \to r) \land (q \to r))$ is satisfiable:

p	q	r	$p \land q$	$(p \land q) \to r$	$p \to r$	$q \to r$	$(p \to r) \land (q \to r)$	$\neg((p \to r) \land (q \to r))$
		F	F	T			F	T

Satisfiability

From the previous slide it is clear that checking if a proposition is satisfiable might require 2^n steps if there are n propositional variables. This is exponential in the size of the problem. Sometime filling out the entire table can be avoided. For example, show that $((p \land q) \rightarrow r) \land \neg((p \rightarrow r) \land (q \rightarrow r))$ is satisfiable:

p	q	r	$p \land q$	$(p \land q) \rightarrow r$	$p \rightarrow r$	$q \rightarrow r$	$(p \rightarrow r) \land (q \rightarrow r)$	$\neg((p \rightarrow r) \land (q \rightarrow r))$
F	T	F	F	T	T	F	F	T

Note here that we made a choice. We could have made $p \rightarrow r$ false and $q \rightarrow r$ true just as well in order to make $(p \rightarrow r) \land (q \rightarrow r)$ false.

Satisfiability

From the previous slide it is clear that checking if a proposition is satisfiable might require 2^n steps if there are n propositional variables. This is exponential in the size of the problem. Sometime filling out the entire table can be avoided. For example, show that $((p \land q) \rightarrow r) \land \neg((p \rightarrow r) \land (q \rightarrow r))$ is satisfiable:

p	q	r	$p \land q$	$(p \land q) \rightarrow r$	$p \rightarrow r$	$q \rightarrow r$	$(p \rightarrow r) \land (q \rightarrow r)$	$\neg((p \rightarrow r) \land (q \rightarrow r))$
F	T	F	F	T	T	F	F	T

What about $\neg((p \land q) \rightarrow r) \land ((p \rightarrow r) \land (q \rightarrow r))$?

p	q	r	$p \land q$	$(p \land q) \rightarrow r$	$p \rightarrow r$	$q \rightarrow r$	$(p \rightarrow r) \land (q \rightarrow r)$	$\neg((p \rightarrow r) \land (q \rightarrow r))$
		F	T	F	T	T	T	

Satisfiability

From the previous slide it is clear that checking if a proposition is satisfiable might require 2^n steps if there are n propositional variables. This is exponential in the size of the problem. Sometime filling out the entire table can be avoided. For example, show that $((p \land q) \to r) \land \neg((p \to r) \land (q \to r))$ is satisfiable:

p	q	r	$p \land q$	$(p \land q) \to r$	$p \to r$	$q \to r$	$(p \to r) \land (q \to r)$	$\neg((p \to r) \land (q \to r))$
F	T	F	F	T	T	F	F	T

What about $\neg((p \land q) \to r) \land ((p \to r) \land (q \to r))$?

p	q	r	$p \land q$	$(p \land q) \to r$	$p \to r$	$q \to r$	$(p \to r) \land (q \to r)$	$\neg((p \to r) \land (q \to r))$
T	T	F	T	F	T	T	T	

Satisfiability

From the previous slide it is clear that checking if a proposition is satisfiable might require 2^n steps if there are n propositional variables. This is exponential in the size of the problem. Sometime filling out the entire table can be avoided. For example, show that $((p \land q) \to r) \land \neg((p \to r) \land (q \to r))$ is satisfiable:

p	q	r	$p \land q$	$(p \land q) \to r$	$p \to r$	$q \to r$	$(p \to r) \land (q \to r)$	$\neg((p \to r) \land (q \to r))$
F	T	F	F	T	T	F	F	T

What about $\neg((p \land q) \to r) \land ((p \to r) \land (q \to r))$?

p	q	r	$p \land q$	$(p \land q) \to r$	$p \to r$	$q \to r$	$(p \to r) \land (q \to r)$	$\neg((p \to r) \land (q \to r))$
T	T	F	T	F	T	T/F	T	

We see here that this proposition is not satisfiable. It is a **contradiction**.

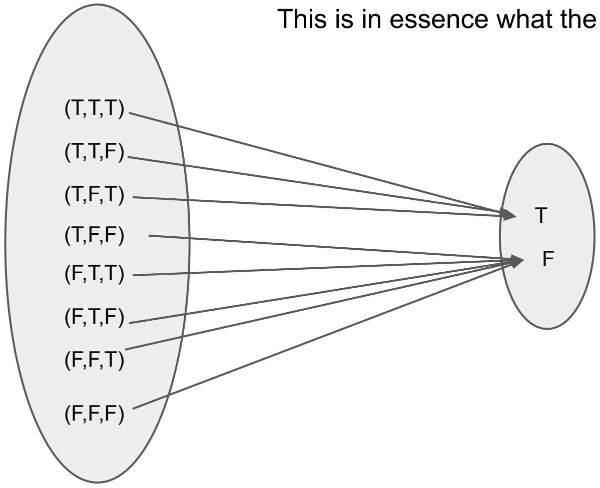
Abstraction

Any compound proposition with atomic propositional variables p_1, ..., p_n can be viewed as a function $f(p_1, \ldots, p_n)$ mapping from $\{T, F\}^n$ into $\{T, F\}$. This is precisely what a truth table visualizes. To be specific I will identify the truth table for a compound proposition with this function.

Example: Consider $p \wedge (q \vee r)$. A truth table might look like

p	q	r	$q \vee r$	$p \wedge (q \vee r)$
T	**T**	**T**	T	**T**
T	**T**	**F**	T	**T**
T	**F**	**T**	T	**T**
T	**F**	**F**	F	**F**
F	**T**	**T**	T	**F**
F	**T**	**F**	T	**F**
F	**F**	**T**	T	**F**
F	**F**	**F**	F	**F**

This is in essence what the truth table is.



Equivalence

Definition: Two compound propositions are ***equivalent*** if they have the same truth table.

Example:

p	q	r	$q \vee r$	$p \wedge (q \vee r)$	$p \wedge q$	$p \wedge r$	$(p \wedge q) \vee (p \wedge r)$
T	**T**	**T**	T	**T**	T	T	**T**
T	**T**	**F**	T	**T**	T	F	**T**
T	**F**	**T**	T	**T**	F	T	**T**
T	**F**	**F**	F	**F**	F	F	**F**
F	**T**	**T**	T	**F**	F	F	**F**
F	**T**	**F**	T	**F**	F	F	**F**
F	**F**	**T**	T	**F**	F	F	**F**
F	**F**	**F**	F	**F**	F	F	**F**

Here we see that $p \wedge (q \vee r)$ and $(p \wedge q) \vee (p \wedge r)$ have the same truth table (truth function) and hence are equivalent propositions.

It should be clear that two compound propositions:

$f(p,q,r)$ and $g(p, q, r)$ are equivalent
iff
$f(p,q,r) \leftrightarrow g(p,q,r)$ is a tautology.

Truth Table (Truth Function) ➪ Compound proposition

A natural question is "Can any truth function be represented as a compound proposition?" The answer is easy and "Yes, in two different ways."

Proof by example:

p	q	r	f(p,q,r)
T	T	T	T
T	T	F	T
T	F	T	T
T	F	F	F
F	T	T	F
F	T	F	F
F	F	T	F
F	F	F	F

Disjunctive normal form (**DNF**):

$(p \land q \land r) \lor (p \land q \land \neg r) \lor (p \land \neg q \land r)$

It is clear that this is T for the first three rows and false for the last five rows, so it has exactly this truth table.

Conjunctive normal form (**CNF**):

$(\neg p \lor q \lor r) \land (p \lor \neg q \lor \neg r) \land (p \lor \neg q \lor r) \land (p \lor q \lor \neg r) \land (p \lor q \lor r)$

CNF is a little harder to see. The formula says, "I am true if and only if none of TFF, FTT, FTF, FFT, or FFF occur." Equivalently, "I am true precisely when one of TTT, TTF, or TFT occur."

Using satisfiability to solve a logic puzzle

There are various sorts of logic puzzles in your text. One sort goes like this: There are treasures in two of the trunks of three trees. Each tree has an inscription, in order, “This trunk is empty,” “There is a treasure in trunk of the first tree,” and “There is a treasure in the trunk of the second tree.” Can the queen who never lies state the following, and if so which tree trunks contain the treasure?

1. All the inscriptions are false.
2. Exactly one inscription is true.
3. Exactly two inscriptions are true.
4. All three inscriptions are true.

A solution: Let p_i be the proposition “There is treasure in the trunk of the i^{th} tree.” Our given inscriptions are: $\neg p_1$, p_1, and p_2. Translating the queen’s sentences gives:

1. $p_1 \wedge \neg p_1 \wedge \neg p_2$
2. $(\neg \mathbf{p_1} \wedge \neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \mathbf{p_1} \wedge \neg p_2) \vee (p_1 \wedge \neg p_1 \wedge \mathbf{p_2})$
3. $(\neg \mathbf{p_1} \wedge \mathbf{p_1} \wedge \neg p_2) \vee (\neg \mathbf{p_1} \wedge \neg p_1 \wedge \mathbf{p_2}) \vee (p_1 \wedge \mathbf{p_1} \wedge \mathbf{p_2})$
4. $\neg \mathbf{p_1} \wedge \mathbf{p_1} \wedge \mathbf{p_2}$

1. F
2. $(\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2)$
3. $(\neg p_1 \wedge p_2) \vee (p_1 \wedge p_2)$
4. F

Puzzle continued

					1 inscription is true			2 inscriptions are true
p_1	p_2	p_3	$\neg p_1 \wedge \neg p_2$	$p_1 \wedge \neg p_2$	$(\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2)$	$p_1 \wedge p_2$	$\neg p_1 \wedge p_2$	$(\neg p_1 \wedge p_2) \vee (p_1 \wedge p_2)$
T	T	F	F	F	**F**	T	F	**T**
T	F	T	F	T	**T**	F	F	**F**
F	T	T	F	F	**F**	F	T	**T**

You only consider cases where treasure is in two of the three trunks.

The queen can speak “Exactly one inscription is true” when the treasure is in the trunk of the first and third trees. In this case the one true inscription is “There is a treasure in trunk of the first tree.”

There are two ways that the queen can speak “Exactly two inscriptions are true.” Namely the treasure can be in the first and second tree, so that the second and third inscriptions are true, or the treasure can be in the second and third tree so that the first and second inscriptions are true.

Puzzle alternate solution

It occurred after I wrote the first solution that there is another perhaps simpler method. The previous method is more general, but for this problem the following works.

S_2	S_3		s_1	s_i is the i^{th} inscription
p_1	p_2	p_3	$\neg p_1$	
T	**T**	F	F	Exactly two are true.
T	F	T	F	Exactly one is true.
F	**T**	T	**T**	Exactly two are true.

Again only consider cases where treasure is in two of the three trunks.

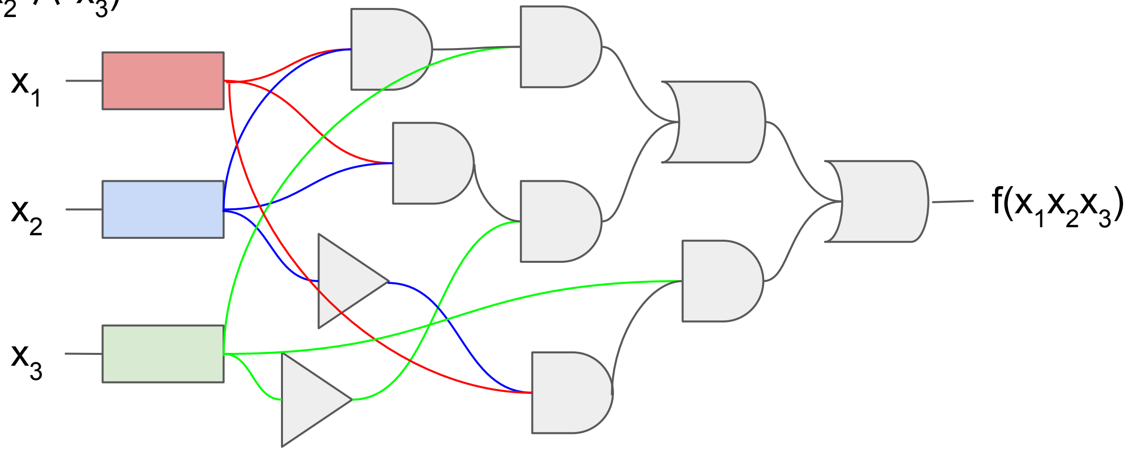
Logic Circuits

As shown above any truth function/table can be represented simply in DNF or CNF form. This same method makes it easy to find circuits that yield any desired boolean function.

Example: Find a circuit that generates the following boolean function. I am reading the binary input left-to-right so $x_1x_2x_3$ = 001

input	output
000	0
001	0
010	0
011	0
100	0
101	1
110	1
111	1

CNF: $(x_1 \wedge x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge \neg x_3) \vee (x_1 \wedge \neg x_2 \wedge x_3)$



Complete sets of connectives.

We have seen by using CNF and DNF that the sets of connectives $\{\wedge, \neg\}$ and $\{\vee, \neg\}$ are complete, in that any truth table (truth function) can be realized by a proposition built from just atomic propositions and those connectives. In terms of circuits, this means {AND, NOT} and {OR, NOT} gates can create any boolean function.

Definition: NAND (|) and **NOR(↓)**. Here $p \mid q \equiv \neg(p \wedge q)$ and $p \downarrow q \equiv \neg(p \vee q)$.

Claim: Both $\{\mid\}$ and $\{\downarrow\}$ are complete. To show $\{\mid\}$ is complete it suffices to show that both $\wedge$ and $\neg$ can be defined from $\mid$. Showing that $\downarrow$ is complete is similar. (Section 1.3, Exercises 54 and 56.)

p	q	p | q
T	T	F
T	F	T
F	T	T
F	F	T

p	p | p
T	F
F	T

$p \mid p \equiv \neg p$

p	q	p | q	(p | q) | (p | q)
T	T	F	T
T	F	T	F
F	T	T	F
F	F	T	F

$(p \mid q) \mid (p \mid q) \equiv p \wedge q$

XOR is not complete

Consider any propositions just built out of XOR, like $p \oplus p$, or $p \oplus q$, or $(p \oplus q) \oplus r$, etc. The truth table for each of these always has exactly ½ T's and ½ F's. This means that neither of $\wedge$ or $\vee$ can be written in terms of $\oplus$.

To see this notice that there are some simple algebraic facts:

- $p \oplus q = q \oplus p$ (commutative)
- $p \oplus (q \oplus r) = (p \oplus q) \oplus r$ (associative)
- $p \oplus F = p = F \oplus p$ (F is the $\oplus$-identity)
- $p \oplus p = F$ (idempotent)
- $p \oplus T = \neg p$

Consider a $\{\oplus\}$ proposition, for example, $(p_1 \oplus p_2) \oplus ((p_3 \oplus p_2) \oplus p_4)$. By associativity this can be rewritten unambiguously as $p_1 \oplus p_2 \oplus p_3 \oplus p_2 \oplus p_4$. Then by commutativity the proposition can be rewritten as $p_1 \oplus p_2 \oplus p_2 \oplus p_3 \oplus p_4$. By idempotency this is the same as $p_1 \oplus F \oplus p_3 \oplus p_4$ and by identity once more as $p_1 \oplus p_3 \oplus p_4$. This example can be turned into a proof that every such proposition can be written in the form $p_1 \oplus p_2 \oplus p_3 \cdots \oplus p_n$ or $p_1 \oplus p_2 \oplus p_3 \cdots \oplus p_n \oplus T$ where each p_i is distinct. Thus it suffices to look at truth tables for these propositions.

XOR is not complete

p	$p \oplus T$
T	F
F	T

p	q	$p \oplus q$	$p \oplus q \oplus T$
T	T	F	T
T	F	T	F
F	T	T	F
F	F	F	T

p	q	r	$p \oplus q$	$p \oplus q \oplus r$	$p \oplus q \oplus r \oplus T$
T	T	T	F	T	F
T	T	F	F	F	T
T	F	T	T	F	T
T	F	F	T	T	F
F	T	T	T	F	T
F	T	F	T	T	F
F	F	T	F	T	F
F	F	F	F	F	T

By induction $p_1 \oplus p_2 \oplus p_3 \cdots \oplus p_n$ has 2^{n-1} T's and same number of F's in it's truth table.