PART I. PROPOSITIONAL LOGIC

1. PROPOSITIONS AND FORMULAE

Propositions

A *proposition* (or *statement*) is a declarative sentence that asserts a potential fact.

A proposition can be true or false, which are called the *truth value* of this proposition. If a proposition is true, we say that the truth value of this proposition is *true*, denoted by T. If a proposition is false, we say that the *truth value* of this proposition is false, denoted by F.

The truth value of a proposition may depend on what happens in real world, or the truth values of some other propositions.

Examples

1. Today is January 11, 2021. (True if I say this on the right day.)

2. Today's highest temperature is 30°C. (False if today’s real highest temperature is 10°C.)

3. Today is January 11, 2021, and today's highest temperature is 30°C. (This statement is true or false depends on the truth values of propositions 1 and 2).

The following are not propositions:

1. Oh, my God! (This is an exclamation).

2. Is this a book? (This is a question).

3. Welcome to MAT 1348. (This is a greeting).

A proposition is a *conjecture* if its truth value is unknown.

Examples

1. Tomorrow will be a sunny day.

2. Every even number $N > 2$ is a sum of two prime numbers.

Formulae

Propositions that cannot be expressed in terms of simpler propositions are called *atomic propositions*.

A *compound proposition* is constructed from atomic propositions and/or other compound propositions using *logical operators* (also called *connectives*) $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$, where $\neg$ is a unary operator, and the others are binary operators.

We can use a letter to represent a proposition, called a (*propositional*) *variable*. We use **T** to represent a proposition that is always true, and use **F** to represent a proposition that is always false.

A (*well-formed propositional logic*) *formula* (plural: *formulae*) is defined recursively in the following way:

- a logical variable, **T**, and **F**, are formulae.

- if p is a formula, then $\neg p$ (*not p*) is a formula.

- if p and q are formulae, then $p \wedge q$ (*p and q*), $p \vee q$ (*p or q*), $p \rightarrow q$ (*p implies q*, or *if p then q*), and $p \leftrightarrow q$ (*p if and only if q*), are formulae.

$\neg p$ is called the *negation* of p;

$p \wedge q$ is called the *conjunction* of p and q;

$p \vee q$ is called the *disjunction* of p and q;

$p \rightarrow q$ is called a *conditional statement* or an *implication*, where p is called the *hypothesis*, and q is called the *conclusion*.

$p \leftrightarrow q$ is called a *bi-conditional statement* or a *bi-implication*.

The precedence relation among the connectives from the highest to the lowest is the following:

$$\neg, \wedge, \vee, \rightarrow, \leftrightarrow.$$

For example, $p \vee q \leftrightarrow \neg r \vee s \wedge t \rightarrow u$ means

$$(p \vee q) \leftrightarrow (((\neg r) \vee (s \wedge t)) \rightarrow u).$$

To avoid confusion, we would sometimes add brackets to specify the order of operations even though it is not necessary.

An assignment of a truth value to each variable involved in a formula is an *evaluation* of this formula.

If the formula is true with an evaluation, then this evaluation is a *true-evaluation* of this formula. If the formula is false with an evaluation, then this evaluation is a *false-evaluation* of this formula.

The truth value of a formula for each possible evaluation can be found by a *truth table*.

The basic truth tables of connectives are given as follows:

p	$\neg p$
T	F
F	T

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

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

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

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

Note that a conditional statement $p \to q$ is always true when p is false.

Example. If a classroom is empty, then the statement "all students in this room are female" is true.

This statement can be expressed as a conditional statement as "if a student is in this room, then this student is a female". As the classroom is empty, the hypothesis is false. Hence, this statement is true.

The truth table of more complicated formulae are constructed step by step as in the following example.

Example. The following is the truth table of the formula $(p \to q) \wedge (\neg q \vee r)$.

p	q	r	$p \to q$	$\neg q$	$\neg q \vee r$	$(p \to q) \wedge (\neg q \vee r)$
T	T	T	T	F	T	T

T	T	F	T	F	F	F
T	F	T	F	T	T	F
T	F	F	F	T	T	F
F	T	T	T	F	T	T
F	T	F	T	F	F	F
F	F	T	T	T	T	T
F	F	F	T	T	T	T

Tautologies and Contradictions

If the truth value of a formula is true for all possible evaluations, this formula is a *tautology*. If the truth value of a formula is false for all possible evaluations, then this formula is a *contradiction*.

A formula that is neither a tautology nor a contradiction is a *contingency*.

Example. Show that $(p \to q) \wedge (p \wedge \neg q)$ is a contradiction.

Construct the truth table of this formula as follows:

p	q	$p \to q$	$p \wedge \neg q$	$(p \to q) \wedge (p \wedge \neg q)$
T	T	T	F	F
T	F	F	T	F
F	T	T	F	F
F	F	T	F	F

The last column is always F. This formula is a contradiction.

2. Logical Equivalence of Formulae

Equivalence of Formulae

Two formulae p and q are (*logically*) *equivalent*, dented by $p \equiv q$, if they have the same truth value for all possible evaluations. In other words, $p \equiv q$ if and only if $p \leftrightarrow q$ is a tautology.

Because all tautologies are equivalent, and all contradictions are equivalent, we use **T** to represent any tautology, and use **F** to represent any contradiction.

Basic Equivalence Properties of Formulae

Identity Laws

$p \wedge \mathbf{T} \equiv p$,
$p \vee \mathbf{F} \equiv p$.

Domination Laws

$p \vee \mathbf{T} \equiv \mathbf{T}$,
$p \wedge \mathbf{F} \equiv \mathbf{F}$.

Negation Laws

$p \wedge \neg p \equiv \mathbf{F}$,
$p \vee \neg p \equiv \mathbf{T}$.

Double Negation Law:

$\neg\neg p \equiv p$.

De Morgan's laws:

$\neg(p \lor q) \equiv \neg p \land \neg q.$
$\neg(p \land q) \equiv \neg p \lor \neg q.$

Commutative Laws:

$p \lor q \equiv q \lor p,$
$p \land q \equiv q \land p.$

Associative Laws:

$p \lor (q \lor r) \equiv (p \lor q) \lor r,$
$p \land (q \land r) \equiv (p \land q) \land r.$

Idempotent Laws

$p \lor p \equiv p,$
$p \land p \equiv p.$

Distributive laws

$p \lor (q \land r) \equiv (p \lor q) \land (p \lor r),$
$p \land (q \lor r) \equiv (p \land q) \lor (p \land r).$

Absorption Laws

$p \lor (p \land q) \equiv p,$

$p \wedge (p \vee q) \equiv p$.

Equivalence Relation Involving Conditional Statements

$p \rightarrow q \equiv \neg p \vee q$.

Equivalence Relation Involving Bi-Conditional Statements

$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$.

All these equivalence properties can be verified by truth tables.

Example. Use a truth table to verify the distributive law

$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$.

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

Since last two columns are identical, these two formulae are equivalent.

Basic equivalence rules can be used to prove other equivalence relations of formulae.

Example. Prove the following equivalence relation:

$(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r.$

Proof. $(p \to r) \vee (q \to r) \equiv (\neg p \vee r) \vee (\neg q \vee r)$
$\equiv (\neg p \vee \neg q) \vee (r \vee r) \equiv \neg(p \wedge q) \vee r \equiv (p \wedge q) \to r.$

Use equivalence relations, we can write the formula in an equivalent form with specified connective.

Example. Write the formula $p \to (q \vee r)$ in an equivalent formula that uses only connectives $\neg$ and $\wedge$.

Solution.

$p \to (q \vee r) \equiv \neg p \vee (q \vee r) \equiv \neg(p \wedge \neg(q \vee r)) \equiv \neg(p \wedge \neg q \wedge \neg r).$

The Converse, Inverse, and Contraposition of a Conditional Statement

Let P: $p \to q$ be a conditional statement. The formula $q \to p$ is called the *converse* of P; the formula $\neg p \to \neg q$ is called the *inverse* of P, and $\neg q \to \neg p$ is called the *contraposition* of P.

The following truth table shows that a conditional statement is equivalent to its contraposition, but it is not equivalent to its converse or inverse.

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

The contraposition of $\neg q \to \neg p$ is $\neg\neg p \to \neg\neg q$ or $p \to q$. In other words, $p \to q$ and $\neg q \to \neg p$ are contrapositions of each other.

The contraposition of $q \to p$ is $\neg p \to \neg q$. In other words, the converse and the inverse of a conditional statement are contrapositions of each other, and, consequently, they are equivalent.

3. DNF AND CNF

Literals

A *literal* is a variable or the negation of a variable, or **T**, or **F**.

Example. If p, q, r are variables, then formulae $p, q, r, \neg p, \neg q, \neg r$, **T**, and **F**, are literals.

Disjunctive Normal Forms

A *conjunction clause* is a formula that uses only the conjunction connective $\wedge$ to join the literals such that every variable appears only once in its un-negated form or negated form, but not both.

A conjunction clause is true if and only if all literals in this clause are true.

Example. If p, q, and r, are variables, then $\neg p$, $p \wedge \neg q$, and $\neg p \wedge q \wedge \neg r$, are conjunction clauses, while $p \wedge \neg q \wedge p$, $p \wedge \neg q \wedge \neg p$, and $p \wedge \neg q \vee r$, are not conjunction clauses.

A formula is said to be in the *Disjunctive Normal Form* (DNF), if it is a disjunction of a number of conjunctive clauses.

Example. $p \wedge \neg q \wedge r$ and $\neg p \vee (p \wedge \neg q) \vee (\neg p \wedge q \wedge \neg r)$ are in DNF.

Formulae of the form $p \wedge \neg p$ are NOT conjunction clauses, but it is allowed in a DNF. This clause is always false denoted by **F**.

A DNF is true if and only if at least one of the conjunction clauses in this formula is true.

A conjunction clause in a DNF that contains all variables in consideration is *complete*.

If all conjunction clauses in a DNF formula are complete, then this DNF is *complete*.

Example. If only variables in consideration are p, q and r, then DNF $(p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge r)$ is complete, while DNF $(p \wedge q) \vee (\neg q \wedge \neg r) \vee (\neg p \wedge r)$ is not complete.

A complete conjunction clause is true if and only if all the literals are true. In other words, every complete conjunction clause has exactly one true-evaluation. Different complete conjunction clauses have different true-evaluations. On the other hand, for every evaluation E, there is a unique complete conjunction clause that has E as the only true-evaluation.

Example. $p \wedge q \wedge \neg r$ is true only when p is true, q is true, and r is false. $p \wedge \neg q \wedge \neg r$ is true only when p is true, q is false, and r is false.

The conjunction clause that is true only if p is false, q is false, and r is true, is $\neg p \wedge \neg q \wedge r$.

If a set of evaluations $S = \{E_1, E_2, \ldots, E_n\}$ is given, we can construct a complete DNF such that it is true if and only if the evaluation is in S.

For each of the given evaluations E_k, construct a complete conjunction clause C_k such that E_k is the only true-evaluation of C_k. Then the DNF $C_1 \vee C_2 \vee \ldots \vee C_n$ is true if and only if the evaluation is in S.

Let p and q be two conjunction clauses. Then $p \vee (p \wedge q) \equiv p$, we say that p *absorbs* $p \wedge q$.

If, in a DNF formula, two conjunction clauses have the form $p \wedge q$ and $p \wedge \neg q$, where p is a conjunction clause and q is a variable, then $(p \wedge q) \vee (p \wedge \neg q) \equiv p \wedge (q \vee \neg q) = p \wedge \mathbf{T} \equiv p$. This is called *coalescing*.

Absorption and coalescing can be used to simplify DNF formulas.

Example. Construct a formula involves variables p, q, and r, and it is true if and only if the evaluation is in set $S = \{(\text{T, T, F}), (\text{T, F, F}), (\text{F, T, T})\}$.

Solution. This formula is $(p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge r)$.

Absorption and coalescing can be used to simplify DNF formulas.

This formula can be simplified using absorption and/or coalescing:

$$
\begin{aligned}
&(p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \\
&\quad \equiv ((p \wedge \neg r) \wedge (q \vee \neg q)) \vee (\neg p \wedge q \wedge r) \\
&\quad \equiv ((p \wedge \neg r) \wedge \mathbf{T}) \vee (\neg p \wedge q \wedge r) \\
&\quad \equiv (p \wedge \neg r) \vee (\neg p \wedge q \wedge r).
\end{aligned}
$$

On the other hand, we can go backwards to convert a general DNF to a complete DNF to find all true-evaluations of this formula.

Example. Consider formula $(p \wedge \neg q) \vee (\neg p \vee q) \wedge r)$.

Convert this formula to a complete DNF:

$$
\begin{aligned}
&(p \wedge \neg q) \vee ((\neg p \vee q) \wedge r) \equiv (p \wedge \neg q) \vee (\neg p \wedge r) \vee (q \wedge r) \\
&\equiv (p \wedge \neg q \wedge (r \vee \neg r)) \vee (\neg p \wedge (q \vee \neg q) \wedge r) \vee ((p \vee \neg p) \wedge q \wedge r) \\
&\equiv (p \wedge \neg q \wedge r) \vee (p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge \neg q \wedge r) \\
&\vee (p \wedge q \wedge r) \vee (\neg p \wedge q \wedge r) \\
&\equiv (p \wedge \neg q \wedge r) \vee (p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge \neg q \wedge r) \\
&\vee (p \wedge q \wedge r).
\end{aligned}
$$

This formula is true if and only if the evaluation of (p, q, r) is (T, F, T), (T, F, F), (F, T, T), (F, F, T), or (T, T, T).

Conjunctive Normal Forms

A *disjunction clause* is a formula that uses only the disjunction connective $\vee$ to join the literals such that every atomic formula appears only once in its un-negated form or negated form, but not both.

A disjunction clause is false if all literals are false.

A formula is in the *Conjunctive Normal Form* (CNF), if it is a conjunction of a number of disjunctive clauses.

Formulae of the form $p \vee \neg p$ are NOT disjunction clauses, but it is allowed in a CNF. This clause is always true denoted by **T**.

A formula in CNF is false if and only if at least one of the disjunction clauses in this formula is false.

A disjunction clause in a formula in CNF that contains all variables in consideration is *complete*.

If all disjunction clauses in a CNF are complete, then this CNF is *complete*.

Every disjunction clause has a unique false-evaluation. Different disjunction clauses have different false-evaluations. On the other hand, for every evaluation E, there is a unique complete disjunction clause that has E as the only false-evaluation.

If a set of evaluations $S = \{E_1, E_2, \ldots, E_n\}$ is given, we can construct a complete CNF that is false if and only if the evaluation is in S.

For each of the given evaluations E_k, construct a complete disjunction clause D_k such that E_k is the only false-evaluation of D_k. Then the CNF $D_1 \wedge D_2 \wedge \ldots \wedge D_n$ is false if and only if the evaluation is in S.

Let p and q be two disjunction clauses. Then $p \wedge (p \vee q) \equiv p$, we say that *$p$ absorbs $p \vee q$*.

If, in a CNF formula, two disjunction clauses have the form $p \vee q$ and $p \vee \neg q$, where p is a disjunction clause and q is a literal, then

$$(p \vee q) \wedge (p \vee \neg q) \equiv p \vee (q \wedge \neg q) \equiv p \vee \mathbf{F} \equiv p.$$

This is called *coalescing*.

Absorption and coalescing are used to simplify CNF formulas.

Example. Construct a formula involves variables p, q, and r, and it is false if and only if the truth values of p, q, and r, are (T, F, F), (F, T, T), or (F, F, T).

Solution. This formula is $(\neg p \vee q \vee r) \wedge (p \vee \neg q \vee \neg r) \wedge (p \vee q \vee \neg r)$.

This formula can be simplified using absorption and/or coalescing.

$$\begin{aligned}
&(\neg p \vee q \vee r) \wedge (p \vee \neg q \vee \neg r) \wedge (p \vee q \vee \neg r) \\
&\equiv (\neg p \vee q \vee r) \wedge ((p \vee \neg r) \vee (\neg q \wedge q)) \\
&\equiv (\neg p \vee q \vee r) \wedge ((p \vee \neg r) \vee \mathbf{F}) \\
&\equiv (\neg p \vee q \vee r) \wedge (p \vee \neg r).
\end{aligned}$$

On the other hand, we can go backwards to convert a general CNF to a completer CNF to find all false-evaluations of this formula.

Example. Consider formula $(p \to q) \wedge (\neg p \to r)$.

Convert it to a complete CNF:

$$(p \to q) \wedge (\neg p \to r) \equiv (\neg p \vee q) \wedge (\neg\neg p \vee r) \equiv (\neg p \vee q) \wedge (p \vee r)$$
$$\equiv (\neg p \vee q \vee (r \wedge \neg r)) \wedge (p \vee (q \wedge \neg q) \vee r)$$
$$\equiv (\neg p \vee q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (p \vee q \vee r) \wedge (p \vee \neg q \vee r).$$

This formula is false if the evaluation of (p, q, r) is (T, F, F), (T, F, T), (F, F, F), or (F, T, F).

4. Translation

There is a number of ways to translate an English sentence into a logical formula. Different English phrases may be translated into the same connective.

The following are some ways to say $\neg p$:

Not p.
p is not true.
p is false.
p does not happen.
p fails.

The following are some ways to say $p \wedge q$:

p and q.
p but q.
p although q.
p, however q.
p, whereas q.
p besides q.
p, nevertheless q.
p even though q.
not only p, but also q.
p, while q.
p as well as q.

The following are some ways to say $p \vee q$:

p or q.
either p or q.
p or else, q.
p or, alternatively, q.
p otherwise q.
p unless q.

Here is an explanation of why "p unless q" is translated into $p \vee q$:

Phrase "p unless q" means "without q, we must have p", or, equivalently, "if q is false, then p must be true". In the form of a formula, we have $\neg q \to p$. By the equivalence rule, $\neg q \to p \equiv \neg\neg q \vee p \equiv q \vee p \equiv p \vee q$.

The following are some ways to say $p \to q$:

If p, then q.
Given that p, (it follows that) q.
In the case that p, q.
p leads to q.
p only if q.
Provide that p, q.
So long as p, q.
When p, q.
p is a sufficient condition for q.
q is a necessary condition for p.
p implies q.

Here is an explanation of why "p only if q" is translated into $p \rightarrow q$:

The sentence "p only if q" means that if q is false, then p must be false. This is translated to $\neg q \rightarrow \neg p$. We know that this is the contraposition of, hence equivalent to, $p \rightarrow q$.

The following are some ways to say $p \leftrightarrow q$:

p if and only if q.
p exactly in the case that q.
p just in the case that q.
p is equivalent to q.
p is a necessary and sufficient condition for q.

Examples

Every inhabitant on the Island of Knights and Knaves is either a *knight*, who always tells the truth, or a *knave*, who always lies.

Let A, B, and C be inhabitant on this island. Define the following formulae:

a: A is a knight.
b: B is a knight.
c: C is a knight.

The following are some translations of sentences using a, b and c as variables:

If A is a knave, then B is a knight.

$\neg a \rightarrow b$.

A is a knight, but B is a knave.

$a \wedge \neg b$.

A is a knight if B is a knave.

$\neg b \rightarrow a$.

A is a knight only if B is a knave.

$b \rightarrow \neg a$, or

$a \rightarrow \neg b$.

"A is a knight" is a necessary condition for B to be a knave.

$\neg a \rightarrow b$, or

$\neg b \rightarrow a$.

Even though A is a knight, B or C is a knave.

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

When *A* is a knight, if *B* is a knave, then *C* is a knight.

$a \to (\neg b \to c)$.

It is not true that, if *A* is a knight, then at least one of *B* and *C* must also be a knight.

$\neg(a \to (b \vee c))$.

Even if both *A* and *B* are knaves, *C* does not have to be a knave.

Rewrite the sentence: It is not true that, if both *A* and *B* are knaves, *C* has to be a knave.

$\neg((\neg a \wedge \neg b) \to \neg c)$.

Exactly one of *A*, *B*, and *C* is a knave.

Rewrite the sentence: *A* and *B* are knights and *C* is a knave, or *A* and *C* are knights and *B* is a knave, or *B* and *C* are knights and *A* is a knave.

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

5. Consistency of Formulae

Consistency of a Set of Formulae

A set of formulae $\{P_1, P_2, \ldots, P_n\}$ is *consistent* if there is an evaluation of involved variables such that all the formulae in this set are true. Otherwise, this set is *inconsistent*.

If the set S of formulae is inconsistent, then any set that contains S is inconsistent.

If a set consists of a single formula, it is inconsistent if and only if this formula is a contradiction.

Examples

1. The set $\{(p \vee q) \rightarrow (q \wedge r), p \wedge (q \rightarrow r), p \wedge r\}$ is consistent.

Indeed, when p, q, and r, are all true, all formulae in this set are true.

2. The set of formulae

$\{$(i) $(p \vee q) \rightarrow (q \wedge r)$, (ii) $\neg r \rightarrow p$, (iii) $(p \rightarrow q) \wedge \neg r\}$ is inconsistent.

Indeed, if (iii) is true, then $\neg r$ is true, and r is false. From (ii), since $\neg r$ is true, p is true. Then, from (i), since $p \vee q$ is true, we must have $q \wedge r$ is true. Then r is true, and we have a contradiction.

We can use an argument as in previous examples to determine the consistency of a set of formulae.

Another way to verify the consistency of a set of formulae is to use a truth table. Construct the truth table of each formula in this set to see if there is an evaluation that makes all formulae in this set true.

Example. Consider a set of formulae $\{\neg p \wedge q, p \rightarrow q, p \vee \neg q\}$.

Construct a truth table as follows:

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

Because there is no evaluation that makes all formulae true, this set is inconsistent.

The argument method is tricky, and the truth table method is tedious. The *Truth Tree* method is another way used to check the consistency of a set of formulae.

Truth Tree Method

The truth tree method decomposes formulae according to different connectives into literals. Two formulae in the same branch indicate conjunction, and two formulae in different branch indicate disjunction.

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

conjunction

$p \wedge q$
p
q

disjunction

$p \vee q$
/ \
p q

implication

$p \rightarrow q$
/ \
$\neg p$ q

negate disjunction

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

negate conjunction

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

negate implication

$\neg(p \rightarrow q)$
p
$\neg q$

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

double negation	biconditional		negate biconditional	
$\neg\neg p$	$p \leftrightarrow q$		$\neg(p \leftrightarrow q)$	
	p	$\neg p$	p	$\neg p$
p	q	$\neg q$	$\neg q$	q

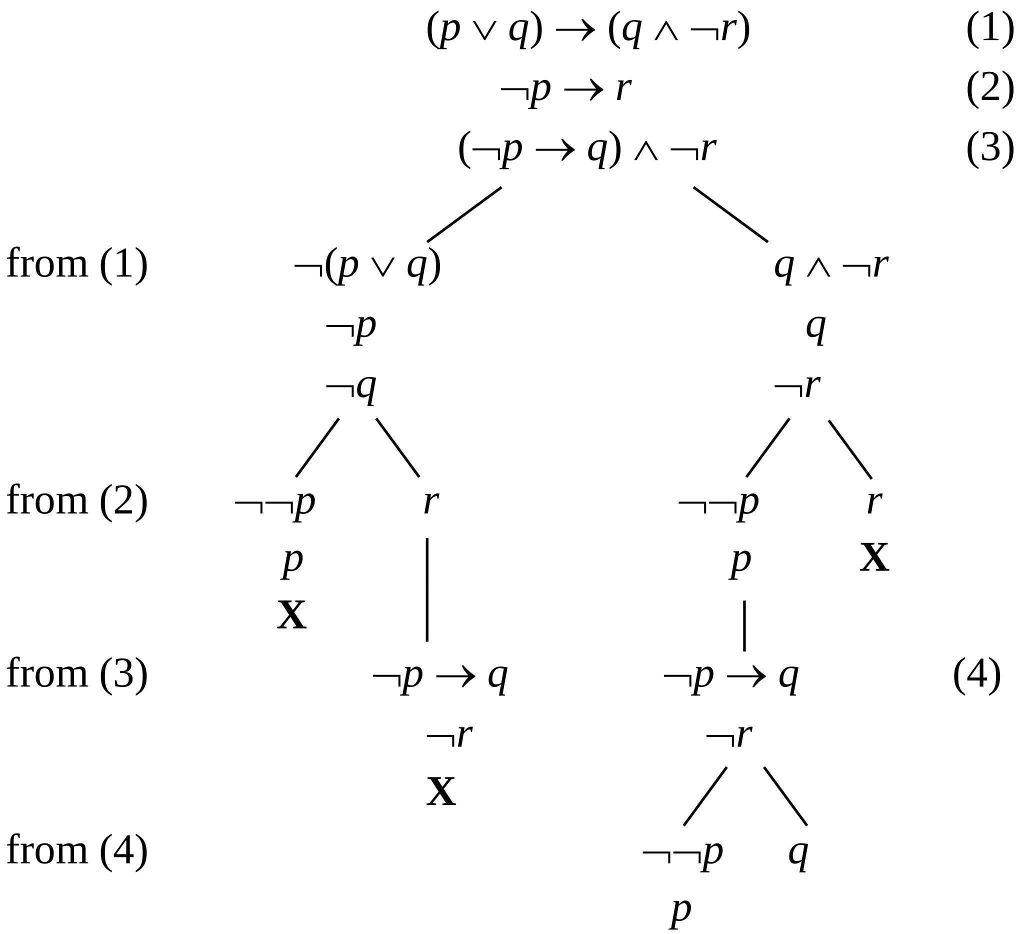
During the construction of a truth tree, if a branch has both an atomic formula and its negation, this branch is *closed*, i.e., it will not grow any further. Use a cross X to indicate a closed branch.

When all formulae are decomposed into literals, every open branch gives an evaluation that all the formulae in this set are true. If all branches are closed, this set is inconsistent.

Examples

1. Determine whether the set of formulae $P = \{(p \vee q) \rightarrow (q \wedge \neg r), \neg p \rightarrow r, (\neg p \rightarrow q) \wedge \neg r\}$ is consistent. If it is consistent, find all true-evaluations.

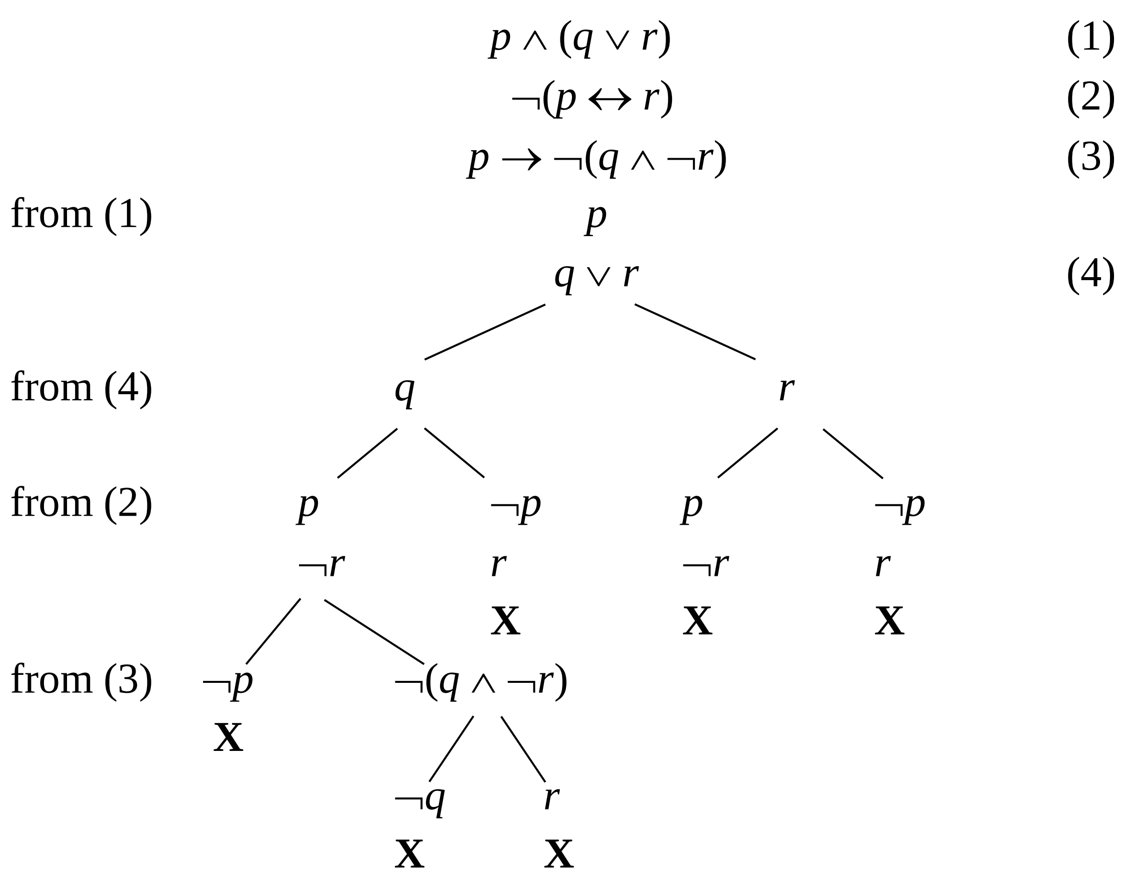
Construct a truth tree of the set as follows:



There are two open branches that give the same evaluation: p: true, q true, and r: false.

2. Show that the set of formulae $P = \{p \wedge (q \vee r), \neg(p \leftrightarrow r), p \rightarrow \neg(q \wedge \neg r)\}$ is inconsistent.

Construct the truth tree:



Since all branches are closed, this set of formulae is inconsistent.