

Propositional and Predicate Logic

CS 536 Lecture 2, Wed Jan 11, 2012

A. Why

- Reviewing/overviewing logic is necessary because we'll be using it in the course.
- We'll be using predicates to write specifications for programs.
- Predicates and programs have meaning relative to states.

B. Outcomes

At the end of today, you should

- Understand what a propositional formula is, how to write them, how to tell whether one is a tautology or contradiction using truth tables, and see a basic set of logical rules for transforming propositions.

C. In Case You Missed Class Monday

- The course webpages are at <http://www.cs.iit.edu/~cs536> . Read the home page and syllabus carefully. Questions: What is the homework collaboration policy? What is the second chance policy for tests?
- Review the material: Propositions are built up from proposition variables, proposition constants, and the connectives \wedge , \vee , \rightarrow , \leftrightarrow , and \neg . Can add/remove parentheses using precedence and associativity rules. Truth table semantics; tautologies, contradictions, contingencies.

D. Propositional Logic

- Last time: proposition variables, the connectives \wedge , \vee , \rightarrow , \leftrightarrow , and \neg . Their precedences and associativities. Truth table semantics.

E. Truth vs Provability

- In addition to semantic truth based on truth tables, there is also a notion of "provable truth" based on syntactic manipulation of propositions.
 - E.g., "if $p \wedge q$ is provable then $q \wedge p$ is provable" or " $q \wedge p$ follows from $p \wedge q$ " or " $p \wedge q$ implies $q \wedge p$ " [different notion of "implies" here].
- Given a set of proof rules, two propositions are **provably equivalent** if each follows from the other according to those rules. E.g., $p \wedge q$ and $q \wedge p$ are provably equivalent.
- Two propositions are **logically equivalent** if they have the same truth table. E.g., $p \wedge q$ and $q \wedge p$ are logically equivalent. Note two propositions p_1 and p_2 are logically equivalent iff $(p_1 \leftrightarrow p_2)$ is a tautology.
- Proofs of propositional equivalence often form a chain: p_1 is equivalent to p_2 , which is equivalent to p_3 etc.
 - But " p_1, p_2 , and p_3 are logically equivalent" is different from $p_1 \leftrightarrow (p_2 \leftrightarrow p_3)$.

- To indicate logical equivalence, we often use a different symbol from \leftrightarrow .
 - " \Leftrightarrow " and "iff" are common.
 - $p_1 \Leftrightarrow p_2 \Leftrightarrow p_3$ means p_1 is equivalent to p_2 and p_2 is equivalent to p_3 (and therefore p_1 is equivalent to p_3).

Similar to logical equivalence, there is a notion of logical implication that is often chained together, and this is different from \rightarrow .

- Write " \Rightarrow " or "then" or "implies".
- $p_1 \Rightarrow p_2 \Rightarrow p_3$ means p_1 implies p_2 and p_2 implies p_3 (and therefore p_1 implies p_3).

F. Some Basic Logical Equivalences/Implications

- **Commutativity**
 - $p \vee q \Leftrightarrow q \vee p$
 - $p \wedge q \Leftrightarrow q \wedge p$
 - $(p \Leftrightarrow q) \Leftrightarrow (q \Leftrightarrow p)$
- **Associativity**
 - $(p \vee q) \vee r \Leftrightarrow p \vee (q \vee r)$
 - $(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$
- **Distributivity**
 - $(p \vee q) \wedge r \Leftrightarrow (p \wedge r) \vee (q \wedge r)$
 - $(p \wedge q) \vee r \Leftrightarrow (p \vee r) \wedge (q \vee r)$
- **Transitivity [Note: \Rightarrow , not \Leftrightarrow here]**
 - $(p \rightarrow q) \wedge (q \rightarrow r) \Rightarrow (p \rightarrow r)$
 - $(p \Leftrightarrow q) \wedge (q \Leftrightarrow r) \Rightarrow (p \Leftrightarrow r)$
- **Identity**
 - $p \wedge T \Leftrightarrow p$
 - $p \vee F \Leftrightarrow p$
- **Domination**
 - $p \vee T \Leftrightarrow T$
 - $p \wedge F \Leftrightarrow F$
- **Contradiction**
 - $p \wedge \neg p \Leftrightarrow F$
- **Excluded middle**
 - $p \vee \neg p \Leftrightarrow T$
- **Double negation (Pierce's Law)**
 - $\neg\neg p \Leftrightarrow p$
- **Idempotency**
 - $p \vee p \Leftrightarrow p$
 - $p \wedge p \Leftrightarrow p$
- **Absurdity**
 - $(F \rightarrow p) \Leftrightarrow T$
- **DeMorgan's Laws**
 - $\neg(p \wedge q) \Leftrightarrow (\neg p \vee \neg q)$
 - $\neg(p \vee q) \Leftrightarrow (\neg p \wedge \neg q)$
- **Definition of \rightarrow and \Leftrightarrow**
 - $(p \rightarrow q) \Leftrightarrow (\neg p \vee q)$
 - $(p \Leftrightarrow q) \Leftrightarrow (p \rightarrow q) \wedge (q \rightarrow p)$
- **Substitution**
 - Say $p \Leftrightarrow q$ and r' is the result of substituting q for occurrences of p in r , then $r \Leftrightarrow r'$.
 - E.g.: Since $(p \rightarrow q) \Leftrightarrow (\neg p \vee q)$, we know $p \wedge (p \rightarrow q) \Leftrightarrow p \wedge (\neg p \vee q)$.

G. Sample Proofs

- Here's a proof of $\neg(p \rightarrow q) \Leftrightarrow (p \wedge \neg q)$ (also known as "negation of \rightarrow ").

$$\begin{aligned}
 & \neg(p \rightarrow q) \\
 \Leftrightarrow & \neg(\neg p \vee q) && \text{Defn } \rightarrow \\
 \Leftrightarrow & \neg\neg p \wedge \neg q && \text{DeMorgan's Law} \\
 \Leftrightarrow & p \wedge \neg q && \text{Pierce's Law}
 \end{aligned}$$

- Here is a proof of $(r \rightarrow s) \wedge r \rightarrow s \Leftrightarrow T$ ("Modus ponens").

$$\begin{aligned}
 & (r \rightarrow s) \wedge r \rightarrow s \\
 \Leftrightarrow & \neg((r \rightarrow s) \wedge r) \vee s && \text{Defn of } \rightarrow \\
 \Leftrightarrow & (\neg(r \rightarrow s) \vee \neg r) \vee s && \text{DeMorgan's Law} \\
 \Leftrightarrow & ((r \wedge \neg s) \vee \neg r) \vee s && \text{Negation of } \rightarrow \\
 \Leftrightarrow & ((r \vee \neg r) \wedge (\neg s \vee \neg r)) \vee s && \text{Distribute } \vee \text{ over } \wedge \\
 \Leftrightarrow & (T \wedge (\neg s \vee \neg r)) \vee s && \text{Excluded middle} \\
 \Leftrightarrow & (\neg s \vee \neg r) \vee s && \text{Identity} \\
 \Leftrightarrow & (\neg s \vee s) \vee \neg r && \vee \text{ commutative and associative} \\
 \Leftrightarrow & T \vee \neg r && \text{Excluded middle} \\
 \Leftrightarrow & T && \text{Domination}
 \end{aligned}$$

- In general, for propositions, $p \rightarrow q$ is a tautology (i.e., $(p \rightarrow q) \Leftrightarrow T$) exactly whenever we can prove $p \Rightarrow q$. Similarly, $(p \Leftrightarrow q) \Leftrightarrow T$ exactly when we can prove $p \Leftrightarrow q$.

H. Derived Rules

- If $p \rightarrow q$ is a tautology (i.e., we can prove $p \rightarrow q \Leftrightarrow T$), then we can use $p \Rightarrow q$ as a **derived rule**. E.g., to prove modus ponens, we showed $(r \rightarrow s) \wedge r \rightarrow s \Leftrightarrow T$. Here's an example of using modus ponens:

$$\begin{aligned}
 & ((p \wedge q) \rightarrow r) \wedge (p \wedge q) \\
 \Rightarrow & r && \text{Modus ponens}
 \end{aligned}$$

- The "and-elimination" rule is $p \wedge q \Rightarrow p$. To use it, we need to prove that $p \wedge q \rightarrow p$ is a tautology:

$$\begin{aligned}
 & p \wedge q \rightarrow p \\
 \Leftrightarrow & \neg(p \wedge q) \vee p && \text{Defn } \rightarrow \\
 \Leftrightarrow & (\neg p \vee \neg q) \vee p && \text{DeMorgan's Law} \\
 \Leftrightarrow & (p \vee \neg p) \vee \neg q && \vee \text{ associative and commutative} \\
 \Leftrightarrow & T \vee \neg q && \text{Excluded middle} \\
 \Leftrightarrow & T && \text{Domination}
 \end{aligned}$$

- Similarly if $p \Leftrightarrow q$ is a tautology (i.e., we can prove $p \Leftrightarrow q \Leftrightarrow T$), then we can use $p \Leftrightarrow q$ as a derived rule. E.g., the rule of contraposition for \Leftrightarrow is $(p \Leftrightarrow q) \Leftrightarrow (\neg p \Leftrightarrow \neg q)$. To use this we have to prove that $(p \Leftrightarrow q) \rightarrow (\neg p \Leftrightarrow \neg q) \Leftrightarrow T$ (proof is omitted here).

- Some other common derived rules:
 - and-introduction: $p \rightarrow (q \rightarrow r) \Leftrightarrow p \wedge q \rightarrow r$
 - or-introduction: $p \Rightarrow p \vee q$
 - or-elimination: $(p \rightarrow r) \wedge (q \rightarrow r) \wedge (p \vee q) \Rightarrow r$
 - not-introduction: $(p \rightarrow F) \Rightarrow \neg p$

I. Predicate Logic

- In propositional logic, we assert truths about boolean values; in predicate logic, we assert truths about values from one or more “domains of discourse” like the integers.
 - We extend propositional logic with domains (sets of values), variables whose values range over these domains, and operations on values (e.g. addition).
 - E.g., with the integers we add the set \mathbb{Z} , operations like $+$, $-$, $*$, $/$, $\%$ (mod), and the relations $=$, \neq , $<$, $>$, \leq , and \geq .
- A **predicate** is a logical assertion that describes some property of values.
 - To describe properties involving values, we add basic relations on values (e.g., less-than), and we add quantified predicates so that we can talk about properties relative to sets of values. E.g., “for all integers x , either x is negative or nonnegative”.

Universal Quantification

- A **universally quantified predicate** (or just “**universal**” for short) has the form $(\forall x \in S . p)$ where S is a set and p (the body of the universal) is a predicate involving x . E.g., every natural number > 1 is $<$ its own square: $(\forall x \in \mathbb{N} . x > 1 \rightarrow x < x^2)$.
 - Often we leave out the set if it is understood. E.g., $(\forall x . x > 1 \rightarrow x < x^2)$.
 - We may abbreviate further this using a “bounded” quantifier: $(\forall x > 1 . x < x^2)$.
 - In general, $\forall p . q$ means $\forall x . p \rightarrow q$ where x appears in p and is understood to be the variable we are quantifying over.

Existential Quantification

- An **existentially quantified predicate** (or just “**existential**” for short) has the form $(\exists x \in S . p)$ where S is a set and p (the body of the existential) is a predicate involving x . E.g., there is a nonzero integer that equals its own square: $(\exists x \in \mathbb{Z} . x \neq 0 \wedge x = x^2)$.
 - Often we leave out the set if it is understood. E.g., $(\exists x . x \neq 0 \wedge x = x^2)$.
 - We may abbreviate further this using a **bounded quantifier** as $(\exists x \neq 0 . x = x^2)$.
 - In general, $\exists p . q$ means $\exists x . p \wedge q$ where x appears in p and is understood to be the variable we are quantifying over.

Parentheses For Quantified Predicates

- We'll treat \forall and \exists as having low precedence. (Note: Some people use high precedence).
 - E.g. $\forall x \in \mathbb{N} . x > 1 \rightarrow x < x^2$ means $\forall x \in \mathbb{N} . (x > 1 \rightarrow (x < x^2))$

- With nested quantifiers, as in $(\forall x \in \mathbb{Z} . (\exists y \in \mathbb{Z} . y \leq x^2))$, we can omit the parentheses around the inner quantified predicate if the right parenthesis is next to the right parenthesis of the outer quantified predicate.
 - E.g., $\forall x \in \mathbb{Z} . \exists y \in \mathbb{Z} . y \leq x^2$ means $(\forall x \in \mathbb{Z} . (\exists y \in \mathbb{Z} . y \leq x^2))$.
 - In the other direction: $\forall x \in \mathbb{Z} . \exists y \in \mathbb{Z} . x > y^2 \rightarrow x > 0$ means $(\forall x \in \mathbb{Z} . (\exists y \in \mathbb{Z} . (x > y^2 \rightarrow x > 0)))$ [which is true], not $(\forall x \in \mathbb{Z} . (\exists y \in \mathbb{Z} . x > y^2 \rightarrow x > 0))$ [which is different].

Proof rules for Quantified Predicates

- In general, to prove $\forall x . p$, you prove p but without imposing any restrictions on x . If you need to restrict x , then this needs to be part of the body of the quantified predicate.
- Example: To prove $\forall x \in \mathbb{Z} . x \neq 0 \rightarrow x \leq x^2$, we can say "Let x be an integer. Assume that x isn't zero. In that case, $x \leq x^2$."
- For quantified predicates, there are two more DeMorgan's Laws :
 - $(\neg \forall x . p) \Leftrightarrow (\exists x . \neg p)$
 - $(\neg \exists x . p) \Leftrightarrow (\forall x . \neg p)$
- With bounded quantifiers, because of how \rightarrow , \neg , and \wedge are related,
 - $(\neg \forall p . q) \Leftrightarrow (\exists p . \neg q)$. I.e., $(\neg \forall x . p \rightarrow q) \Leftrightarrow (\exists x . \neg(p \rightarrow q)) \Leftrightarrow (\exists x . p \wedge \neg q) \Leftrightarrow (\exists p . \neg q)$.
 - $(\neg \exists p . q) \Leftrightarrow (\forall p . \neg q)$ I.e., $(\neg \exists x . p \wedge q) \Leftrightarrow (\forall x . \neg(p \wedge q)) \Leftrightarrow (\forall x . \neg p \vee \neg q) \Leftrightarrow (\forall x . p \rightarrow \neg q) \Leftrightarrow (\forall p . \neg q)$.
- Example: $\neg(\exists x . x > 0) \Leftrightarrow (\forall x . \neg(x > 0)) \Leftrightarrow (\forall x . x \leq 0)$ [these are all false]
- Example: $\neg(\exists x > 0 . x^2 = x) \Leftrightarrow (\forall x > 0 . x^2 \neq x)$ [these are false]
- Example $\neg(\exists x . x \leq 0 \wedge x > 0) \Leftrightarrow (\forall x . \neg(x \leq 0 \wedge x > 0)) \Leftrightarrow (\forall x . x > 0 \vee x \leq 0)$.

• Predicate Functions

- Often, we'll give names to predicates and parameterize them.
- Example: we might define $Even(x) \equiv (x \% 2) = 0$, where $\%$ is the remainder operator.
 - So e.g., $Even(3) \equiv (3 \% 2) = 0 \Leftrightarrow 1 = 0 \Leftrightarrow F$.
- Example: $IsZero(b, m) \equiv \forall j . 0 \leq j < m < sizeof(b) \rightarrow b[j] = 0$
 - Here b is an array with indexes $0, 1, \dots$; we're assuming $sizeof(b)$ gives the number of elements in b . $IsZero(b, m)$ is true exactly when all of $b[0], b[1], \dots, b[m-1]$ are zero.
- Example: $SortedUp(b, m, n) \Leftrightarrow \forall i . m \leq i \wedge i < n < sizeof(b) \rightarrow b[i] \leq b[i+1]$
 - Here b is an array; it's sorted upward on the segment $m..n$ if each element $b[m], b[m+1], \dots, b[n-1]$ is \leq the element to its right.
 - E.g., if $b[0..3]$ are 1, 3, 5, 2, then $SortedUp(b, 0, 2)$ is true but $SortedUp(b, 0, 3)$ is false.