CS477 Formal Software Development Methods

Elsa L Gunter 2112 SC, UIUC egunter@illinois.edu http://courses.engr.illinois.edu/cs477

Slides based in part on previous lectures by Mahesh Vishwanathan, and by Gul Agha

Semantics of Propositional Logic: Model Theory

Model for Propositional Logic has three parts

- Mathematical set of values used as meaning of propositions
- Interpretation function giving meaning to props built from logical connectives, via structural recursion

Standard Model of Propositional Logic

- $\mathcal{B} = \{ true, false \}$ boolean values
- $v : AP \rightarrow B$ a valuation
- Interpretation function ...

Semantics of Propositional Logic: Model Theory

Standard Model of Propositional Logic (cont)

Elsa L Gunter ()

- Standard interpretation \mathcal{I}_{v} defined by structural induction on formulae:
 - $\mathcal{I}_{\nu}(\mathbf{T}) = \text{true and } \mathcal{I}_{\nu}(\mathbf{F}) = \text{false}$
 - If $a \in AP$ then $\mathcal{I}_v(a) = v(a)$
 - For $p \in PROP$, if $\mathcal{I}_{v}(p) = \text{true}$ then $\mathcal{I}_{v}(\neg p) = \text{false}$, and if
 - $\mathcal{I}_{v}(p) =$ false then $\mathcal{I}_{v}(\neg p) =$ true
 - For $p, q \in PROP$
 - If $\mathcal{I}_{\nu}(p)=\mathrm{true}$ and $\mathcal{I}_{\nu}(q)=\mathrm{true},$ then $\mathcal{I}_{\nu}(p\wedge q)=\mathrm{true},$ else
 - $\mathcal{I}_{\nu}(p \land q) = \text{false}$ If $\mathcal{I}_{\nu}(p) = \text{true or } \mathcal{I}_{\nu}(q) = \text{true, then } \mathcal{I}_{\nu}(p \lor q) = \text{true, else}$
 - $\mathcal{I}_{\nu}(p \lor q) = \text{false}$
 - If $\mathcal{I}_{\nu}(q) = \text{true or } \mathcal{I}_{\nu}(p) = \text{false, then } \mathcal{I}_{\nu}(p \Rightarrow q) = \text{true, else } \mathcal{I}_{\nu}(p \Rightarrow q) = \text{false}$
 - If $\mathcal{I}_{\nu}(p) = \mathcal{I}_{\nu}(q)$ then $\mathcal{I}_{\nu}(p \Leftrightarrow q) = \text{true}$, else $\mathcal{I}_{\nu}(p \Leftrightarrow q) = \text{false}$

Truth Tables

Truth Tables

sa L Gunter ()

Interpretation function often described by truth table

р	q	$\neg p$	$p \wedge q$	$p \lor q$	$p \Rightarrow q$	$p \Leftrightarrow q$
true	true					
true	false					
false	true					
false	false					

Interpretation function often described by truth table

p	q	$\neg p$	$p \wedge q$	$p \lor q$	$p \Rightarrow q$	$p \Leftrightarrow q$
true	true	false	true			
true	false	false	false			
false	true	true	false			
false	false	true	false			

Truth Tables

Elsa L Gunter ()

Interpretation function often described by truth table

р	q	$\neg p$	$p \wedge q$	$p \lor q$	$p \Rightarrow q$	$p \Leftrightarrow q$
true	true	false				
true	false	false				
false	true	true				
false	false	true				

Truth Tables

Elsa L Gunter ()

Interpretation	function	often	described	by	truth	table
----------------	----------	-------	-----------	----	-------	-------

p	q	$\neg p$	$p \wedge q$	$p \lor q$	$p \Rightarrow q$	$p \Leftrightarrow q$
true	true	false	true	true		
true	false	false	false	true		
false	true	true	false	true		
false	false	true	false	false		

CS477 Formal Software Develop

Truth Tables

Elsa L Gunter ()

Interpretation function often described by truth table

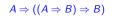
р	q	$\neg p$	$p \wedge q$	$p \lor q$	$p \Rightarrow q$	$p \Leftrightarrow q$
true	true	false	true	true	true	
true	false	false	false	true	false	
false	true	true	false	true	true	
false	false	true	false	false	true	

CS477 Formal Software Develo

terpretatio	on fun	ction o	ften de	scribed	by trut	h table		 (B, I) is the standard model of proposition logic Given valuation v and proposition p ∈ PROP, write v ⊨ p iff I_v(p) = true
[p true	q true	<i>¬p</i> false	$p \wedge q$ true	$p \lor q$ true	$p \Rightarrow q$ true	$p \Leftrightarrow q$ true	 More fully written as B, I, v ⊨ p Say v satisfies p, or v models p Write v ⊭ p if I_v(p) = false
	true false	false true	false true	false false	true true	false true	false false	 <i>p</i> is satisfiable if there exists valuation <i>v</i> such that <i>v</i> = <i>p</i> <i>p</i> is valid, a.k.a. a tautology if for every valuation <i>v</i> we have <i>v</i> =
L	false	false	true	false	false	true	true	 <i>p</i> is logically equivalent to <i>q</i>, <i>p</i> ≡ <i>q</i> if for every valuation, <i>v</i>, we h <i>v</i> ⊨ <i>p</i> iff <i>v</i> ⊨ <i>q</i> Claim: Logical equivalence is an equivalence relation

Examp	le]	lauto	logv
Examp		auto	·~БЛ

Elsa L Gunter ()



Α	В	$A \Rightarrow B$	$(A \Rightarrow B) \Rightarrow B$	$A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$
true	true			
true	false			
false	true			
false	false			

CS477 Formal Software Development Me

- AP

. =

/ 10

Example Tautology

nter ()

Elsa I. G

$$A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$$

A	В	$A \Rightarrow B$	$(A \Rightarrow B) \Rightarrow B$	$A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$
true	true	true		
true	false	false		
false	true	true		
false	false	true		

CS477 Formal Software Development Method

ple Tautology	Example Tautology
$A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$	$A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$
$A B A \Rightarrow B (A \Rightarrow B) \Rightarrow B A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$	$ \begin{array}{ c c c c c } \hline A & B & A \Rightarrow B & (A \Rightarrow B) \Rightarrow B & A \Rightarrow ((A \Rightarrow B) \Rightarrow B) \\ \hline \end{array} $
true true true true	true true true true true
true false false true false true false true true false true true true true false true false true false true false true false true false false true false fal	truefalsefalsetruetruefalsetruetruetruetrue
false false true false	false false true false true
েচ্য কেন্ট্র হেট্য হেট্য হ sa L Gunter () CS477 Formal Software Development Method / 19	(미) (중477 Formal Software Development Method
ple Tautology – Your Turn	Example: Logical Equivalence
	$A \Rightarrow B \equiv ((\neg A) \lor B)$
	$A \mid B \mid A \Rightarrow B \mid \neg A \mid (\neg A) \lor B$
	true true true false true
	truefalsefalsefalsefalsetruetruetrue
	false false true true true
(미) (CS477 Formal Software Development Method / 19	Elsa L Gunter () CS477 Formal Software Development Method
Useful Logical Equivalences	Logical Equivalence a Structural Congruence
$\neg \neg A \equiv A \qquad \neg \mathbf{T} \equiv \mathbf{F} \qquad \neg \mathbf{F} \equiv \mathbf{T}$ $(A \lor A) \equiv A \qquad (A \lor B) \lor C \equiv A \lor (B \lor C)$ $(A \land A) \equiv A \qquad (A \land B) \land C \equiv A \land (B \land C)$ $A \lor B \equiv B \lor A \qquad \neg (A \lor B) \equiv (\neg A) \land (\neg B)$ $A \land B \equiv B \land A \qquad \neg (A \land B) \equiv (\neg A) \lor (\neg B)$ $(A \land \neg A) \equiv \mathbf{F} \qquad A \lor (B \land C) \equiv (A \lor B) \land (A \lor C)$ $(A \lor \neg A) \equiv \mathbf{T} \qquad (A \land B) \lor C \equiv (A \lor C) \land (B \lor C)$ $(\mathbf{T} \land A) \equiv \mathbf{T} \qquad (A \land B) \lor C \equiv (A \land C) \lor (B \land C)$ $(\mathbf{T} \lor A) \equiv \mathbf{T} \qquad (A \land B) \lor C \equiv (A \land C) \lor (B \land C)$ $(\mathbf{T} \lor A) \equiv \mathbf{T} \qquad (A \land B) \lor C \equiv (A \land C) \lor (B \land C)$ $(\mathbf{F} \land A) \equiv \mathbf{F} \qquad (\mathbf{F} \lor A) \equiv A$	Theorem Logical equivalence is a structural congruence. That is, if $p \equiv p'$ and $q \equiv q'$ then $\bigcirc \neg p \equiv \neg p'$ $\bigcirc p \land q \equiv p' \land q'$ $\bigcirc p \lor q \equiv p' \land q'$ $\bigcirc p \Rightarrow q \equiv p' \Rightarrow q'$ $\bigcirc p \Leftrightarrow q \equiv p' \Leftrightarrow q'$

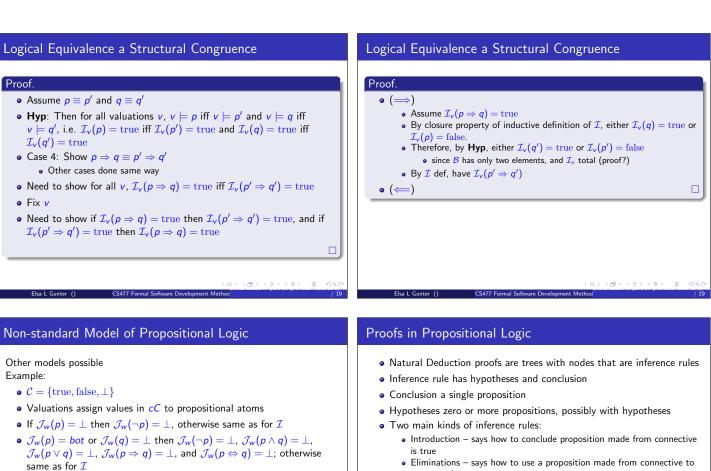
< ㅁ > < @ > < 흔 > < 흔 > · 흔 · 이익

· ()

CS477 Formal Software Developm

Elsa I (

nt Ma



• Note: $A \lor \neg A \not\equiv \mathbf{T}$

- Eliminations says how to use a proposition made from connective to prove result
- Inference rules associated with connectives
- Rule with no hypotheses called an axiom

