CS 214
Introduction to Discrete Structures

Chapter 1
Formal Logic

Mikel D. Petty, Ph.D.

THE UNIVERSITY OF
ALABAMA IN HUNTSVILLE

Chapter sections and objectives

- 1.1 Statements, Symbolic Representation, and Tautologies
  - Use the formal symbols of propositional logic
  - Construct truth tables for logic expressions
- 1.2 Propositional Logic
  - Construct formal proofs in propositional logic
  - Use formal proofs to analyze English statements
- 1.3 Quantifiers, Predicates, and Validity
  - Use the symbols of predicate logic
  - Find the truth value of interpretations of predicate logic expressions
  - Use predicate logic to represent English statements

- 1.4 Predicate Logic — Not covered in CS 214
- 1.5 Logic Programming — Not covered in CS 214
- 1.6 Proof of Correctness — Not covered in CS 214

Sample problem

You have been selected to serve on jury duty for a criminal case. The attorney for the defense argues as follows:

"If my client is guilty, then the knife was in the drawer. Either the knife was not in the drawer or Jason Pritchard saw the knife. If the knife was not there on October 10, it follows that Jason Pritchard did not see the knife. Furthermore, if the knife was there on October 10, then the knife was in the drawer and also the hammer was in the barn. But we all know that the hammer was not in the barn. Therefore, ladies and gentlemen of the jury, my client is innocent."

Is the attorney's argument sound? How should you vote?

1.1 Statements, Symbolic Representation, and Tautologies

Statements

- Definition
  - A statement is a sentence (or assertion) that is either true or false.
  - AKA proposition, hence propositional logic
- Examples
  - Ten is less than seven. Statement
  - Cheyenne is the capital of Wyoming. Statement
  - She is very talented. Not a statement
  - There is life on other planets. Statement
- Notation for statements
  - Italic capital letters near beginning of alphabet, e.g., A, B, C, used to represent statements
  - Statement letters have truth value, true (T) or false (F)

- Compound statements
  - A statement combining multiple sub-statements
  - Combination done with logical connectives
    - Conjunction $\wedge$
    - Disjunction $\vee$
    - Implication $\rightarrow$
    - Equivalence $\leftrightarrow$
    - Negation $'$

    Defined later
- Truth value of compound statement depends on
  - Truth values of sub-statements
  - Logical Connectives
- Logical expression
  - Sequence of statement letters, connective symbols, grouping symbols, e.g., $(A \wedge B) \rightarrow C$
  - Generic term for any logic statement, simple or compound, or any part thereof

Conjunction

- Definition
  - English "and"
  - e.g., "Elephants are big and baseballs are round."
  - True if and only if both sub-expressions are true
- Notation
  - Conjunction symbol $\wedge$, e.g., $A \wedge B$, read "A and B"
  - $A \wedge B$ is conjunction, A and B are conjuncts

A	B	$A \wedge B$
T	T	T
T	F	F
F	T	F
F	F	F

Table 1.1

Truth table
Gives truth value of compound statement for all possible truth values of sub-statements

Disjunction

- Definition
  - English "or"
  - e.g., "He is hungry or she is thirsty."
  - True if either or both sub-expressions are true
- Notation
  - Disjunction symbol $\vee$, e.g., $A \vee B$, read "A or B"
  - $A \vee B$ is disjunction, A and B are disjuncts

A	B	$A \vee B$
T	T	T
T	F	T
F	T	T
F	F	F

Table 1.2

Implication

- Definition
  - English "if … then …"
  - e.g., "If it rains, then the road is wet."
  - True if … see table
- Notation
  - Implication symbol $\rightarrow$, e.g., $A \rightarrow B$, read "A implies B"
  - $A \rightarrow B$ is implication, A is antecedent, B is consequent

Truth of A and B

Truth of implication statement $A \rightarrow B$

A	B	$A \rightarrow B$
T	T	T
T	F	F
F	T	T
F	F	T

"Trivially true"

Practice 2

Equivalence

- Definition
  - English "if and only if" AKA iff
  - e.g., "She will pass if and only if she studies."
  - True if both sub-expressions have same truth value
  - Shorthand for $(A \to B) \wedge (B \to A)$
- Notation
  - Symbol $\leftrightarrow$, e.g., $A \leftrightarrow B$, read "A is equivalent to B"

Table 1.3

A	B	$A \leftrightarrow B$	$A \to B$	$B \to A$	$(A \to B) \wedge (B \to A)$
T	T	T	T	T	T
T	F	F	F	T	F
F	T	F	T	F	F
F	F	T	T	T	T

Negation

- Definition
  - English "not"
  - e.g., "It will not rain tomorrow."
  - True if sub-expression false, and vice versa
  - Unary, rather than binary, connective
- Notation
  - Symbol $'$, e.g., A', read "not A"

A	A'
T	F
F	T

Practice 4

Connectives summary

A	B	$A \wedge B$	$A \vee B$	$A \rightarrow B$	$A \leftrightarrow B$	A'
T	T	T	T	T	T	F
T	F	F	T	F	F	F
F	T	F	T	T	F	T
F	F	F	F	T	T	T

Table 1.4

"Connectives" AKA "logical operations" or "operations".

Logical connectives in English

English Word	Logical Connective	Logical Expression
and; but; also; in addition; moreover	Conjunction	$A \wedge B$
or	Disjunction	$A \vee B$
if A, then B A implies B A, therefore B A only if B B follows from A A is a sufficient condition for B B is a necessary condition for A	Implication	$A \rightarrow B$
A if and only if B A is necessary and sufficient for B	Equivalence	$A \leftrightarrow B$
not A it is false that A it is not true that A	Negation	A'

Table 1.5

- Implication sentences
  - Different sentence forms for different English terms
  - Rewriting in if-then form clarifies meaning
- Examples

1. Fire is a necessary condition for smoke.
   If there is smoke, then there is fire. Example 2

2. A sufficient condition for network failure
   is that the central switch goes down.
   If the central switch goes down, then the network will fail.

3. The avocados are ripe only if they are dark and soft.
   If the avocados are ripe, then they are dark and soft.

4. A good diet is a necessary condition for a healthy cat.
   If the cat is healthy, then it has a good diet. Practice 5bcd

Negations in English

- Expressing negations of compound statements in English must be done carefully
- Same is true in logic

English Statement	Correct Negations	Incorrect Negations
It will rain tomorrow.	It is false that it will rain tomorrow. It will not rain tomorrow.	
Peter is tall and thin.	It is false that Peter is tall and thin. Peter is not tall or he is not thin. Peter is short or fat.	Peter is short and fat. Too strong; Peter fails to have both properties, but may still have one of the properties.
The river is shallow or polluted.	It is false that the river is shallow or polluted. The river is neither shallow nor polluted. The river is deep and unpolluted.	The river is not shallow or not polluted. Too weak; the river fails to have either property, not just fails to have one property.

Table 1.6

Negating implication

- English

  Statement: If it is raining, then the road is wet.

  Negation: It is raining but (or and) the road is not wet.

- Logic

  Statement: $A \rightarrow B$

  Negation: $(A \rightarrow B)'$ which is equivalent to $A \wedge B'$

A	B	$A \rightarrow B$	$(A \rightarrow B)'$
T	T	T	F
T	F	F	T
F	T	T	F
F	F	T	F

A	B	B'	$A \wedge B'$
T	T	F	F
T	F	T	T
F	T	F	F
F	F	T	F

Well-formed formulas

- Definition
  - Logical expressions formed from statement letters, connectives, and parentheses
  - Syntax rules define correct logical expressions
    - e.g., $(A \rightarrow B) \wedge (B \rightarrow A)$, correct syntax, logical expression
    - e.g., $A\ )) \wedge \wedge \rightarrow BC$, not correct syntax, not logical expression
  - Logical expressions AKA "well-formed formulas", AKA wffs, pronounced "wiffs"
- Some wff syntax rules (not complete)
  - Connectives have correct number of sub-expressions
  - Parentheses balanced

- Precedence determines expression evaluation
  - Connective precedence order
    1. connectives in parentheses, innermost first
    2. $'$ (not)
    3. $\wedge$ (and), $\vee$ (or)
    4. $\rightarrow$ (implies)
    5. $\leftrightarrow$ (equivalence)
  - e.g., $A \vee B'$ is $A \vee (B')$, not $(A \vee B)'$
  - e.g., $A \vee B \rightarrow C$ is $(A \vee B) \rightarrow C$, not $A \vee (B \rightarrow C)$

What about connectives at the same level, e.g., $A \wedge B \vee C$? Problems can arise; see table. Use parens to disambiguate.

A	B	C	$A \wedge B$	$(A \wedge B) \vee C$	$B \vee C$	$A \wedge (B \vee C)$
TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE
TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE
FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE

- Main connective
  - Connective applied last by precedence in a wff is main connective
  - Examples
    - In $A \wedge (B \rightarrow C)'$, main connective is $\wedge$
    - In $((A \vee B) \wedge C) \rightarrow (B \vee C')$, main connective is $\rightarrow$
- wff notation
  - Capital letters near end of alphabet, e.g., P, Q, R, S, used to represent wffs
  - e.g., $((A \vee B) \wedge C) \rightarrow (B \vee C')$ can be written as $P \rightarrow Q$ where $P = ((A \vee B) \wedge C)$ and $Q = (B \vee C')$
  - No change in logical meaning

Constructing truth tables

- List all combinations of statement truth values
- Determine truth value of each sub-expression
- Apply connectives in precedence order
- Main connective in rightmost column

Example truth table construction

$$A \vee B' \rightarrow (A \vee B)'$$

A	B	B'	$A \vee B'$	$A \vee B$	$(A \vee B)'$	$A \vee B' \rightarrow (A \vee B)'$
T	T	F	T	T	F	F
T	F	T	T	T	F	F
F	T	F	F	T	F	T
F	F	T	T	F	T	T
precedence		2	4	1	3	5

Table 1.7

1 Parentheses first
2 $'$ before $\vee$ and $\rightarrow$
3 $'$ before $\vee$ and $\rightarrow$; 2 and 3 could be reversed
4 $\vee$ before $\rightarrow$
5 $\rightarrow$ is main connective

How many rows in a truth table?

- All combinations of statement letter truth values
- For n different statement letters, 2^n rows
- Mnemonic: binary numbers, 1 = T and 0 = F

A	*B*
T	T
T	F
F	T
F	F

A	*B*	*C*	
T	T	T	= 111 (7)
T	T	F	= 110 (6)
T	F	T	= 101 (5)
T	F	F	= 100 (4)
F	T	T	= 011 (3)
F	T	F	= 010 (2)
F	F	T	= 001 (1)
F	F	F	= 000 (0)

Table 1.8

Example truth table construction

$$[(A \wedge B') \rightarrow C']'$$

A	B	C	B'	$A \wedge B'$	C'	$(A \wedge B') \rightarrow C'$	$[(A \wedge B') \rightarrow C']'$
T	T	T	F	F	F	T	F
T	T	F	F	F	T	T	F
T	F	T	T	T	F	F	T
T	F	F	T	T	T	T	F
F	T	T	F	F	F	T	F
F	T	F	F	F	T	T	F
F	F	T	T	F	F	T	F
F	F	F	T	F	T	T	F

Practice 7c

Example truth table construction

$(A \rightarrow B) \leftrightarrow (B \rightarrow A)$ AKA converse

A	B	$A \rightarrow B$	$B \rightarrow A$	$(A \rightarrow B) \leftrightarrow (B \rightarrow A)$
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

Practice 7a

$(A \rightarrow B) \leftrightarrow (B' \rightarrow A')$ AKA contrapositive

A	B	A'	B'	$A \rightarrow B$	$B' \rightarrow A'$	$(A \rightarrow B) \leftrightarrow (B' \rightarrow A')$
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

Practice 7d

Variations of the implication statement

				Implication	Contrapositive	Converse	Inverse
A	B	A'	B'	$A \rightarrow B$	$B' \rightarrow A'$	$B \rightarrow A$	$A' \rightarrow B'$
T	T	F	F	T	T	T	T
T	F	F	T	F	F	T	T
F	T	T	F	T	T	F	F
F	F	T	T	T	T	T	T

$A \rightarrow B$ Implication
$B' \rightarrow A'$ Contrapositive, equivalent to implication
$B \rightarrow A$ Converse, not equivalent to implication
$A' \rightarrow B'$ Inverse, not equivalent to implication

Table 3.26 [Angel, 1997]

Tautologies and contradictions

- Tautology
  - A wff whose truth values are always true, regardless of truth values of statement letters
  - e.g., $A \vee A'$
  - Considered "intrinsically true" by its structure
- Contradication
  - A wff whose truth values are always false, regardless of truth values of statement letters
  - e.g., $A \wedge A'$
  - Considered "intrinsically false" by its structure

Equivalent wffs

- Definition
  - Two wffs that always have the same truth value, regardless of truth values of their statement letters, are equivalent
  - i.e., if $P \leftrightarrow Q$ a tautology then P and Q equivalent wffs
  - Written $P \Leftrightarrow Q$
- $\leftrightarrow$ compared to $\Leftrightarrow$
  - $P \leftrightarrow Q$ an expression, may be T or F; "equivalence"
  - $P \Leftrightarrow Q$ states fact that $P \leftrightarrow Q$ always true; "tautological equivalence"

Tautological equivalences

- Important tautological equivalences

Commutative	1a.	$A \vee B \Leftrightarrow B \vee A$			
	1b.	$A \wedge B \Leftrightarrow B \wedge A$			
Associative	2a.	$(A \vee B) \vee C \Leftrightarrow A \vee (B \vee C)$			
	2b.	$(A \wedge B) \wedge C \Leftrightarrow A \wedge (B \wedge C)$			
Distributive	3a.	$A \vee (B \wedge C) \Leftrightarrow (A \vee B) \wedge (A \vee C)$			
	3b.	$A \wedge (B \vee C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$			
Identity	4a.	$A \vee 0 \Leftrightarrow A$	[Gersting, 2014]	$A \vee \mathrm{F} \Leftrightarrow A$	Alternate
	4b.	$A \wedge 1 \Leftrightarrow A$		$A \wedge \mathrm{T} \Leftrightarrow A$	
Complement	5a.	$A \vee A' \Leftrightarrow 1$		$A \vee A' \Leftrightarrow \mathrm{T}$	
	5b.	$A \wedge A' \Leftrightarrow 0$		$A \wedge A' \Leftrightarrow \mathrm{F}$	
De Morgan's Laws	6a.	$(A \vee B)' \Leftrightarrow A' \wedge B'$			
	6b.	$(A \wedge B)' \Leftrightarrow A' \vee B'$			

where 1 = T = any tautology and 0 = F = any contradiction.

- Verifying tautological equivalences
  - Construct truth table
  - Confirm always true

Equivalence 1a $A \vee B \Leftrightarrow B \vee A$
Commutative property for disjunction

A	B	$A \vee B$	$B \vee A$	$A \vee B \leftrightarrow B \vee A$
T	T	T	T	T
T	F	T	T	T
F	T	T	T	T
F	F	F	F	T

Table 1.9(a)

Equivalence 4b $A \wedge 1 \Leftrightarrow A$
Identity property for conjunction

A	1	$A \wedge 1$	$A \wedge 1 \leftrightarrow A$
T	T	T	T
F	T	F	T

Table 1.9(b)

Replacing wffs

- Definition
  - Suppose $P \Leftrightarrow Q$
  - Then wff P can be replaced by wff Q in any wff R that contains P
  - The resulting wff $R_Q \Leftrightarrow R$
- Example

$R = (A \rightarrow B) \rightarrow B$

$P = A \rightarrow B$

$Q = B' \rightarrow A'$

$P \Leftrightarrow Q$ (from Practice 7)

$R_Q = (B' \rightarrow A') \rightarrow B$ (replace P with Q)

A	B	$A \rightarrow B$	$(A \rightarrow B) \rightarrow B$
T	T	T	T
T	F	F	T
F	T	T	T
F	F	T	F

Table 1.10(a)

R

A	B	A'	B'	$B' \rightarrow A'$	$(B' \rightarrow A') \rightarrow B$
T	T	F	F	T	T
T	F	F	T	F	T
F	T	T	F	T	T
F	F	T	T	T	F

Table 1.10(b)

R_Q, i.e., R with P replaced by Q

$$R \Leftrightarrow R_Q$$

Example tautological equivalence proofs

Prove $(A \wedge B') \wedge C \leftrightarrow (A \wedge C) \wedge B'$ Exercise 27a

$$
\begin{array}{lll}
 & (A \wedge B') \wedge C & \\
\Leftrightarrow & A \wedge (B' \wedge C) & \text{Associative} \\
\Leftrightarrow & A \wedge (C \wedge B') & \text{Commutative} \\
\Leftrightarrow & (A \wedge C) \wedge B' & \text{Associative}
\end{array}
$$

Prove $(A \vee B) \wedge (A \vee B') \leftrightarrow A$ Exercise 27b

$$
\begin{array}{lll}
 & (A \vee B) \wedge (A \vee B') & \\
\Leftrightarrow & A \vee (B \wedge B') & \text{Distributive} \\
\Leftrightarrow & A \vee 0 & \text{Complement} \\
\Leftrightarrow & A & \text{Identity}
\end{array}
$$

Example tautological equivalence proof

Prove $(A \wedge B')' \vee B \leftrightarrow A' \vee B$ Exercise 28a

$(A \wedge B')' \vee B$	
$\Leftrightarrow (A' \vee (B')') \vee B$	De Morgan
$\Leftrightarrow (A' \vee B) \vee B$	Double negative (dn), Exercise 26b
$\Leftrightarrow A' \vee (B \vee B)$	Associative
$\Leftrightarrow A' \vee B$	Self reference, Exercise 26g

Tautological equivalences in programming

```
if    ((outflow > inflow) and not ((outflow > inflow) and (pressure < 1000)))
      do something;
else
      do something else;
```

Let A = "outflow > inflow" and B = "pressure < 1000".
Then conditional expression (if statement) has form $A \wedge (A \wedge B)'$.

$A \wedge (A \wedge B)'$	
$\Leftrightarrow A \wedge (A' \vee B')$	De Morgan's
$\Leftrightarrow (A \wedge A') \vee (A \wedge B')$	Distributive
$\Leftrightarrow 0 \vee (A \wedge B')$	Complement **?**
$\Leftrightarrow (A \wedge B') \vee 0$	Commutative
$\Leftrightarrow (A \wedge B')$	Identity

```
if    (outflow > inflow) and not (pressure < 1000)
      do something;
else
      do something else;
```

Example 7

Algorithm

- Central concept of computer science
- Definitions of “algorithm”
  - “A set of instructions that can be mechanically executed in a finite amount of time to solve some problem.” [Gersting, 2014]
  - “A sequence of operations that tells you what to do at each step, depending on what the outcome of the previous step was.” [Goldstein, 2005]
  - “An ordered set of unambiguous, executable steps that define a terminating activity.” [Brookshear, 2000]
- An algorithm must be mechanically executable, unambiguous, terminating, and correct
- A program is an implementation of an algorithm

An algorithm to test for tautologies

- Logic can be processed algorithmically
- e.g., to determine if a wff is a tautology
  - Construct a truth table, check for all T
  - Other algorithms exist as well
- Example non-truth table algorithm
  - Applies to implication $P \rightarrow Q$
    - P, Q are wffs
    - $\rightarrow$ is main connective
  - Algorithm's basic method is "proof by contradiction"
    - Assume $P \rightarrow Q$ is not a tautology
    - Determine if this leads to contradiction
    - If so, $P \rightarrow Q$ must be tautology

Algorithm TautologyTest

TautologyTest(wff P; wff Q)

// Given wffs P and Q, decides whether the wff $P \rightarrow Q$ is a tautology.

1 // Assume $P \rightarrow Q$ is not a tautology

P = true // assign T to P

Q = false // assign F to Q **?**

repeat

  for each compound wff already assigned a truth value,

   assign the truth values determined for its components

2 **until** all occurrences of statement letters have truth values assigned

if some statement letter has two truth values

then // contradiction, assumption false

  write ("$P \rightarrow Q$ is a tautology.")

else // found a way to make $P \rightarrow Q$ false

  write ("$P \rightarrow Q$ is not a tautology.")

end if

end TautologyTest

TautologyTest example

- wff $(A \rightarrow B) \rightarrow (B' \rightarrow A')$
  - Implication, suitable for TautologyTest
  - $P \rightarrow Q$, where $P = A \rightarrow B$, $Q = B' \rightarrow A'$
- Processing

Initial assignments: $P = A \rightarrow B = \mathrm{T}$, $Q = B' \rightarrow A' = \mathrm{F}$

Loop assignments: $B' \rightarrow A' = \mathrm{F}$ determines $B' = \mathrm{T}$, $A' = \mathrm{F}$
therefore $B = \mathrm{F}$, $A = \mathrm{T}$
$A \rightarrow B = \mathrm{T}$, $A = \mathrm{T}$ determines $B = \mathrm{T}$

All statement letter occurrences assigned truth values, loop ends

Statement letter values: $A = \mathrm{T}$, $B = \mathrm{T}$, $B = \mathrm{F}$.

B has two values, contradiction found, assumption false,
thus wff $(A \rightarrow B) \rightarrow (B' \rightarrow A')$ a tautology.

Example 8

Algorithm TautologyTest

TautologyTest(wff P; wff Q)
// Given wffs P and Q, decides whether the wff $P \rightarrow Q$ is a tautology.

```
// Assume P → Q is not a tautology
P = true     // assign T to P
Q = false    // assign F to Q
repeat
   for each compound wff already assigned a truth value,
      assign the truth values determined for its components      ?1    ?2
   until all occurrences of statement letters have truth values assigned

if some statement letter has two truth values
then          // contradiction, assumption false
   write ("P → Q is a tautology.")
else          // found a way to make P → Q false
   write ("P → Q is not a tautology.")
end if
```

end TautologyTest

Section 1.1 homework assignment

See homework list for specific exercises and hints.

RFGMPR #1: Read the problem instructions carefully and completely.

1.2 Propositional Logic

Propositional logic

- Propositional logic
  - Reaching logical conclusions using given statements
  - aka statement logic, propositional calculus
  - Statements aka propositions,
    thus wffs aka propositional wffs
- Informal example

If Katherine studies, then she will pass.
Katherine studies.
Therefore, Katherine will pass.

What makes this argument valid?
How can valid arguments be recognized?

Valid and invalid arguments

- Definition of argument
  - Specific type of wff: $\rightarrow$ is main connective
  - Structure $P_1 \wedge P_2 \wedge \ldots \wedge P_n \rightarrow Q$
  - $P_1, P_2, \ldots, P_n$ are hypotheses, Q is conclusion
- When is an argument valid?
  - When can Q be logically deduced from $P_1, P_2, \ldots, P_n$?
  - When is Q a logical conclusion from $P_1, P_2, \ldots, P_n$?
  - When do $P_1, P_2, \ldots, P_n$ logically imply Q?
  - When does Q follow logically from $P_1, P_2, \ldots, P_n$?
- Clarification
  - Not interested in trivially true cases (false hypotheses)
  - The question is whether the truth of the hypotheses necessarily implies the truth of the conclusion

- An argument is valid
  - When $P_1 \wedge P_2 \wedge \ldots \wedge P_n \rightarrow Q$ is "intrinsically true", based on logical structure
  - The propositional wff $P_1 \wedge P_2 \wedge \ldots \wedge P_n \rightarrow Q$ is a valid argument when it is a tautology.
- Example argument, valid

Hyp: If Washington was the first president of the U. S., then Adams was the first vice president. $(A \rightarrow B)$
Washington was the first president. A
Con: Adams was the first vice president. B

- Logical structure of valid argument
  - $(A \rightarrow B) \wedge A \rightarrow B$
  - $(A \rightarrow B)$ and A hypotheses, B conclusion
  - AKA modus ponens

Example 10

- Confirming valid argument (modus ponens)
  - Logical structure: $(A \rightarrow B) \wedge A \rightarrow B$

A	B	$A \rightarrow B$	$(A \rightarrow B) \wedge A$	$(A \rightarrow B) \wedge A \rightarrow B$
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

Example 10

  - Equivalent by commutivity: $A \wedge (A \rightarrow B) \rightarrow B$

- Example argument, not valid

  Hyp: Washington was the first president of the U. S. A
  Jefferson wrote the Declaration of Independence. B
  Con: Every day has 24 hours. C

  Though true, hypotheses do not imply conclusion.

- Logical structure of invalid argument
  - $A \wedge B \rightarrow C$
  - A and B hypotheses, C conclusion
  - Not a tautology

Example 9

- Confirming invalid argument
  - Logical structure: $A \wedge B \rightarrow C$

A	B	C	$A \wedge B$	$(A \wedge B) \rightarrow C$
T	T	T	T	T
T	T	F	T	F
T	F	T	F	T
T	F	F	F	T
F	T	T	F	T
F	T	F	F	T
F	F	T	F	T
F	F	F	F	T

Not a tautology

Example 9

Determining validity

- Methods to determine validity
  - Construct truth table
  - Use special algorithm, e.g., TautologyTest from § 1.1
  - Prove using derivation rules
- Derivation rules
  - Manipulate wffs in a truth preserving manner
  - Starting from hypotheses, apply rules to get conclusion
- Proof sequence
  - Sequence of wffs
  - Each wff in sequence is either
    - Hypothesis wff
    - Result of applying derivation rule to earlier wff in sequence
- "Determining validity" AKA "proving" argument

Generic proof sequence

P_1 (hypothesis)
P_2 (hypothesis)
. . .
P_n (hypothesis)
wff_1 (obtained by applying derivation rule to earlier wff)
wff_2 (obtained by applying derivation rule to earlier wff)
. . .
Q (obtained by applying derivation rule to earlier wff)

Starting from hypotheses $P_1 \wedge P_2 \wedge \ldots \wedge P_n$,
derive conclusion Q
by applying truth preserving derivation rules.

Types of derivation rules

- Equivalence rules
  - Certain pairs of wffs are tautologically equivalent
  - Allow substitution of wff for equivalent wff, thus rewriting wffs in proof sequence
  - Can be applied in either "direction"
- Inference rules
  - If certain wffs are in sequence, a certain new wff can be added
  - Allow creation new wffs in proof sequence
  - Can be applied only in one "direction"
- Both types are truth preserving

Equivalence rules

Expression	Equivalent to	Name, abbreviation for rule
$P \vee Q$	$Q \vee P$	Commutative (comm)
$P \wedge Q$	$Q \wedge P$	
$(P \vee Q) \vee R$	$P \vee (Q \vee R)$	Associative (ass)
$(P \wedge Q) \wedge R$	$P \wedge (Q \wedge R)$	
$(P \vee Q)'$	$P' \wedge Q'$	De Morgan's Laws (De Morgan)
$(P \wedge Q)'$	$P' \vee Q'$	
$P \rightarrow Q$	$P' \vee Q$	Implication (imp)
P	$(P')'$	Double negation (dn)
$P \leftrightarrow Q$	$(P \rightarrow Q) \wedge (Q \rightarrow P)$	Definition of equivalence (equ)
$P \vee 0$	P	Identity
$P \wedge 1$	P	
$P \vee P'$	1	Complement
$P \wedge P'$	0	

?

Equivalence rules may be applied in either direction.

Table 1.11

Equivalence rule confirmation example

Confirming the implication equivalence rule,
i.e., $(P \rightarrow Q) \leftrightarrow (P' \vee Q)$

P	Q	$P \rightarrow Q$	P'	$P' \vee Q$	$(P \rightarrow Q) \leftrightarrow (P' \vee Q)$
T	T	T	F	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

Inference rules

From	Can derive	Name, abbreviation for rule
$P, P \to Q$	Q	Modus ponens (mp)
$P \to Q, Q'$	P'	Modus tollens (mt)
P, Q	$P \wedge Q$	Conjunction (con)
$P \wedge Q$	P, Q	Simplification (sim)
P	$P \vee Q$	Addition (add)

Table 1.12

Inference rules may be applied only one way.

Partial proof sequence example

Suppose $(A \wedge B') \rightarrow C$ and C' are hypotheses; then a proof sequence could begin as follows:

1.	$(A \wedge B') \rightarrow C$	hyp
2.	C'	hyp
3.	$(A \wedge B')'$	1, 2, mt
4.	$A' \vee (B')'$	3, De Morgan
5.	$A' \vee B$	4, dn

Practice 10

RFGMPR # 2: Don't skip the double negatives.

Complete proof sequence example

Prove the following argument is valid:

$$\underset{P_1}{A} \wedge \underset{P_2}{(B \longrightarrow C)} \wedge \underset{P_3}{[(A \wedge B) \longrightarrow (D \vee C')]} \wedge \underset{P_4}{B} \rightarrow \underset{Q}{D}$$

1.	A	hyp
2.	$B \longrightarrow C$	hyp
3.	$(A \wedge B) \longrightarrow (D \vee C')$	hyp
4.	B	hyp
5.	C	2, 4, mp
6.	$A \wedge B$	1, 4, con
7.	$D \vee C'$	3, 6, mp
8.	$C' \vee D$	7, comm
9.	$C \rightarrow D$	8, imp
10.	D	5, 9, mp

RFGMPR # 3: Start by listing what you know.

RFGMPR # 4: If you don't know what to do next, think about your goal.

Example 14

Complete proof sequence example

Prove the following argument is valid:

$[(A \vee B') \rightarrow C] \wedge (C \rightarrow D) \wedge A \rightarrow D$ Practice 11

1.	$(A \vee B') \rightarrow C$	hyp
2.	$C \rightarrow D$	hyp
3.	A	hyp
4.	$A \vee B'$	3, add
5.	C	1, 4, mp
6.	D	2, 5, mp

Proof sequence hints

1. Modus ponens often useful.
2. $(P \wedge Q)'$ and $(P \vee Q)'$ seldom useful.
   Use De Morgan's Law to give $P' \vee Q'$ and $P' \wedge Q'$.
3. $P \vee Q$ seldom useful.
   Use double negation to give $(P')' \vee Q$,
   then implication to give $P' \rightarrow Q$,
   may be useful later for modus ponens or modus tollens.

Deduction method

- Method
  - If argument to prove has implication as conclusion …
    $P_1 \wedge P_2 \wedge \ldots \wedge P_n \rightarrow (R \rightarrow S)$
  - … add R as additional hypothesis and derive S:
    $P_1 \wedge P_2 \wedge \ldots \wedge P_n \wedge R \rightarrow S$
- Applicability
  - Logically equivalent to the original argument
  - Can be applied more than once

Example proofs using deduction method

Prove $(A \rightarrow B) \wedge (B \rightarrow C) \rightarrow (A \rightarrow C)$

1.	$(A \rightarrow B)$	hyp
2.	$(B \rightarrow C)$	hyp
3.	A	hyp, ded
4.	B	1, 3, mp
5.	C	2, 4, mp

Practice 12

Prove $[(P \wedge Q) \rightarrow R] \rightarrow [P \rightarrow (Q \rightarrow R)]$

1.	$(P \wedge Q) \rightarrow R$	hyp
2.	P	hyp, deduction
3.	Q	hyp, deduction
4.	$P \wedge Q$	2, 3, con
5.	R	1, 4, mp

Exercise 30

Hypothetical syllogism

From $A \rightarrow B$ and $B \rightarrow C$, $A \rightarrow C$ can be derived, i.e.,

$(A \rightarrow B) \wedge (B \rightarrow C) \rightarrow (A \rightarrow C)$

This was just proven as an example of deduction.

Example use of hypothetical syllogism
Prove $(A' \vee B) \wedge (B \rightarrow C) \rightarrow (A \rightarrow C)$

1.	$A' \vee B$	hyp
2.	$B \rightarrow C$	hyp
3.	$A \rightarrow B$	1, imp
4.	$A \rightarrow C$	2, 3, hs

Example 17

More inference rules

From	Can derive	Name, abbreviation for rule
$P \rightarrow Q, Q \rightarrow R$	$P \rightarrow R$	Hypothetical syllogism - hs
$P \vee Q, P'$	Q	Disjunctive syllogism - ds
$P \rightarrow Q$	$Q' \rightarrow P'$	Contraposition - cont
$Q' \rightarrow P'$	$P \rightarrow Q$	
P	$P \wedge P$	Self reference - self
$P \vee P$	P	
$(P \wedge Q) \rightarrow R$	$P \rightarrow (Q \rightarrow R)$	Exportation - exp
P, P'	Q	Inconsistency
$P \wedge (Q \vee R)$	$(P \wedge Q) \vee (P \wedge R)$	Distributive - dist
$P \vee (Q \wedge R)$	$(P \vee Q) \wedge (P \vee R)$	Distributive - dist

Table 1.14

Complete proof sequence example

A = I was reading the newspaper in the kitchen.
B = My glasses are on the kitchen table.
C = I saw my glasses at breakfast.
D = I was reading the newspaper in the living room.
E = My glasses are on the coffee table.

All of the following are known to be true:

If was reading the newspaper in the kitchen, then my glasses are on the kitchen table.	$A \rightarrow B$
If my glasses are on the kitchen table, then I saw them at breakfast.	$B \rightarrow C$
I did not see my glasses at breakfast.	C'
I was reading the newspaper in the living room or I was reading the newspaper in the kitchen.	$D \vee A$
If I was reading the newspaper in the living room, then my glasses are on the coffee table.	$D \rightarrow E$

Example 2.3.8
[Epp, 2011]

Prove that my glasses are on the coffee table:

$(A \rightarrow B) \wedge (B \rightarrow C) \wedge C' \wedge (D \vee A) \wedge (D \rightarrow E) \rightarrow E$

1.	$A \rightarrow B$	hyp
2.	$B \rightarrow C$	hyp
3.	C'	hyp
4.	$D \vee A$	hyp
5.	$D \rightarrow E$	hyp
6.	$A \rightarrow C$	1, 2, hs
7.	A'	3, 6, mt
8.	D	4, 7, ds
9.	E	5, 8, mp

Example 2.3.8
[Epp, 2011]

Verbal arguments

- Arguments in English
  - e.g., attorney's claim, advertisement, debate
  - Can be checked for validity using propositional logic
- Process
  - Represent argument using propositional wffs
  - Prove argument valid
    - Truth table
    - Algorithm
    - Prove using derivation rules
- Meaning of analysis
  - Determines if argument is valid,
    i.e., if conclusion follows logically from hypotheses
  - Does not show that hypotheses are true

Verbal arguments example

If interest rates drop, the housing market will improve. Either the federal discount rate will drop or the housing market will not improve. Interest rates will drop. Therefore the federal discount rate will drop.

Statement letters

I = interest rates will drop

H = housing market will improve

F = federal discount rate will drop

Argument

$(I \rightarrow H) \wedge (F \vee H') \wedge I \rightarrow F$

Argument

$(I \longrightarrow H) \wedge (F \vee H') \wedge I \longrightarrow F$

Proof sequence

1.	$I \longrightarrow H$	hyp
2.	$F \vee H'$	hyp
3.	I	hyp
4.	$H' \vee F$	2, comm
5.	$H \rightarrow F$	4, imp
6.	$I \rightarrow F$	1, 5, hs
7.	F	3, 6, mp

Example 18

Solution to sample problem

If my client is guilty, then the knife was in the drawer.

Either the knife was not in the drawer or Jason Pritchard saw the knife.

If the knife was not there on October 10,
it follows that Jason Pritchard did not see the knife.

Furthermore, if the knife was there on October 10,
then the knife was in the drawer and also the hammer was in the barn.

But we all know that the hammer was not in the barn.

Therefore, ladies and gentlemen of the jury, my client is innocent.

Statement letters

A = client guilty

B = knife in drawer

C = Jason Pritchard saw knife

D = knife present October 10

E = hammer in the barn

Argument

$(A \rightarrow B) \wedge (B' \vee C) \wedge (D' \rightarrow C') \wedge (D \rightarrow (B \wedge E)) \wedge E' \rightarrow A'$

Argument

$(A \to B) \wedge (B' \vee C) \wedge (D' \to C') \wedge (D \to (B \wedge E)) \wedge E' \to A'$

Proof sequence

1.	$A \to B$	hyp
2.	$B' \vee C$	hyp
3.	$D' \to C'$	hyp
4.	$D \to (B \wedge E)$	hyp
5.	E'	hyp
6.	$E' \vee B'$	5, add
7.	$B' \vee E'$	6, comm
8.	$(B \wedge E)'$	7, De Morgan
9.	D'	4, 8, mt
10.	C'	3, 9, mp
11.	$B \to C$	2, imp
12.	B'	10, 11, mt
13.	A'	1, 12, mt

Exercise 56

Section 1.2 homework assignment

See homework list for specific exercises and hints.

1.3 Quantifiers, Predicates, and Validity

Elements of predicate logic

- Propositional wffs and propositional logic
  - Limited expressive power
  - e.g., cannot express "for every x, $x > 0$"
- Predicate logic
  - Increased expressive power
  - New features
    - Quantifiers
    - Predicates
    - Variables
    - Constants
  - New concept
    - Interpretation

Predicates

- Symbol $P(x)$, read "P of x"
  - Italic capital letters (not only P; Q, R, and others used)
- Represent some property an object may have
  - Letter sometimes mnemonic w.r.t. property
- $P(x)$ has truth value true or false,
  depending whether object x has property P
  - e.g., $P(x)$ is true iff x is positive, else false
- Predicates may have two or more arguments
  - e.g., $Q(x, y)$ is true iff $x > y$, else false

Syntax of predicate wffs

- Predicate wffs built from
  - Predicates
  - Quantifiers
  - Grouping symbols ( ), [ ]
  - Logical connectives from §1.1: $\wedge\ \vee \rightarrow \leftrightarrow\ '$
- Examples

predicate wffs $P(x) \vee Q(y)$

$(\forall x)[P(x) \rightarrow Q(x)]$

$(\forall x)((\exists y)[P(x, y) \wedge Q(x, y)] \rightarrow R(x))$

not a predicate wff $P(x)(\forall x) \wedge)\exists y$

Quantifiers

- Express how many objects have some property
- Universal quantifier
  - Symbol $\forall$, read “for every”
  - e.g., $(\forall x)(x > 0)$
  - Asserts that every object x has property $x > 0$
  - Variable x represents any object being discussed
- Existential quantifier
  - Symbol $\exists$, read “there exists”
  - e.g., $(\exists x)(x > 0)$
  - Asserts that at least one object x has property $x > 0$
  - Variable x represents any object being discussed

Interpretation of a predicate wff, first look

- Predicate wff has a truth value
- A wff's truth value depends on its interpretation
- Parts of an interpretation of a wff
  - Collection or set of objects from which x (or other variables) may be chosen; AKA domain
  - Definitions of properties that predicates $P(x)$ represent
- Examples
  - e.g., $(\forall x)P(x)$ true iff predicate $P(x)$ true for every object x in domain
  - e.g., $(\exists x)P(x)$ true iff predicate $P(x)$ true for at least one object x in domain

Ex	Interpretation		Truth values	
	Domain x	Property $P(x)$	$(\forall x)P(x)$	$(\exists x)P(x)$
1	positive integers	$x > 0$	T	T
2	all integers	$x > 0$	F	T
3	books in UAH library	x has a red cover	F	T
4	daffodils	x is yellow	F	T
5	flowers	x is yellow	F	T
6	flowers	x is a plant	T	T
7	all integers	x is positive or negative	F	T
8	birds	x can swim	F	T
9	positive integers	$x < 0$	F	F

?

Practice 15 & 16

$(\forall x)P(x) \Rightarrow (\exists x)P(x)$
but not vice versa.

4

8

Predicate arity

- Multivariable predicates
  - Predicates may have > 1 variables
    - unary, $P(x)$, e.g., $x > 0$
    - binary, $Q(x, y)$, e.g., $x < y$
    - ternary, $R(x, y, z)$, e.g., $x - y = z$
    - n-ary, $S(x_1, x_2, \ldots, x_n)$, e.g., $x_1 + x_2 + \ldots + x_n = 0$
- Quantifiers for multivariable predicates
  - Each variable may have a quantifier
  - Order of quantifiers is important
    - e.g., $(\forall x)[(\exists y)Q(x, y)]$, domain integers, $Q(x,y)$ property $x < y$
      "for every integer x, there exists integer y that is larger" true
    - e.g., $(\exists y)[(\forall x)Q(x, y)]$, domain integers, $Q(x,y)$ property $x < y$
      "there exists integer y larger than every integer x" false

Example 20

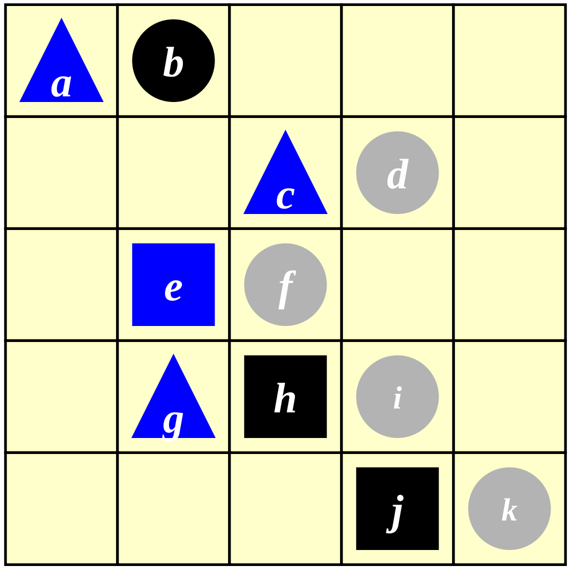
Predicate variables and constants

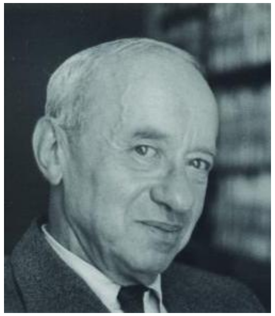
- Variables
  - Specific letter used for variable not important, e.g., $(\forall x)P(x)$ has same truth value $(\forall y)P(y)$
  - Different variables represent different or same objects
    - e.g., $(\forall x)(\exists y)Q(x,y)$, $Q(x, y)$ is $x \neq y$, is true; $y = -x$
    - e.g., $(\forall x)(\forall y)Q(x,y)$, $Q(x, y)$ is $x \neq y$, is false; $x = y = 0$
  - Same variable represents same object
    - e.g., $(\forall x)P(x) \wedge (\forall x)Q(x)$
    - e.g., $(\exists x)R(x, x)$
- Constants
  - May appear in predicates, e.g., 1, 2, 3, a, b, c
  - Interpreted as specific fixed object in domain
  - e.g., for domain integers, predicate $Q(x, y)$ is $x < y$ then $(\exists x)Q(x, 7)$ is true, but $(\forall x)Q(x, a)$ is false

Interpretation of a predicate wff, revisited

- A predicate wff interpretation consists of
  - The domain, a non-empty collection of objects
  - A definition of a property of domain objects for each predicate
  - An assignment of a particular domain object to each constant
- Predicate wff truth values are of interest
  - Within a given interpretation
  - Within all interpretations
  - Predicate wff that is true in all interpretations is valid

Predicate wff examples: Tarski's World





Alfred Tarski

$(\forall x)[Triangle(x) \rightarrow Blue(x)]$ True; a, c, g all blue

$(\forall x)[Blue(x) \rightarrow Triangle(x)]$ False; counterexample e

$(\exists y)[Square(y) \land RightOf(d, y)]$ True; example h

$(\exists z)[Square(z) \land Gray(z)]$ False; no gray square

Example 3.1.13 [Epp, 2011]

Quantifier scope

- Scope; portion of wff to which quantifier applies
- Scope rules; scope of quantifier is
  - Predicate or grouped expression following quantifier
  - Next quantifier and its scope following quantifier
- Scope examples

no quantifier	$P(x) \vee Q(y)$
$(\forall x)$	$(\forall x)[P(x) \rightarrow Q(x)]$
$(\exists y)$	$(\forall x)((\exists y)[P(x, y) \wedge Q(x, y)] \rightarrow R(x))$
$(\forall x)$	$(\forall x)((\exists y)[P(x, y) \wedge Q(x, y)] \rightarrow R(x))$
$(\exists y)$	$(\forall x)(\exists y)Q(x,y)$
$(\forall x)$	$(\forall x)(\exists y)Q(x,y)$

Free variables

- Free variable; variable not in scope of quantifier, e.g., $(\forall x)[Q(x, y) \rightarrow (\exists y)R(x, y)]$, first y is free
- Represents any object in domain
- Truth value of expressions containing free variables may be indeterminate
  - Define $P(x)$ as $x > 0$
  - e.g., $P(y) \wedge P(5)$, y free; truth unknown
  - e.g., $P(y) \vee P(5)$, y free; true because $P(5)$ true

Example interpretation

predicate wff $(\forall x)(\exists y)[S(x, y) \wedge L(y, a)]$ Example 21

Quantifier scope

scope $(\exists y)$ $(\forall x)(\exists y)[S(x, y) \wedge L(y, a)]$

scope $(\forall x)$ $(\forall x)(\exists y)[S(x, y) \wedge L(y, a)]$

Interpretation

domain cities in United States

predicates $S(x, y) = x$ and y are in same state

$L(y, z) =$ names of y and z begin with same letter

constant $a =$ Albuquerque

"For every city in the U. S. x, there exists a city in the U. S. y such that x and y are in the same state and y and Albuquerque start with the same letter."

truth value true iff every state has city beginning with A

Example interpretation

predicate wff $(\exists x)(A(x) \wedge (\forall y)[B(x, y) \rightarrow C(y)])$ Practice 17

Quantifier scope

scope $(\forall y)$ $(\exists x)(A(x) \wedge (\forall y)[B(x, y) \rightarrow C(y)])$

scope $(\exists x)$ $(\exists x)(A(x) \wedge (\forall y)[B(x, y) \rightarrow C(y)])$

Interpretation

domain integers

predicates $A(x) = x > 0$

$B(x, y) = x > y$

$C(y) = y \leq 0$

"There exists integer x such that x is positive and for any integer y, if $x > y$ then $y \leq 0$."

truth value true $x = 1$

Translating English to predicate logic

- Many English sentences can be expressed as predicate wffs
- Translation tips
  - Rewrite sentence in more precise intermediate form
  - Specify domain and predicates first
  - Certain quantifiers and connectives often go together
  - Take care with quantifier scope, logical connectives, and negated sentences

Example translations

domain all objects; $P(x) = x$ is a parrot; $U(x) = x$ is ugly

Sentence “Every parrot is ugly.”
Intermediate “For any thing, if it is a parrot, it is ugly.”
Correct $(\forall x)[P(x) \rightarrow U(x)]$
Incorrect $(\forall x)[P(x) \land U(x)]$ “Every x is an ugly parrot.”
Heuristic $\forall$ and $\rightarrow$ often go together

Sentence “There is an ugly parrot.”
Intermediate “There is at least one thing that is both a parrot and ugly.”
Correct $(\exists x)[P(x) \land U(x)]$
Incorrect $(\exists x)[P(x) \rightarrow U(x)]$ Trivially true by non-parrot x
Heuristic $\exists$ and $\land$ often go together

Example translations

domain all objects
$D(x) = x$ is a dog; $R(x) = x$ is a rabbit; $C(x, y) = x$ chases y

sentence "All dogs chase all rabbits."
intermediate "For any thing, if it is a dog,
then for any other thing, if it is a rabbit,
then the dog chases it."
correct $(\forall x)[D(x) \rightarrow (\forall y)(R(y) \rightarrow C(x, y))]$

sentence "Some dogs chase all rabbits."
intermediate "There is some thing that is a dog,
and for any other thing, if that thing is a rabbit,
then the dog chases it."
correct $(\exists x)[D(x) \wedge (\forall y)(R(y) \rightarrow C(x, y))]$

Example 22

sentence "Only dogs chase rabbits."
intermediate "For any thing, if it is a rabbit, then if any other thing chases it, that thing is a dog."
correct $(\forall y)[R(y) \rightarrow (\forall x)(C(x, y) \rightarrow D(x))]$

sentence "Only dogs chase rabbits."
intermediate "For any two things, if one is a rabbit and other thing chases it, then the other thing is a dog."
correct $(\forall x)(\forall y)[R(y) \wedge C(x, y) \rightarrow D(x)]$

Equivalent wffs

Sentence	"Some dogs chase all rabbits."
Intermediate	"There is some thing that is a dog, and for any other thing, if that thing is a rabbit, then the dog chases it."
wff	$(\exists x)[D(x) \wedge (\forall y)(R(y) \rightarrow C(x, y))]$
Equivalent 1	$(\exists x)[D(x) \wedge (\forall y)([R(y)]' \vee C(x, y))]$
Alternate 1	$(\exists x)[D(x) \wedge (\forall y)(R(y)' \vee C(x, y))]$
Equivalent 2	$(\exists x)(\forall y)[D(x) \wedge (R(y) \rightarrow C(x, y))]$

Translation examples

Interpretation

domain	Set of all people
predicates	$S(x) = x$ is a student
	$I(x) = x$ is intelligent
	$M(x) = x$ likes music

Translations

Sentence	"All students are intelligent."
wff	$(\forall x)[S(x) \rightarrow I(x)]$
Sentence	"Some intelligent students like music."
wff	$(\exists x)[S(x) \wedge I(x) \wedge M(x)]$
Sentence	"Everyone who likes music is a stupid student."
wff	$(\forall x)(M(x) \rightarrow S(x) \wedge I(x)')$
Sentence	"Only intelligent students like music."
wff	$(\forall x)(M(x) \rightarrow S(x) \wedge I(x))$

Practice 18

Translating negations

domain all objects; $B(x) = x$ is beautiful

sentence	"Everything is beautiful."	$(\forall x)B(x)$
incorrect	"Everything is not beautiful."	$(\forall x)[B(x)]'$
correct 1	"It is false that everything is beautiful."	$[(\forall x)B(x)]'$
correct 2	"There is a thing that is not beautiful."	$(\exists x)[B(x)]'$

sentence	"Something is beautiful."	$(\exists x)B(x)$
incorrect	"Not everything is beautiful."	$(\exists x)[B(x)]'$
correct 1	"Nothing is beautiful."	$[(\exists x)B(x)]'$
correct 2	"Everything is not beautiful."	$(\forall x)[B(x)]'$

Reasoning about validity

- Predicate wff validity
  - No algorithm exists, or can exist, to establish predicate wff validity in general.
  - Logical arguments can be given for specific wffs.
- Examples

Example 24

$(\forall x)P(x) \rightarrow (\exists x)P(x)$ valid
If $P(x)$ true for every x, it must be true for at least one x.

$(\forall x)P(x) \rightarrow P(a)$ valid
If $P(x)$ true for every x, it must be true for specific a.

$(\forall x)[P(x) \wedge Q(x)] \leftrightarrow (\forall x)P(x) \wedge (\forall x)Q(x)$ valid
If $P(x)$ and $Q(x)$ true for every x,
then $P(x)$ true for every x and $Q(x)$ true for every x.

$P(x) \to [Q(x) \to P(x)]$ valid

If $P(x)$ true, then $Q(x) \to P(x)$ true regardless of $Q(x)$.

If $P(x)$ false, then main implication trivially true.

$(\exists x)P(x) \to (\forall x)P(x)$ not valid

Counterexample: domain integers, $P(x)$ is x is even;

2 is even, but not all integers are even.

$(\forall x)[P(x) \vee Q(x)] \leftrightarrow (\forall x)P(x) \vee (\forall x)Q(x)$ not valid

Practice 21

Counterexample: domain integers, $P(x)$ is x is even,

$Q(x)$ is x is odd; every integer is even or odd,

but it is false that all integers are even

or all integers are odd.

A	B	$A \to B$
T	T	T
T	F	F
F	T	T
F	F	T

Predicate wff validity

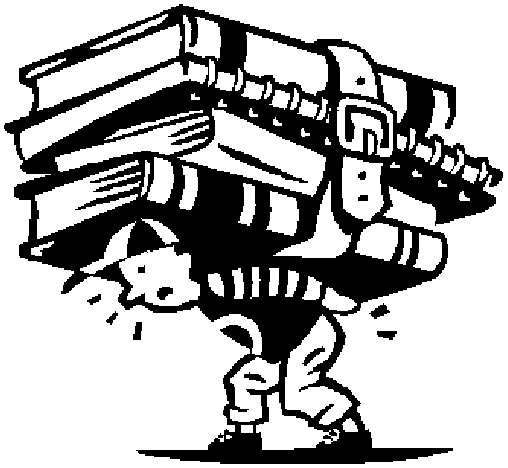
- Predicate wff validity analogous to propositional wff tautology

Table 1.16

Characteristic	Propositional wffs	Predicate wffs
Truth value depends on	Statement letter truth values; logical connectives	Interpretation
Possible truth values	True or false	True, false, or indeterminate
Number of possible truth values	2^n rows in truth table	∞ interpretations
Intrinsic truth	Tautological wff; true for all statement truth values	Valid wff; true for all interpretations
Determining intrinsic truth	Algorithm; construct truth table	No algorithm possible

Section 1.3 homework assignment

See homework list for specific exercises and hints.



1.4 Predicate Logic

1.5 Logic Programming

1.6 Proof of Correctness

References

[Gersting, 2014] J. L. Gersting, *Mathematical Structures for Computer Science: Discrete Mathematics and Its Applications*, *Seventh Edition*, W. H. Freeman and Company, New York NY, 2014.

[Angel, 1997] A. R. Angel and S. R. Porter, *A Survey of Mathematics with Applications*, Fifth Edition, Addison-Wesley, Reading MA, 1997.

[Brookshear, 2000] J. G. Brookshear, *Computer Science: An Overview, Sixth Edition*, Addison-Wesley, Reading MA, 2000.

[Epp, 2011] S. S. Epp, *Discrete Mathematics with Applications, Fourth Edition*, Brooks/Cole, Boston MA, 2011.

[Goldstein, 2005] R. Goldstein, *Incompleteness: The Proof and Paradox of Kurt Gödel*, W. W. Norton & Company, New York NY, 2005.

End

THE UNIVERSITY OF ALABAMA IN HUNTSVILLE