binary expressions:

theorems:

antitheorems:

theorems: represent one kind

antitheorems: represent the other kind

represent statements about the world (natural or constructed, real or imaginary)

theorems: represent one kind

represent true statements

antitheorems: represent the other kind

represent false statements

represent statements about the world (natural or constructed, real or imaginary) represent digital circuits

theorems: represent one kind

represent true statements

represent circuits with high voltage output

antitheorems: represent the other kind

represent false statements

represent circuits with low voltage output

represent statements about the world (natural or constructed, real or imaginary) represent digital circuits represent human behavior

theorems: represent one kind

represent true statements

represent circuits with high voltage output

represent innocent behavior

antitheorems: represent the other kind

represent false statements

represent circuits with low voltage output

represent guilty behavior

### 0 operands $\top \perp$

0 operands  $\top \perp$ 

1 operand  $\neg x$ 

3 operands	if x then y else z fi
2 operands	$x \land y  x \lor y  x \Longrightarrow y  x \Leftarrow y  x \equiv y  x \neq y$
1 operand	$\neg x$
0 operands	ТТ

0 operands	⊤⊥
1 operand	$\neg x$
2 operands	$x \land y  x \lor y  x \Longrightarrow y  x \Leftarrow y  x = y  x \neq y$
3 operands	if x then y else z fi

precedence and parentheses

0 operands $\top \perp$ 1 operand $\neg x$ 2 operands $x \land y \quad x \lor y \quad x \Rightarrow y \quad x \Leftarrow y \quad x = y \quad x \neq y$ 3 operandsif x then y else z fi

precedence and parentheses

associative operators:  $\land \lor = =$ 

 $x \wedge y \wedge z$  means either  $(x \wedge y) \wedge z$  or  $x \wedge (y \wedge z)$ 

 $x \lor y \lor z$  means either  $(x \lor y) \lor z$  or  $x \lor (y \lor z)$ 

0 operands $\top \perp$ 1 operand $\neg x$ 2 operands $x \land y \quad x \lor y \quad x \Rightarrow y \quad x \leftarrow y \quad x = y \quad x \neq y$ 3 operandsif x then y else z fi

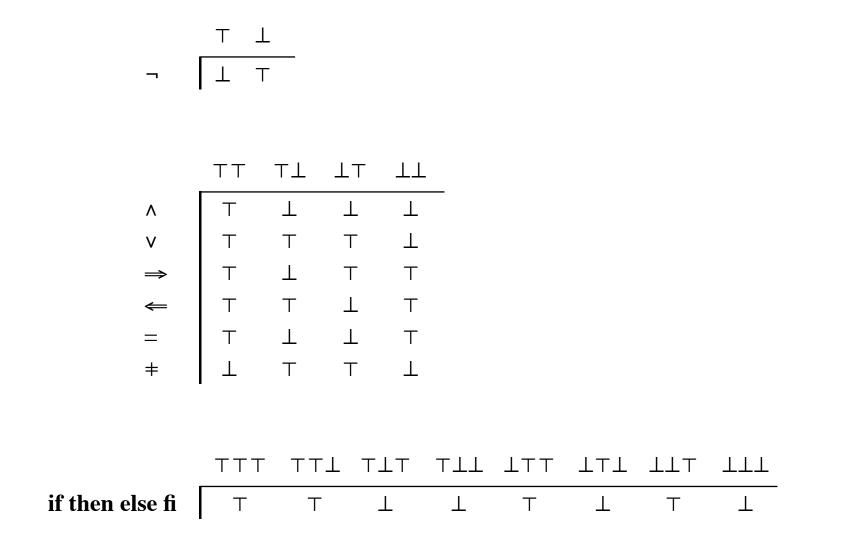
precedence and parentheses

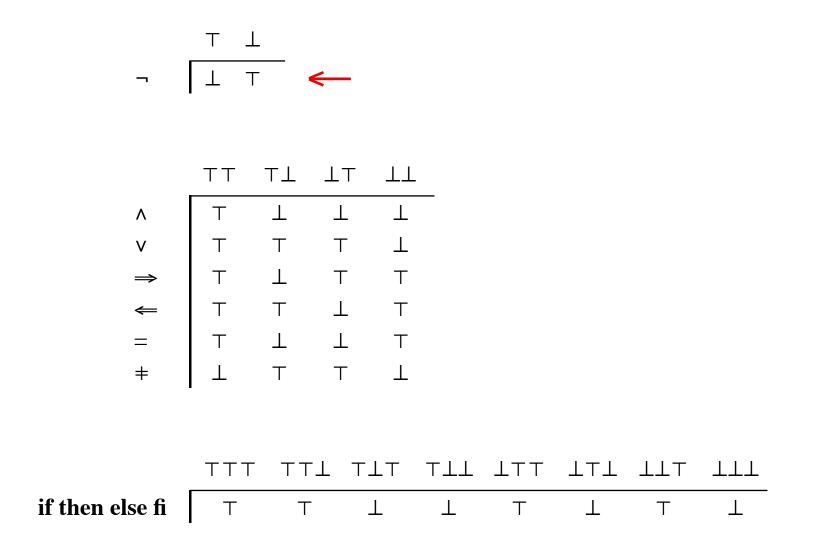
associative operators:  $\land \lor = =$   $x \land y \land z$  means either  $(x \land y) \land z$  or  $x \land (y \land z)$   $x \lor y \lor z$  means either  $(x \lor y) \lor z$  or  $x \lor (y \lor z)$ continuing operators:  $\Rightarrow \Leftarrow = =$  x = y = z means  $x = y \land y = z$  $x \Rightarrow y \Rightarrow z$  means  $(x \Rightarrow y) \land (y \Rightarrow z)$  0 operands $\top \perp$ 1 operand $\neg x$ 2 operands $x \land y \quad x \lor y \quad x \Rightarrow y \quad x \leftarrow y \quad x = y \quad x \neq y$ 3 operandsif x then y else z fi

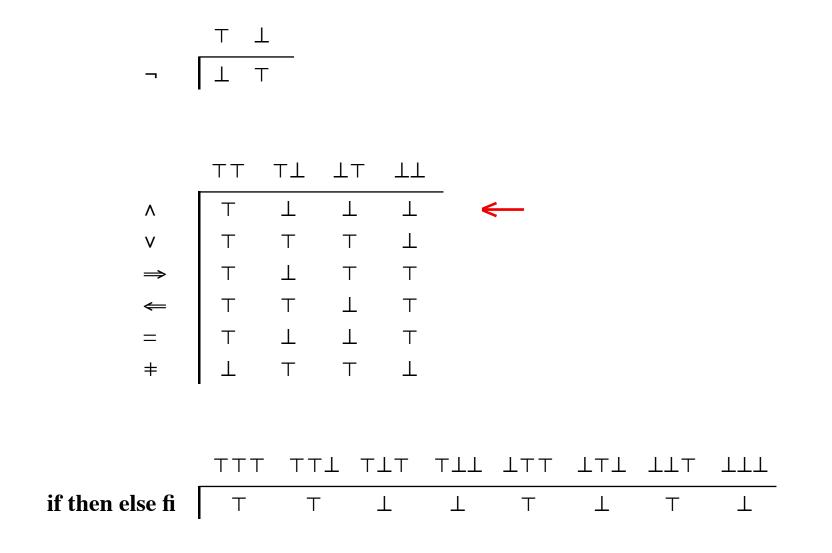
precedence and parentheses

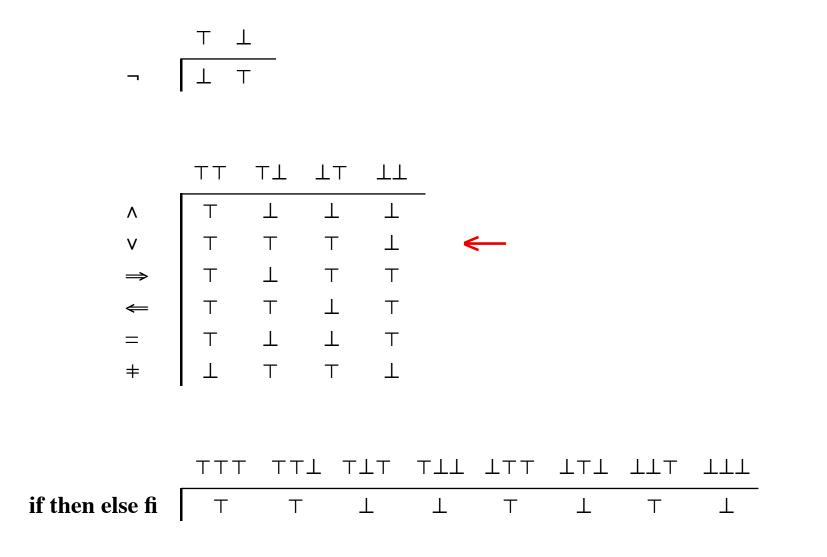
associative operators:  $\land \lor = = =$  $x \wedge y \wedge z$  means either  $(x \wedge y) \wedge z$  or  $x \wedge (y \wedge z)$  $x \lor y \lor z$  means either  $(x \lor y) \lor z$  or  $x \lor (y \lor z)$ continuing operators:  $\Rightarrow \Leftarrow = \pm$ x = y = z means  $x = y \land y = z$  $x \Rightarrow y \Rightarrow z$  means  $(x \Rightarrow y) \land (y \Rightarrow z)$ big operators:  $= \Rightarrow \Leftarrow$ same as  $= \Rightarrow \Leftarrow$  but later precedence  $x = y \Longrightarrow z$  means  $(x = y) \land (y \Longrightarrow z)$ 

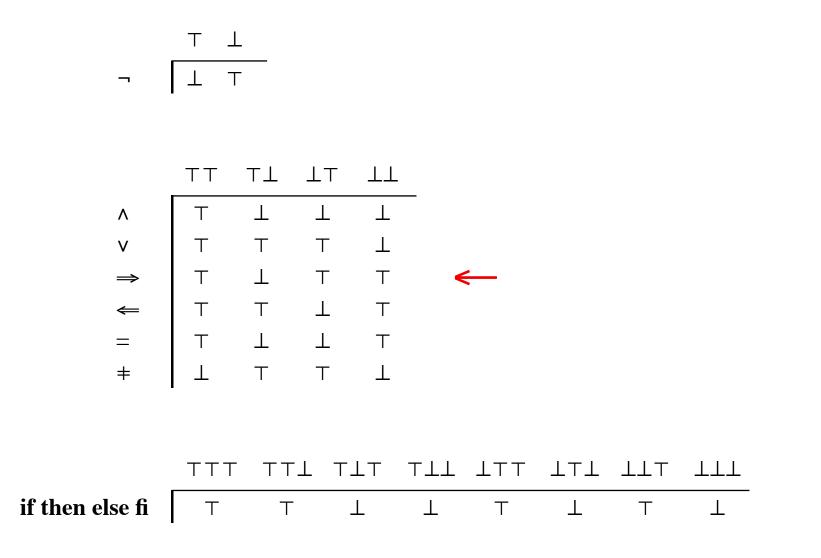
19/85

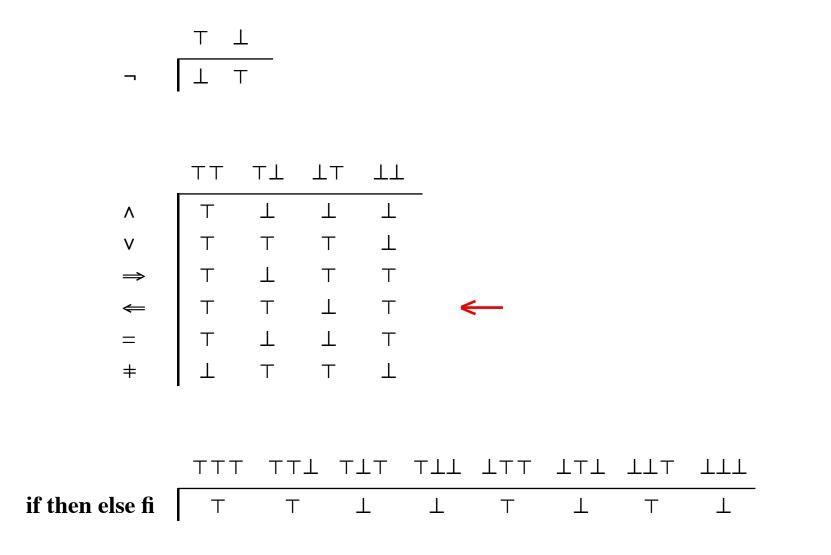


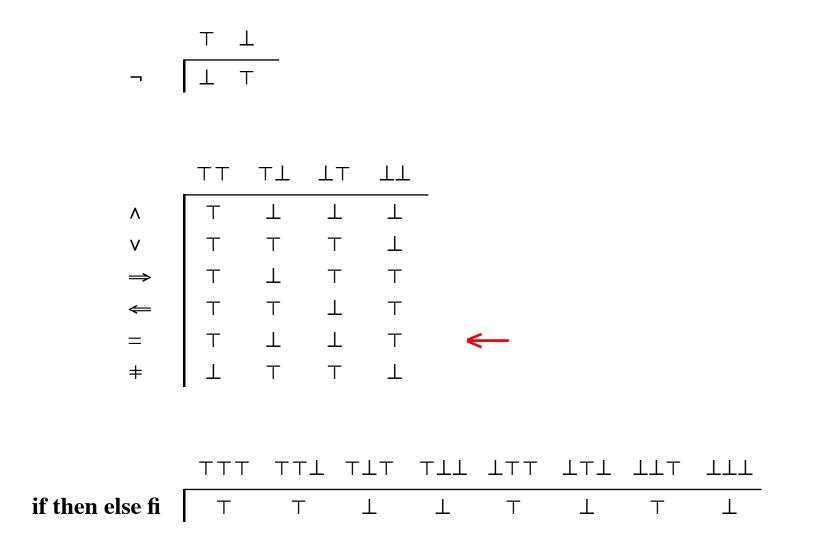




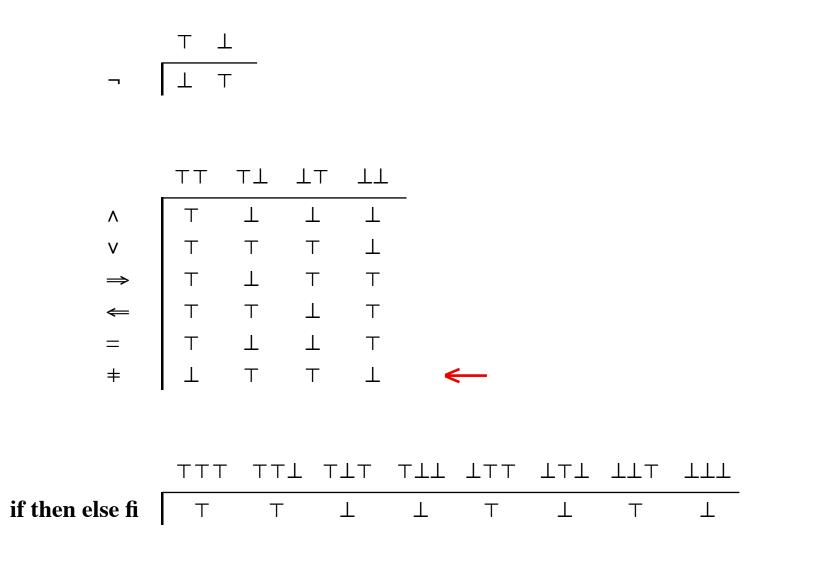


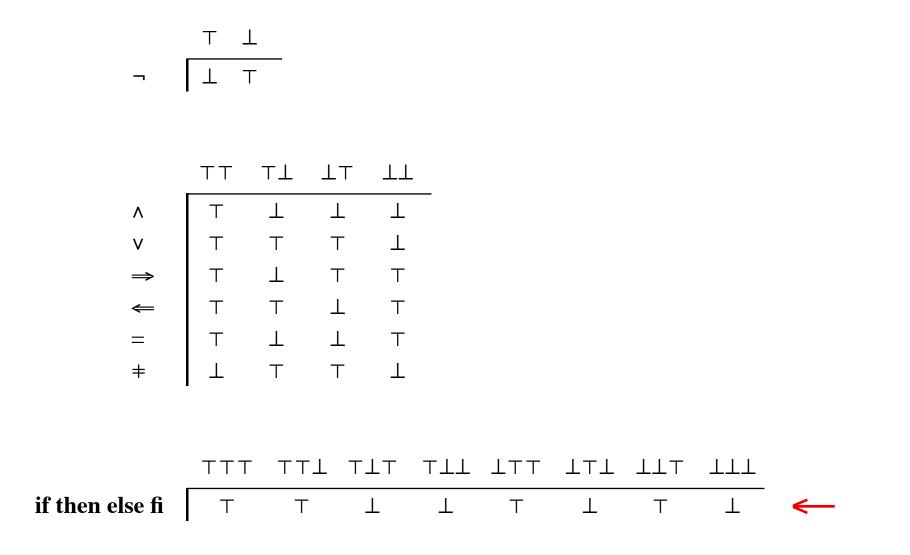






26/85





28/85

• add parentheses to maintain precedence

```
in x \land y replace x by \bot and y by \bot \lor \top result: \bot \land (\bot \lor \top)
```

• add parentheses to maintain precedence

in  $x \land y$  replace x by  $\bot$  and y by  $\bot \lor \top$  result:  $\bot \land (\bot \lor \top)$ 

• every occurrence of a variable must be replaced by the same expression

in  $x \wedge x$  replace x by  $\perp$ 

result:  $\bot \land \bot$ 

• add parentheses to maintain precedence

in  $x \wedge y$  replace x by  $\perp$  and y by  $\perp \lor \top$  result:  $\perp \land (\perp \lor \top)$ 

• every occurrence of a variable must be replaced by the same expression

in  $x \wedge x$  replace x by  $\perp$  result:  $\perp \wedge \perp$ 

different variables can be replaced by the same expression or different expressions
in x ∧ y replace x by ⊥ and y by ⊥ result: ⊥ ∧ ⊥

• add parentheses to maintain precedence

in  $x \wedge y$  replace x by  $\perp$  and y by  $\perp \lor \top$  result:  $\perp \land (\perp \lor \top)$ 

• every occurrence of a variable must be replaced by the same expression

in  $x \wedge x$  replace x by  $\perp$  result:  $\perp \wedge \perp$ 

• different variables can be replaced by the same expression or different expressions

in  $x \wedge y$  replace x by  $\perp$  and y by  $\perp$  result:  $\perp \wedge \perp$ in  $x \wedge y$  replace x by  $\top$  and y by  $\perp$  result:  $\top \wedge \perp$ 

# new binary expressions

(the grass is green)

(the sky is green)

(there is life elsewhere in the universe)

(intelligent messages are coming from space)

# new binary expressions

(the grass is green)

(the sky is green)

(there is life elsewhere in the universe)

(intelligent messages are coming from space)

1 + 1 = 20 / 0 = 5

### new binary expressions

(the grass is green)

(the sky is green)

(there is life elsewhere in the universe)

(intelligent messages are coming from space)

1 + 1 = 20 / 0 = 5

**consistent**: no binary expression is both a theorem and an antitheorem (no overclassified expressions)

#### new binary expressions

(the grass is green)

(the sky is green)

(there is life elsewhere in the universe)

(intelligent messages are coming from space)

1 + 1 = 20 / 0 = 5

consistent: no binary expression is both a theorem and an antitheorem

(no overclassified expressions)

**complete**: every fully instantiated binary expression is either a theorem or an antitheorem (no unclassified expressions)

**Axiom Rule** If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

**Axiom Rule** If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

x+y = y+x is a mathematical expression

Axiom Rule If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

x+y = y+x is a mathematical expression

represents a truth in an application such that

when you put quantities together, the total quantity does not depend on the order in which you put them together

Axiom Rule If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

x+y = y+x is a mathematical expression

represents a truth in an application such that

when you put quantities together, the total quantity does not depend on the order in which you put them together

is an axiom

Axiom Rule If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

x+y = y+x is a mathematical expression

represents a truth in an application such that

when you put quantities together, the total quantity does not depend on the order in which you put them together

is an axiom

is a theorem

Axiom Rule If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

x+y = y+x is a mathematical expression

represents a truth in an application such that

when you put quantities together, the total quantity does not depend on the order in which you put them together

is an axiom

is a theorem

is equivalent to  $\top$ 

Axiom Rule If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

x+y = y+x is a mathematical expression

represents a truth in an application such that

when you put quantities together, the total quantity does not depend on the order in which you put them together

is an axiom

is a theorem

is equivalent to  $\top$ 

x+y = y+x is true (not really)

**Axiom Rule** If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

**Axiom Rule** If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

axiom: T

antiaxiom:  $\bot$ 

**Axiom Rule** If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

axiom: ⊤

antiaxiom:  $\bot$ 

axiom: (the grass is green)

antiaxiom: (the sky is green)

**Axiom Rule** If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

axiom:  $\top$ 

antiaxiom:  $\bot$ 

axiom: (the grass is green)

antiaxiom: (the sky is green)

axiom: (intelligent messages are coming from space)

 $\Rightarrow$  (there is life elsewhere in the universe)

**Axiom Rule** If a binary expression is an axiom, then it is a theorem.

If a binary expression is an antiaxiom, then it is an antitheorem.

axiom:  $\top$ 

antiaxiom:  $\bot$ 

axiom: (the grass is green)

antiaxiom: (the sky is green)

axiom: (intelligent messages are coming from space)

 $\Rightarrow$  (there is life elsewhere in the universe)

**Evaluation Rule** If all the binary subexpressions of a binary expression are classified,

then it is classified according to the theorem tables.

**Completion Rule** If a binary expression contains unclassified binary subexpressions,

and all ways of classifying them place it in the same class, then it is in that class.

**Completion Rule** If a binary expression contains unclassified binary subexpressions,

and all ways of classifying them place it in the same class, then it is in that class.

theorem: (there is life elsewhere in the universe)  $\vee \top$ 

**Completion Rule** If a binary expression contains unclassified binary subexpressions,

and all ways of classifying them place it in the same class, then it is in that class.

theorem: (there is life elsewhere in the universe)  $\vee \top$ 

theorem: (there is life elsewhere in the universe)

 $\vee$   $\neg$  (there is life elsewhere in the universe)

**Completion Rule** If a binary expression contains unclassified binary subexpressions,

and all ways of classifying them place it in the same class, then it is in that class.

theorem: (there is life elsewhere in the universe)  $\vee \top$ 

theorem: (there is life elsewhere in the universe)

 $\vee$   $\neg$  (there is life elsewhere in the universe)

antitheorem: (there is life elsewhere in the universe)

 $\wedge$   $\neg$  (there is life elsewhere in the universe)

**Consistency Rule** If a classified binary expression contains binary subexpressions, and only one way of classifying them is consistent, then they are classified that way.

**Consistency Rule** If a classified binary expression contains binary subexpressions, and only one way of classifying them is consistent, then they are classified that way.

We are given that x and  $x \rightarrow y$  are theorems. What is y?

**Consistency Rule** If a classified binary expression contains binary subexpressions, and only one way of classifying them is consistent, then they are classified that way.

We are given that x and  $x \rightarrow y$  are theorems. What is y?

If y were an antitheorem, then by the Evaluation Rule,  $x \rightarrow y$  would be an antitheorem.

That would be inconsistent. So y is a theorem.

**Consistency Rule** If a classified binary expression contains binary subexpressions, and only one way of classifying them is consistent, then they are classified that way.

We are given that x and  $x \rightarrow y$  are theorems. What is y?

If y were an antitheorem, then by the Evaluation Rule,  $x \rightarrow y$  would be an antitheorem.

That would be inconsistent. So y is a theorem.

We are given that  $\neg x$  is a theorem. What is x?

**Consistency Rule** If a classified binary expression contains binary subexpressions, and only one way of classifying them is consistent, then they are classified that way.

We are given that x and  $x \rightarrow y$  are theorems. What is y?

If y were an antitheorem, then by the Evaluation Rule,  $x \Rightarrow y$  would be an antitheorem. That would be inconsistent. So y is a theorem.

We are given that  $\neg x$  is a theorem. What is x?

If x were a theorem, then by the Evaluation Rule,  $\neg x$  would be an antitheorem.

That would be inconsistent. So x is an antitheorem.

**Consistency Rule** If a classified binary expression contains binary subexpressions, and only one way of classifying them is consistent, then they are classified that way.

We are given that x and  $x \rightarrow y$  are theorems. What is y?

If y were an antitheorem, then by the Evaluation Rule,  $x \Rightarrow y$  would be an antitheorem. That would be inconsistent. So y is a theorem.

We are given that  $\neg x$  is a theorem. What is x?

If x were a theorem, then by the Evaluation Rule,  $\neg x$  would be an antitheorem.

That would be inconsistent. So x is an antitheorem.

No need to talk about antiaxioms and antitheorems.

Transparency Rule A binary expression does not gain, lose, or change classification

Transparency Rule A binary expression does not gain, lose, or change classification

$$\neg x \Rightarrow (x \land x) \lor y = \top$$

Transparency Rule A binary expression does not gain, lose, or change classification

$$\neg x \Rightarrow (x \land x) \lor y = \top$$
$$\Box$$
$$\neg x \Rightarrow (x \land x) \lor y = (y \lor \top)$$

Transparency Rule A binary expression does not gain, lose, or change classification

$$\neg x \Rightarrow (x \land x) \lor y = \top$$
$$\neg x \Rightarrow (x \land x) \lor y = (y \lor \top)$$
$$\neg x \Rightarrow x \lor y = (y \lor \top)$$

Instance Rule If a binary expression is classified,

then all its instances have that same classification.

Instance Rule If a binary expression is classified,

then all its instances have that same classification.

axiom: x = x

Instance Rule If a binary expression is classified,

then all its instances have that same classification.

- axiom: x = x
- theorem: x = x

#### Instance Rule If a binary expression is classified,

then all its instances have that same classification.

axiom:	x = x
theorem:	x = x
theorem:	$(\top = \bot \lor \bot) = (\top = \bot \lor \bot)$

#### Instance Rule If a binary expression is classified,

then all its instances have that same classification.

axiom: x = xtheorem: x = xtheorem:  $(\top = \bot \lor \bot) = (\top = \bot \lor \bot)$ theorem: (intelligent messages are coming from space)

= (intelligent messages are coming from space)

#### Instance Rule If a binary expression is classified,

then all its instances have that same classification.

axiom:x = xtheorem:x = xtheorem: $(\top = \bot \lor \bot) = (\top = \bot \lor \bot)$ theorem:(intelligent messages are coming from space)

= (intelligent messages are coming from space)

Classical Logic:	all six rules
Constructive Logic:	not Completion Rule
<b>Evaluation Logic:</b>	neither Consistency Rule nor Completion Rule

# **Expression and Proof Format**

 $a \wedge b \vee c$ 

# **Expression and Proof Format**

 $a \wedge b \vee c$  **NOT**  $a \wedge b \vee c$ 

# **Expression and Proof Format**



- ( first part
- $\land$  second part )



- ( first part
- $\land$  second part )

C and Java convention

while (something) {

various lines

in the body

of the loop

}



- ( first part
- $\land$  second part )

- $a \wedge b \vee c$  **NOT**  $a \wedge b \vee c$
- ( first part
- $\land$  second part )

first part

*= second part* 

- $a \wedge b \vee c$  **NOT**  $a \wedge b \vee c$
- ( first part
- ∧ *second part* )

### first part

- *= second part*
- expression0
- = *expression1*
- = *expression2*
- *= expression3*

- $a \wedge b \vee c$  **NOT**  $a \wedge b \vee c$
- ( first part
- ∧ *second part* )

#### first part

- *= second part*
- expression0
- = *expression1* means
- *= expression2*
- *= expression3*

#### expression0=expression1

- ∧ *expression1=expression2*
- ∧ expression2=expression3

- $a \wedge b \vee c$  **NOT**  $a \wedge b \vee c$
- ( first part
- ∧ *second part* )

### first part

- *= second part*
- expression0
- = *expression1*
- = *expression2*
- *= expression3*

- $a \wedge b \vee c$  **NOT**  $a \wedge b \vee c$
- ( first part
- ∧ second part )

### first part

*= second part* 

	expression0	hint0
=	expression1	hint1
=	expression2	hint2

*= expression3* 

Prove  $a \land b \Rightarrow c = a \Rightarrow (b \Rightarrow c)$ 

Prove 
$$a \land b \Rightarrow c = a \Rightarrow (b \Rightarrow c)$$

 $a \Rightarrow (b \Rightarrow c)$ 

=

	$a \land b \Rightarrow c$	Material Implication
=	$\neg(a \land b) \lor c$	Duality
=	$\neg a \lor \neg b \lor c$	Material Implication
=	$a \Rightarrow \neg b \lor c$	Material Implication

Prove 
$$a \land b \Rightarrow c = a \Rightarrow (b \Rightarrow c)$$

	$a \land b \Rightarrow c$	Material Implication
=	$\neg(a \land b) \lor c$	Duality
=	$\neg a \lor \neg b \lor c$	Material Implication
=	$a \Rightarrow \neg b \lor c$	Material Implication
=	$a \Rightarrow (b \Rightarrow c)$	

Material Implication:  $a \Rightarrow b \equiv \neg a \lor b$ 

Prove 
$$a \land b \Rightarrow c = a \Rightarrow (b \Rightarrow c)$$

	$a \land b \Rightarrow c$	Material Implication
=	$\neg(a \land b) \lor c$	Duality
=	$\neg a \lor \neg b \lor c$	Material Implication
=	$a \Rightarrow \neg b \lor c$	Material Implication

 $= a \Rightarrow (b \Rightarrow c)$ 

Material Implication: Instance of Material Implication:  $a \Rightarrow b = \neg a \lor b$  $a \land b \Rightarrow c = \neg (a \land b) \lor c$ 

83/85

Prove 
$$a \land b \Rightarrow c = a \Rightarrow (b \Rightarrow c)$$

 $a \Rightarrow (b \Rightarrow c)$ 

=

$a \land b \Rightarrow c$	Material Implication
$\neg(a \land b) \lor c$	Duality
$\neg a \lor \neg b \lor c$	Material Implication
$a \Rightarrow \neg b \lor c$	Material Implication
	*

Prove 
$$a \land b \Rightarrow c = a \Rightarrow (b \Rightarrow c)$$

	$a \land b \Rightarrow c$	Material Implication
=	$\neg(a \land b) \lor c$	Duality
=	$\neg a \lor \neg b \lor c$	Material Implication
=	$a \Rightarrow \neg b \lor c$	Material Implication
=	$a \Rightarrow (b \Rightarrow c)$	

$$(a \land b \Rightarrow c = a \Rightarrow (b \Rightarrow c))$$
Material Implication 3 times  
$$= (\neg (a \land b) \lor c = \neg a \lor (\neg b \lor c))$$
Duality  
$$= (\neg a \lor \neg b \lor c = \neg a \lor \neg b \lor c)$$
Reflexivity of =  
$$= \top$$