

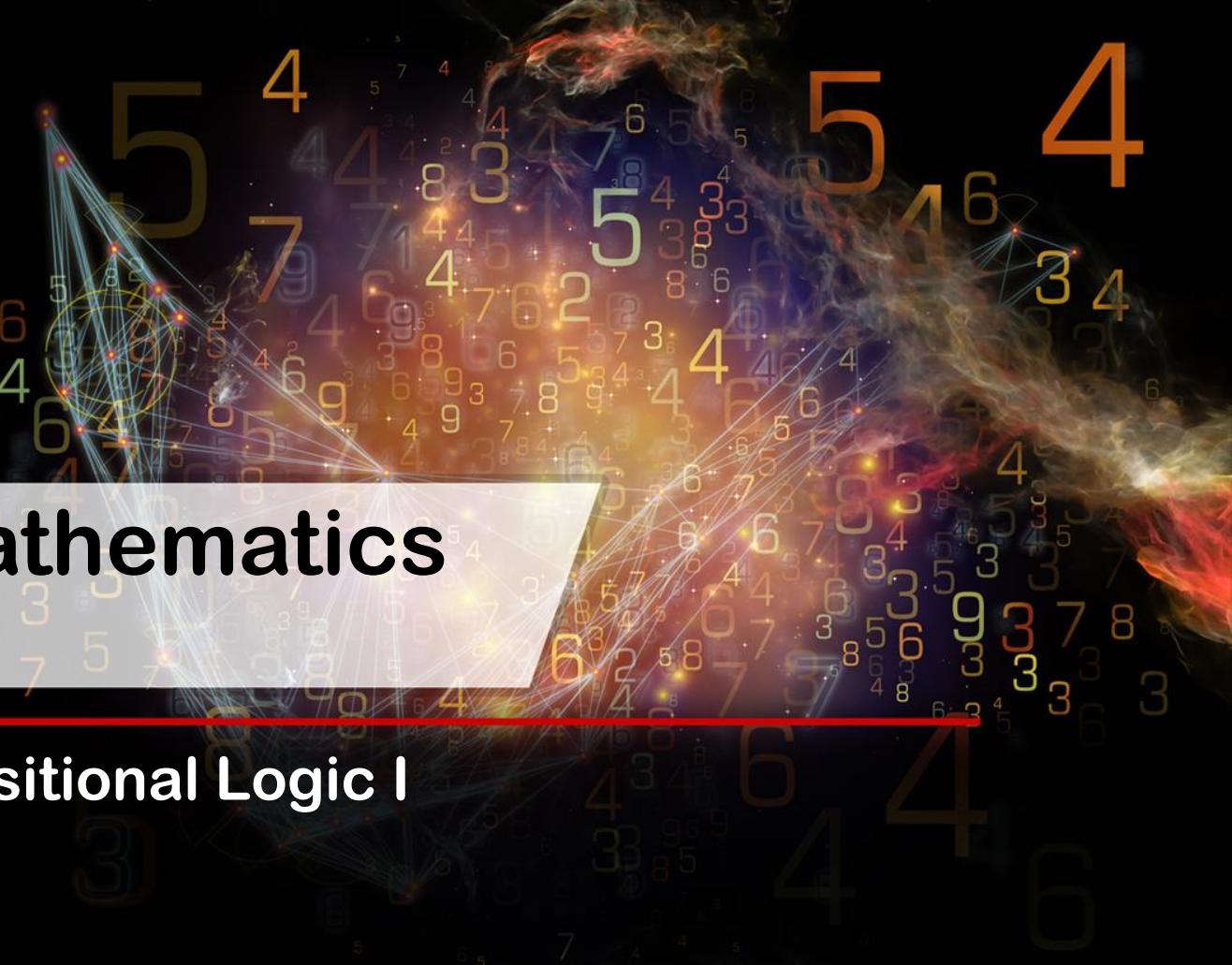


NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE

Discrete Mathematics

MH1812

Topic 2.1 - Propositional Logic I
Dr. Gary Greaves



Topic Overview



What's in store...

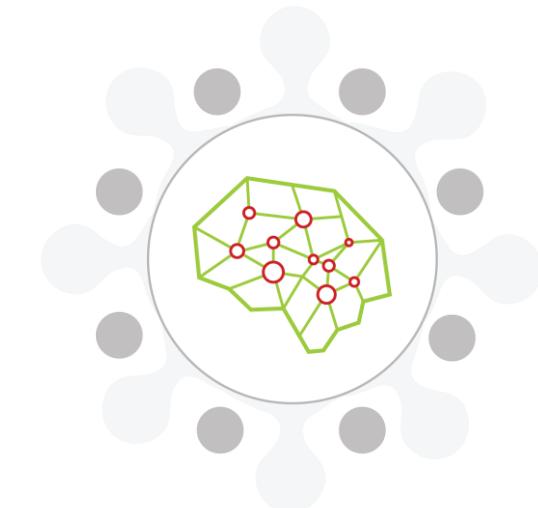
P roposition and Paradox

L ogical Operators

D e Morgan's Laws

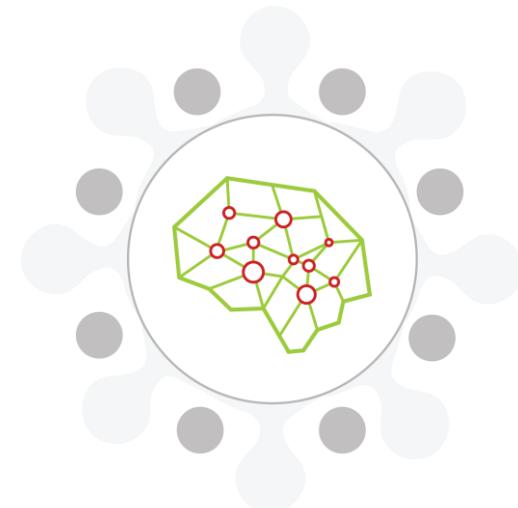
C ontradiction and Tautology

E quivalent Expressions

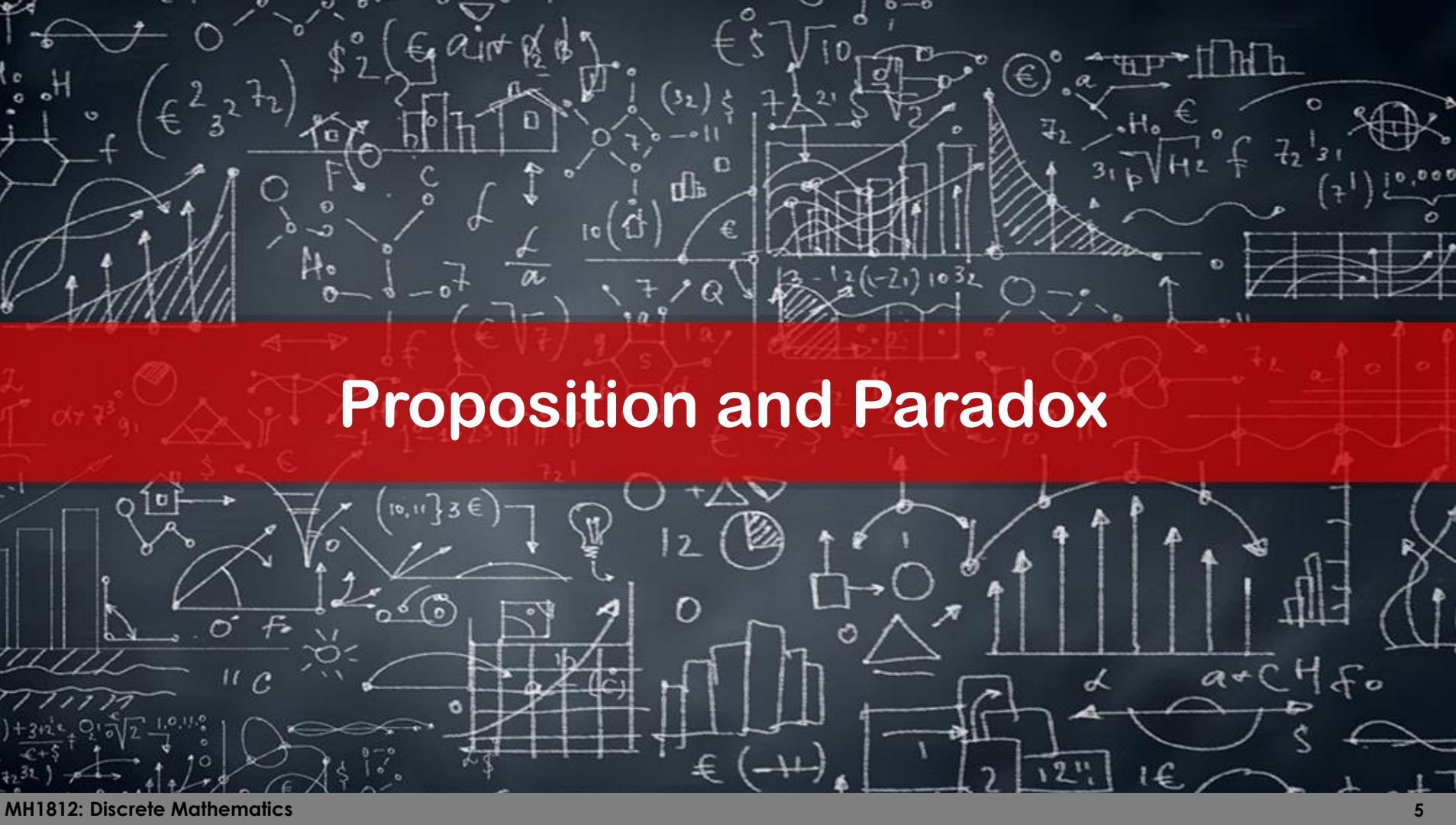


By the end of this lesson, you should be able to...

- Explain what is a proposition and a paradox.
- Use logical operators to combine statements.
- Apply De Morgan's Laws.
- Explain what is a contradiction and a tautology.
- Identify equivalent expressions.
- Demonstrate that two expressions are equivalent.



Proposition and Paradox



Proposition and Paradox: Logic

- Accepted rules for making **precise** statements
- Logic for computer science:
 - Programming
 - Artificial intelligence
 - Logic circuits
 - Database
- Logic:
 - Represents **knowledge precisely**
 - Helps to **extract information** (inference)



Proposition and Paradox: Proposition



A **proposition** is a declarative statement that is either **true** or **false**.



Examples of propositions

- “ $1 + 1 = 2$ ”... **True**
- “ $1 + 1 > 3$ ”... **False**
- “Singapore is in Europe.”... **False**

```
gap> (5>3);  
true  
gap> (1>3);  
false  
gap>
```

Proposition and Paradox: Proposition



A **proposition** is a declarative statement that is either **true** or **false**.



Examples that are not propositions

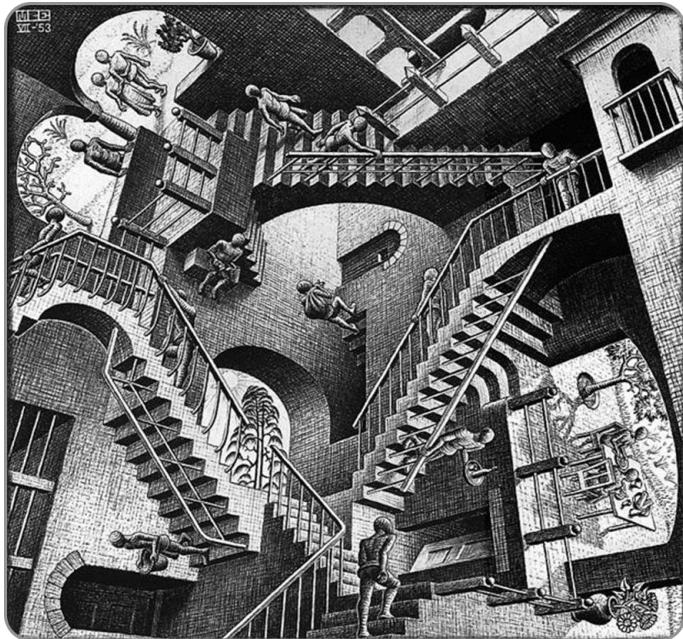
- “ $1 + 1 > x$ ”... **X**
- “What a great book!”... **X**
- “Is Singapore in Asia?”... **X**

Proposition and Paradox: Paradox



A declarative statement that **cannot** be assigned a **truth value** is called a **paradox**.

- A paradox is not a proposition.
- E.g., the liar paradox:
“This statement is false”.



Relativity Lattice (M.C. Escher)

Logical Operators



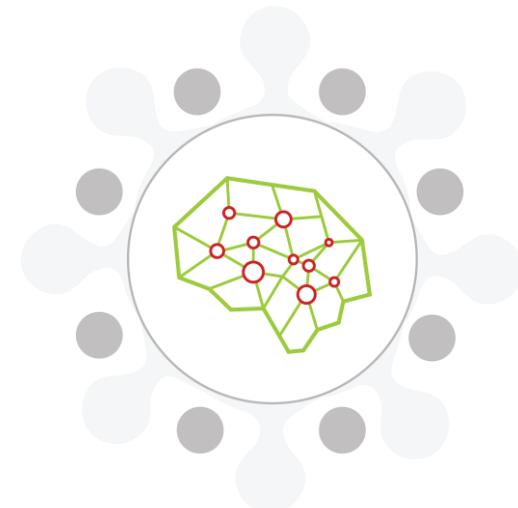
Logical Operators: Symbolic Logic

- Use **symbols** to represent statements (both have the **same truth values**)
- Use **logical operators** to combine statements:

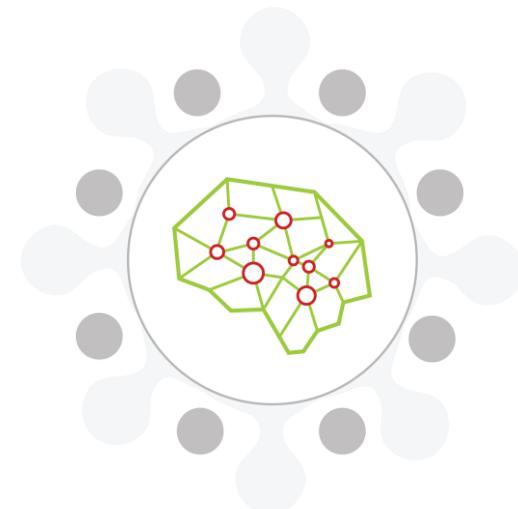
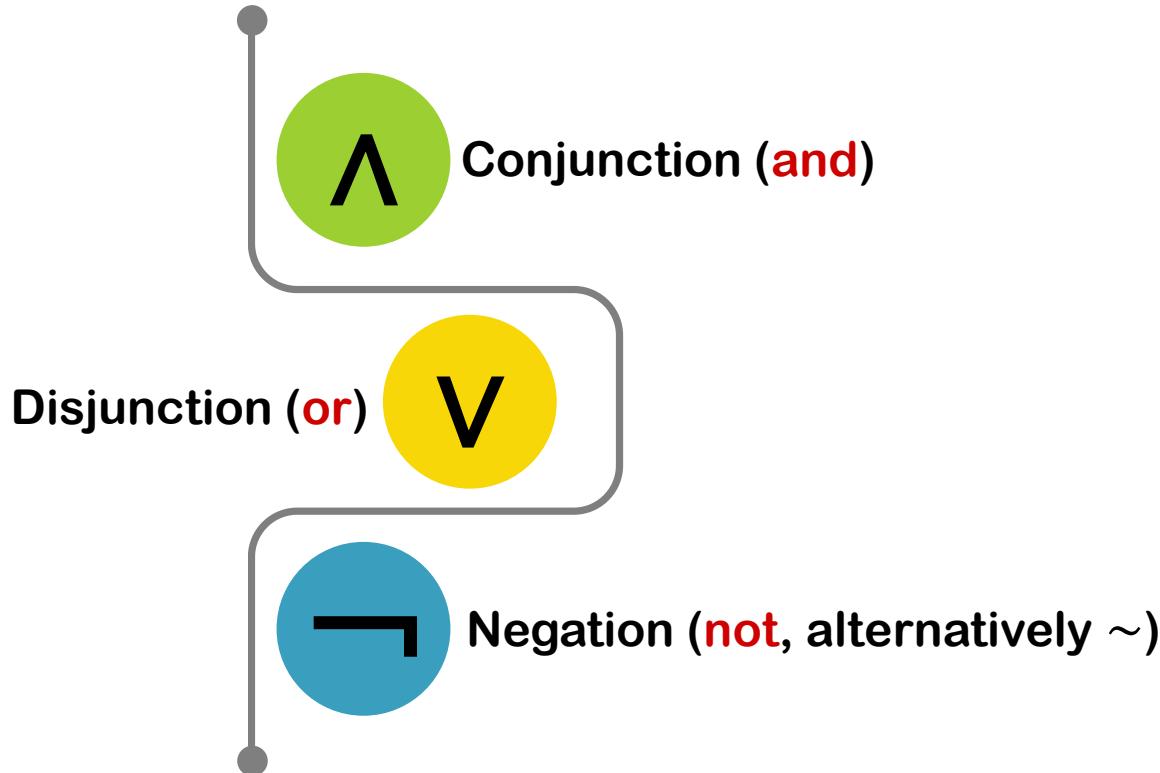
Compound
Propositions

=

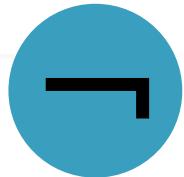
Propositions Combined
with Logical Operator(s)



Logical Operators: Three Basic Operators



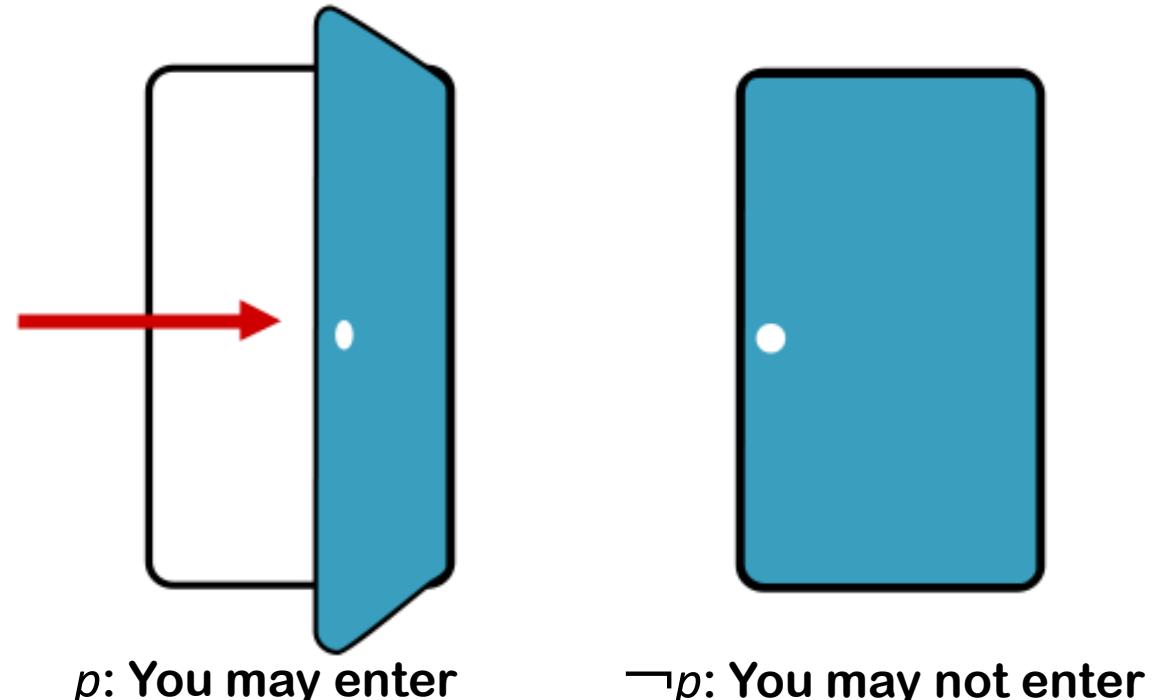
Logical Operators: Negation



- Negation (**not**) of p : $\neg p$ ($\sim p$ is also used)

p	$\neg p$
T	F
F	T

Truth Table



p : You may enter

$\neg p$: You may not enter

Logical Operators: Disjunction



- Disjunction (**or**) of p with q : $p \vee q$

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

True when “at least one” of them is true

Truth Table

$$p \vee q \equiv q \vee p$$

i.e., operator \vee commutes

Means “equivalent”

```
gap>
gap> (5>3) or (1>5);
true
gap>
```

Logical Operators: Conjunction



- Conjunction (**and**) of p with q : $p \wedge q$

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

True only when “both” of them are true

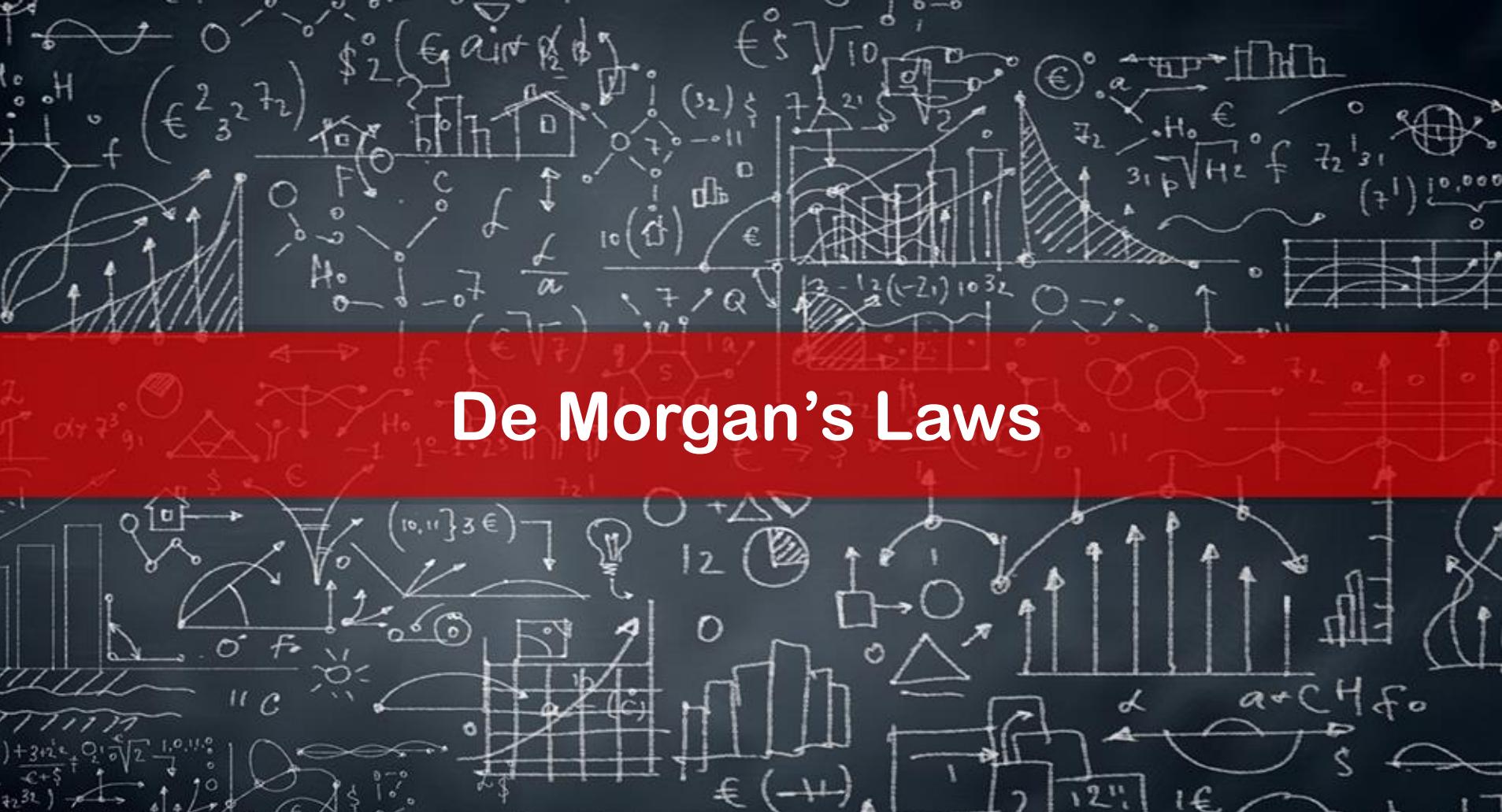
Truth Table

\wedge is also commutative:

$$p \wedge q \equiv q \wedge p$$

```
gap> (5>3) and (7>5);  
true  
gap>  
gap>  
gap> (5>3) and (1>5);  
false
```

De Morgan's Laws



De Morgan's Laws: Definition



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

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

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

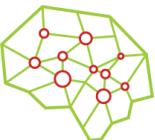


Augustus De Morgan
(1806 - 1871)

Augustus De Morgan by Sophia Elizabeth De Morgan under WikiCommons (PD-US)

Contradiction and Tautology

Contradiction and Tautology: Definition



A compound proposition that is always false is called a **contradiction**.



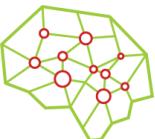
Example

This course is easy “and” this course is not easy.

$$p \wedge (\neg p) \equiv F$$

p	$\neg p$	$p \wedge \neg p$
T	F	F
F	T	F

Contradiction and Tautology: Definition



A compound proposition that always gives a true value is called a **tautology**.



Example

$$p \vee (\neg p) \equiv T$$

p	$\neg p$	$p \vee \neg p$
T	F	T
F	T	T

Always
true!

Equivalent Expressions

Equivalent Expressions: Bob and Alice

1. Alice is not married but Bob is not single.

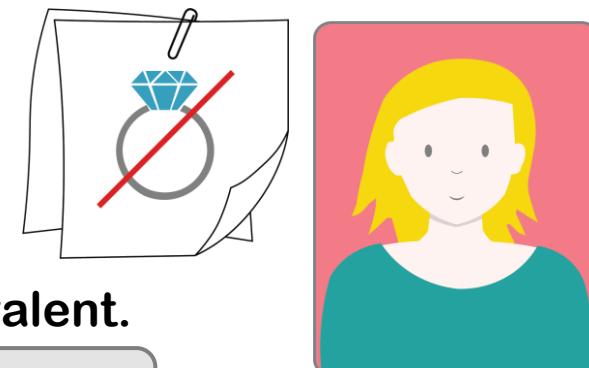
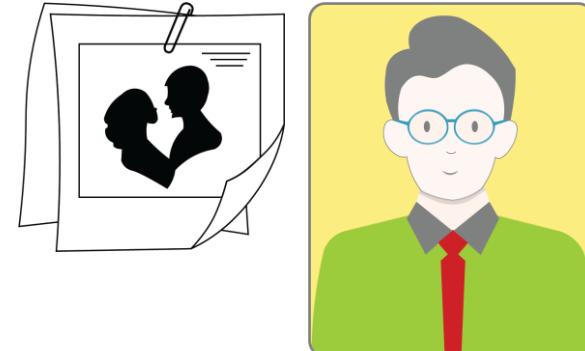
$$\neg h \wedge \neg b$$

2. Bob is not single and Alice is not married.

$$\neg b \wedge \neg h$$

3. Neither Bob is single nor Alice is married.

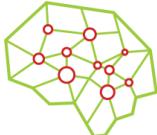
$$\neg(b \vee h)$$



These three statements are equivalent.

$$\neg h \wedge \neg b \equiv \neg b \wedge \neg h \equiv \neg(b \vee h)$$

Equivalent Expressions: The Statements



These three statements are equivalent:

$$\neg h \wedge \neg b \equiv \neg b \wedge \neg h \equiv \neg(b \vee h)$$

b	h	$\neg b$	$\neg h$	$b \vee h$	$\neg h \wedge \neg b$	$\neg b \wedge \neg h$	$\neg(b \vee h)$
T	T	F	F	T	F	F	F
T	F	F	T	T	F	F	F
F	T	T	F	T	F	F	F
F	F	T	T	F	T	T	T

Topic Summary



Let's recap...

- We have covered:
 - Proposition (Compound Propositions)
 - Paradox
 - Contradiction
 - Tautology
 - Equivalent Expressions
- Basic logical operators (and De Morgan's laws):
 - Negation
 - Conjunction
 - Disjunction



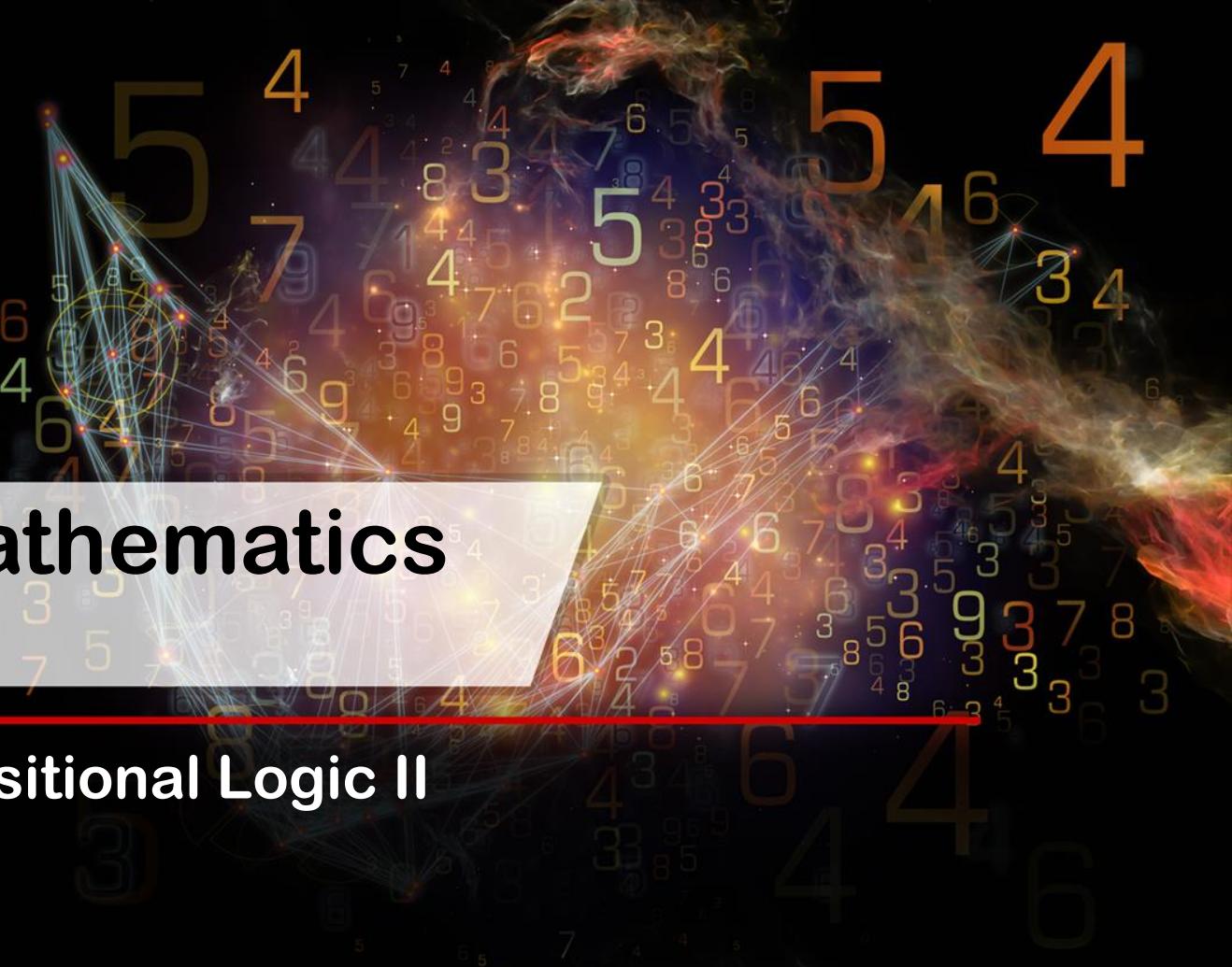


NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE

Discrete Mathematics

MH1812

Topic 2.2 - Propositional Logic II
Dr. Gary Greaves



Topic Overview



What's in store...

L

ogical Equivalence Laws

C

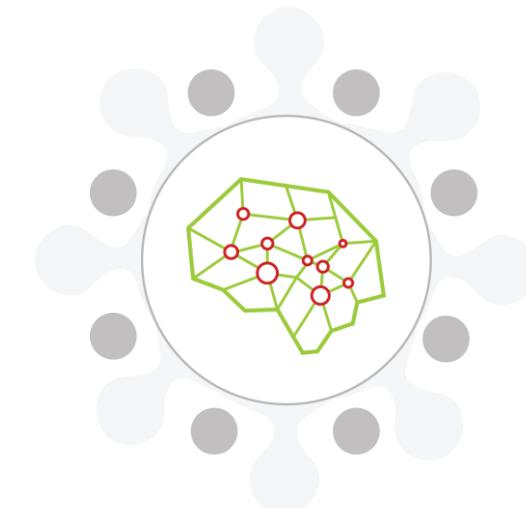
onditional Operator

B

iconditional Operator

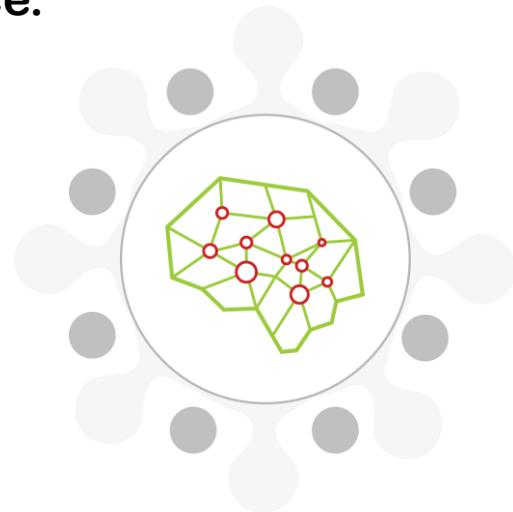
O

perator Precedence



By the end of this lesson, you should be able to...

- Prove equivalence using logical equivalence laws.
- Use the conditional operator to combine propositions.
- Use the biconditional operator to combine propositions.
- Evaluate logical expressions using operator precedence.



Logical Equivalence Laws

Logical Equivalence Laws: Already Seen

- Useful laws to **transform** one logical expression to an equivalent one

Axioms

$T \equiv$ Tautology

$C \equiv$ Contradiction

$$\neg T \equiv F$$

$$\neg F \equiv T$$

$$\neg T \equiv C \equiv F$$

$$\neg C \equiv T \equiv T$$

De Morgan

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

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

Commutativity

$$p \wedge q \equiv q \wedge p$$

$$p \vee q \equiv q \vee p$$

Logical Equivalence Laws: More Laws

Double Negation

$$\neg(\neg p) \equiv p$$

Absorption

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

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

Idempotent

$$p \wedge p \equiv p$$

$$p \vee p \equiv p$$

Logical Equivalence Laws: Distributive Law

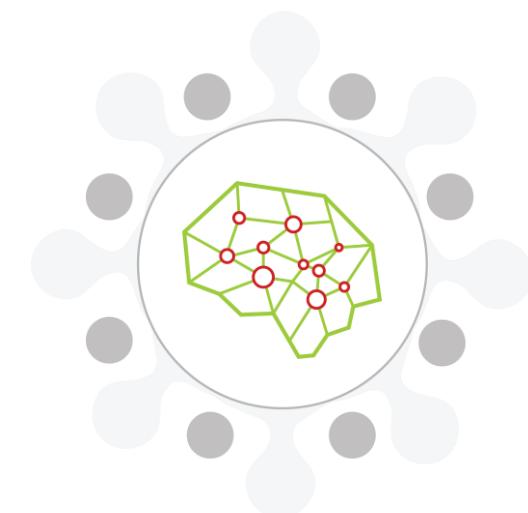
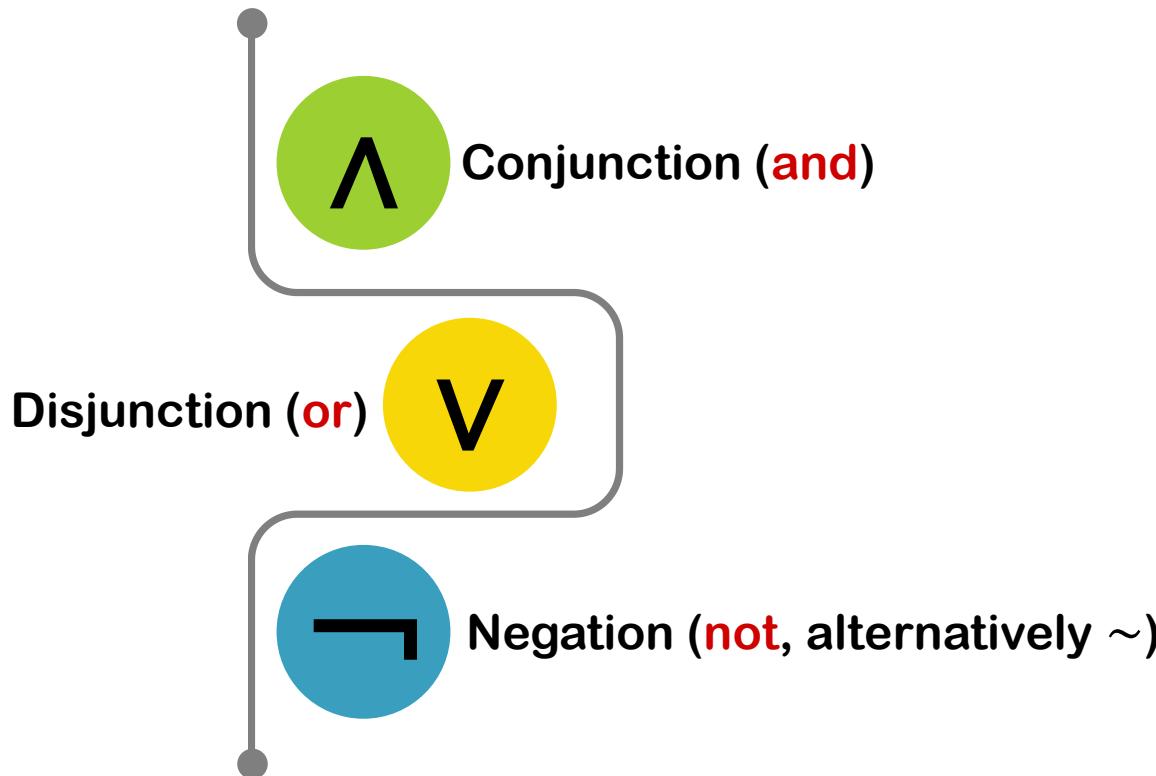
Distributivity

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

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

Conditional Operator

Conditional Operator: Known Operators



Conditional Operator: If Then



If p then q : $p \rightarrow q$.

By definition, when p is false, $p \rightarrow q$ is true.

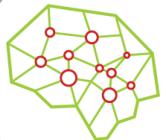
This is called **vacuously true** or **true by default**.

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

```
gap>  
gap> a:=10;; if (a>5) then Print("yes"); fi;  
yes  
gap> a:=1;; if (a>5) then Print("yes"); fi;  
gap>  
gap>
```

Not really the same!

Conditional Operator: Conversion Theorem



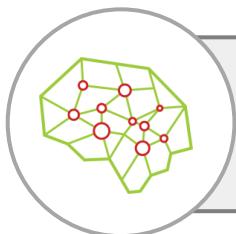
Theorem: $p \rightarrow q \equiv \neg p \vee q$

Proof

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

Conditional Operator: Converse, Inverse, Contrapositive

Statement	$p \rightarrow q$
Converse	$q \rightarrow p$
Inverse	$\neg p \rightarrow \neg q$
Contrapositive	$\neg q \rightarrow \neg p$



Theorem: $\neg q \rightarrow \neg p \equiv p \rightarrow q$

Proof

$$\begin{aligned}\neg q \rightarrow \neg p \\ \equiv \neg(\neg q) \vee \neg p \\ \equiv q \vee \neg p \\ \equiv \neg p \vee q \\ \equiv p \rightarrow q\end{aligned}$$

Conditional Operator: Only If

- $p \text{ only if } q \triangleq \neg q \rightarrow \neg p$
- $\neg q \rightarrow \neg p$ is the **contrapositive** of $p \rightarrow q$
- $(\text{If not } q \text{ then not } p) \equiv (p \rightarrow q)$ (why?)



Example

“Bob pays taxes **only if** his income $\geq \$1000$ ”

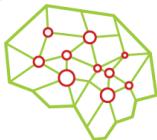
\triangleq “if Bob’s income $< \$1000$ then he does not pay taxes”

\equiv “if Bob pays tax then his income $\geq \$1000$ ”



Bob

Conditional Operator: Sufficient and Necessary Conditions



When $p \rightarrow q$, p is called a **sufficient condition** for q ,
 q is a **necessary condition** for p .

- Being an apple is a **sufficient condition** for being a fruit.
≡ If it is an apple then it must be a fruit.
- Being a fruit is a **necessary condition** for being an apple.
≡ If it is not a fruit then it cannot be an apple.



Conditional Operator: Example

Let f : “you fix my ceiling”, p : “I will pay my rent”

“You fix my ceiling **or** I won’t pay my rent!”

$$f \vee \neg p \equiv p \rightarrow f$$



Landlord

“**If** you do not fix my ceiling, **then**
I won’t pay my rent.”

$$\neg f \rightarrow \neg p \equiv p \rightarrow f$$

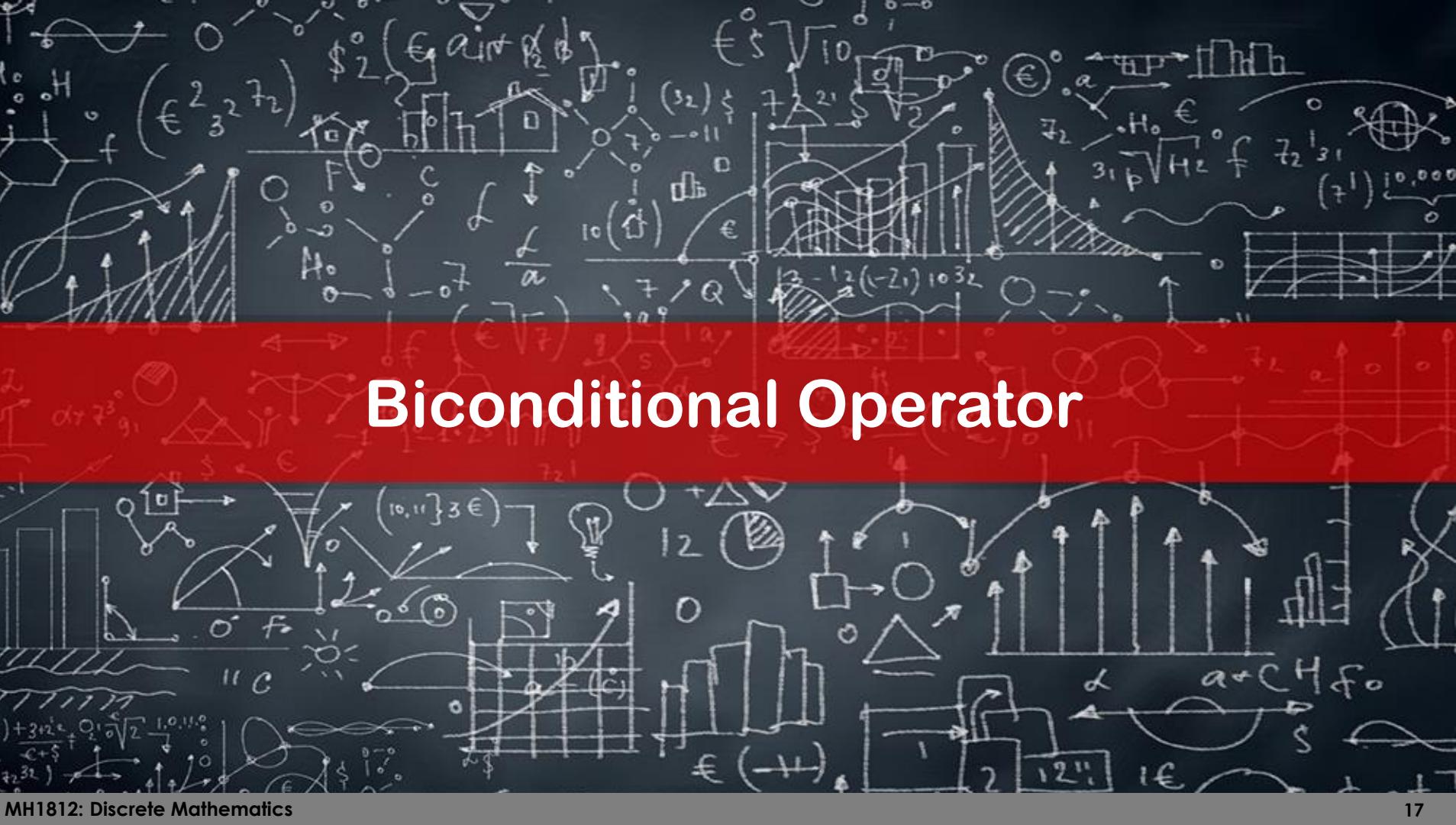


Tenant

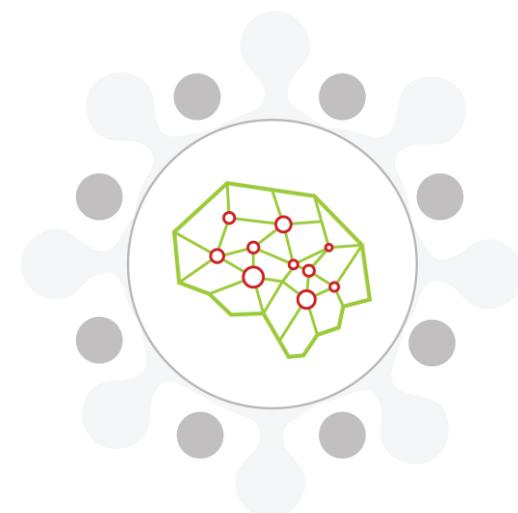
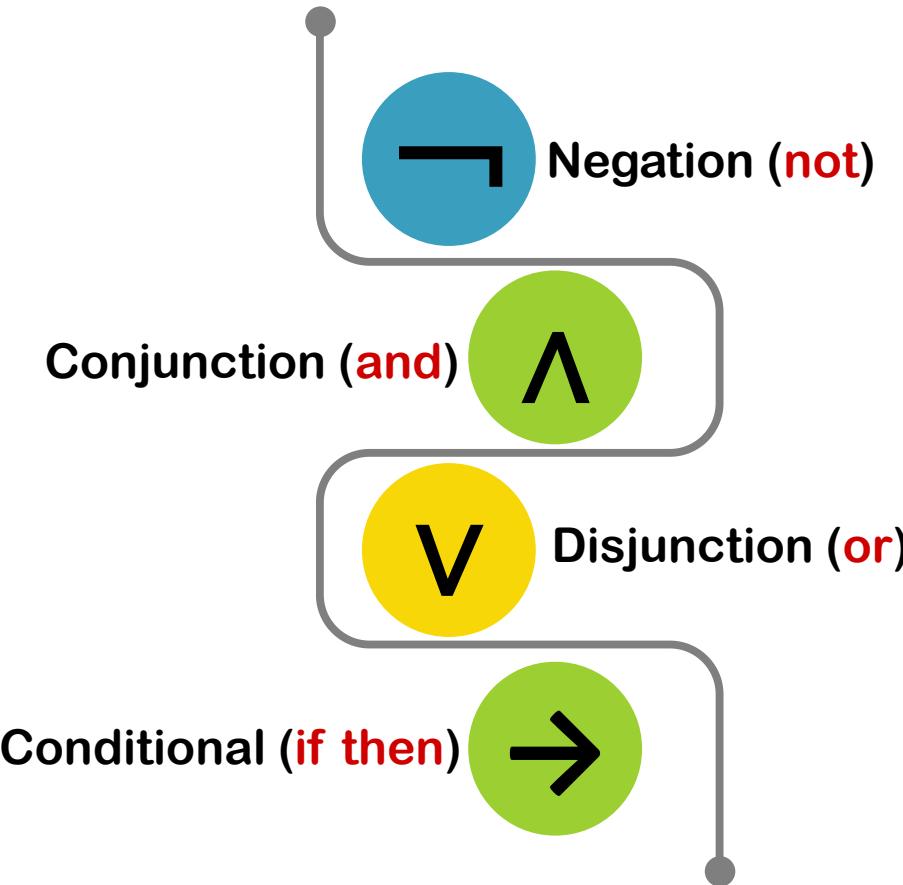
“I will pay my rent **only if** you fix my ceiling.”

$$\neg f \rightarrow \neg p \equiv p \rightarrow f$$

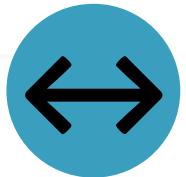
Biconditional Operator



Biconditional Operator: Known Operators

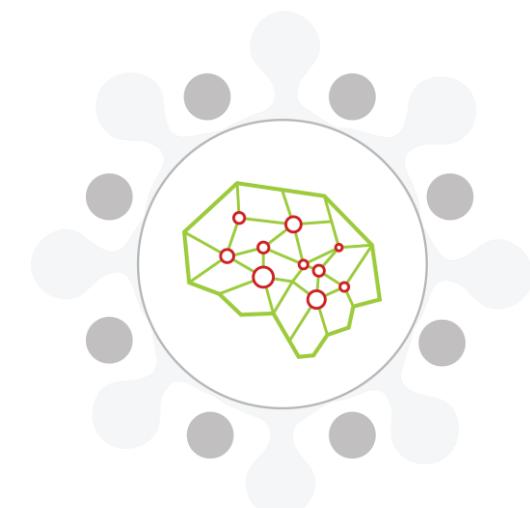


Biconditional Operator: If and Only If

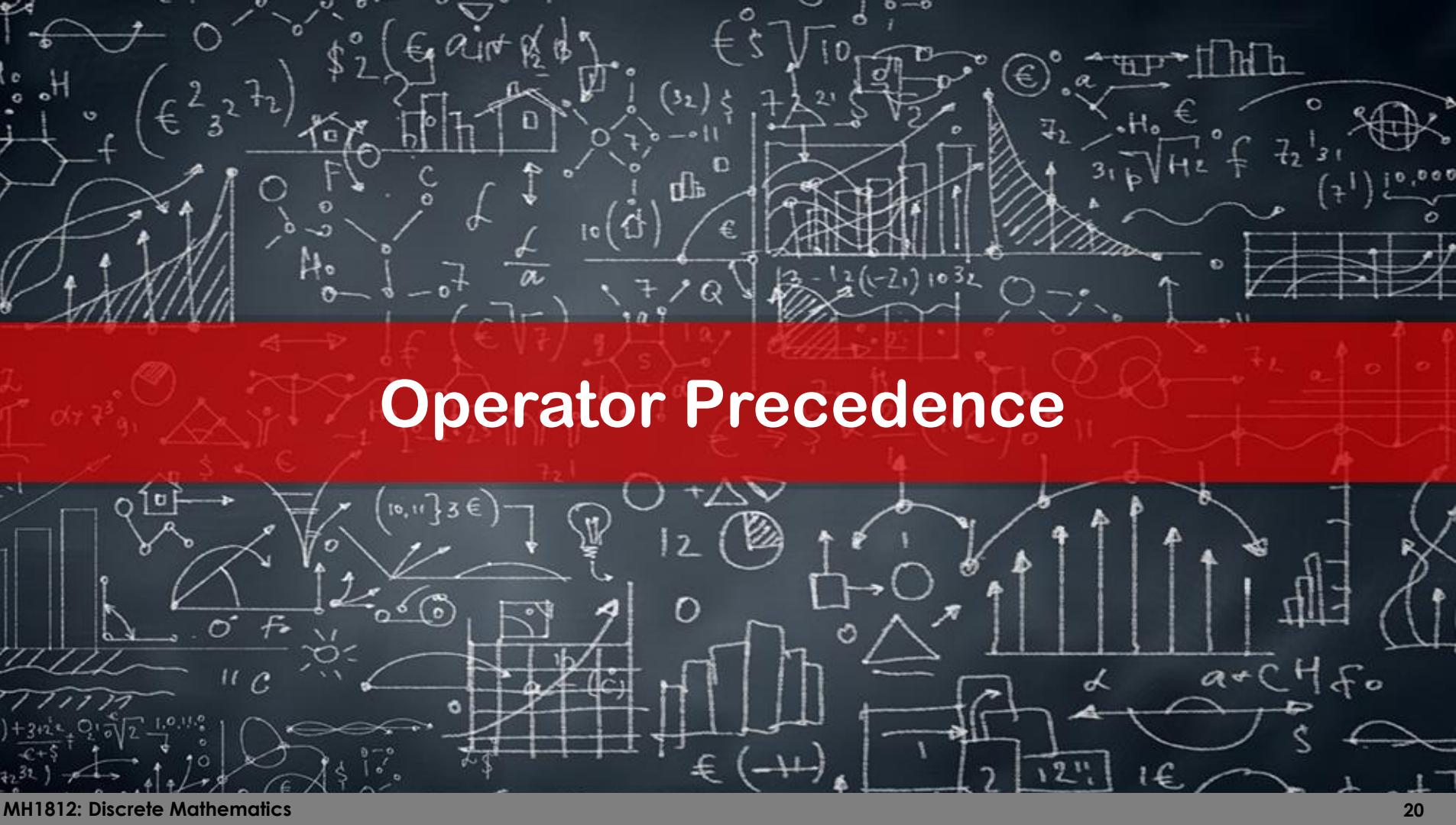


- The **biconditional** of p and q : $p \leftrightarrow q \triangleq (p \rightarrow q) \wedge (q \rightarrow p)$
 - True only when p and q have identical truth value
- If and only if (**iff**)

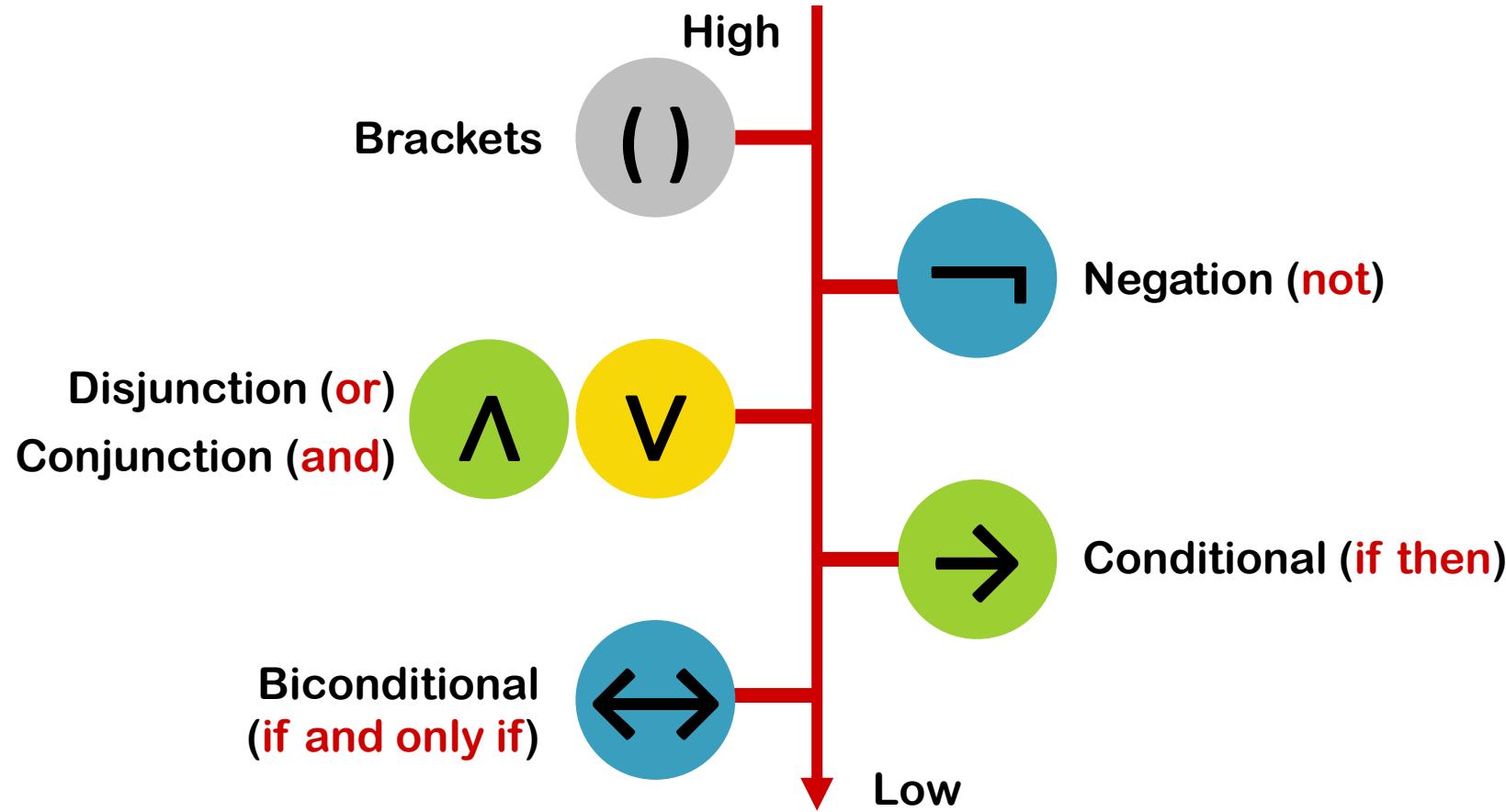
p	q	$p \rightarrow q$	$q \rightarrow p$	$p \leftrightarrow q$
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T



Operator Precedence



Operator Precedence: High to Low



Operator Precedence: Leftmost and Rightmost

Leftmost Precedence

When equal priority instances of **binary connectives** are not separated by (), the **leftmost** one has precedence.

E.g., $p \rightarrow q \rightarrow r \equiv (p \rightarrow q) \rightarrow r$



Rightmost Precedence

When instances of \neg are not separated by (), the **rightmost** one has precedence.

E.g., $\neg\neg\neg p \equiv \neg(\neg(\neg p))$



Operator Precedence: Example



Show that $p \vee q \rightarrow r \equiv (p \rightarrow r) \wedge (q \rightarrow r)$

$$p \vee q \rightarrow r$$

$$\equiv (p \vee q) \rightarrow r$$

Operator precedence

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

Why?

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

De Morgan's

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

Why?

$$\equiv (p \rightarrow r) \wedge (q \rightarrow r)$$

Why?

Topic Summary



Let's recap...

- Useful logical equivalence laws:
 - Proving equivalence using these laws
- Conditional and biconditional operators:
 - Sufficient and necessary conditions
- Operator precedence



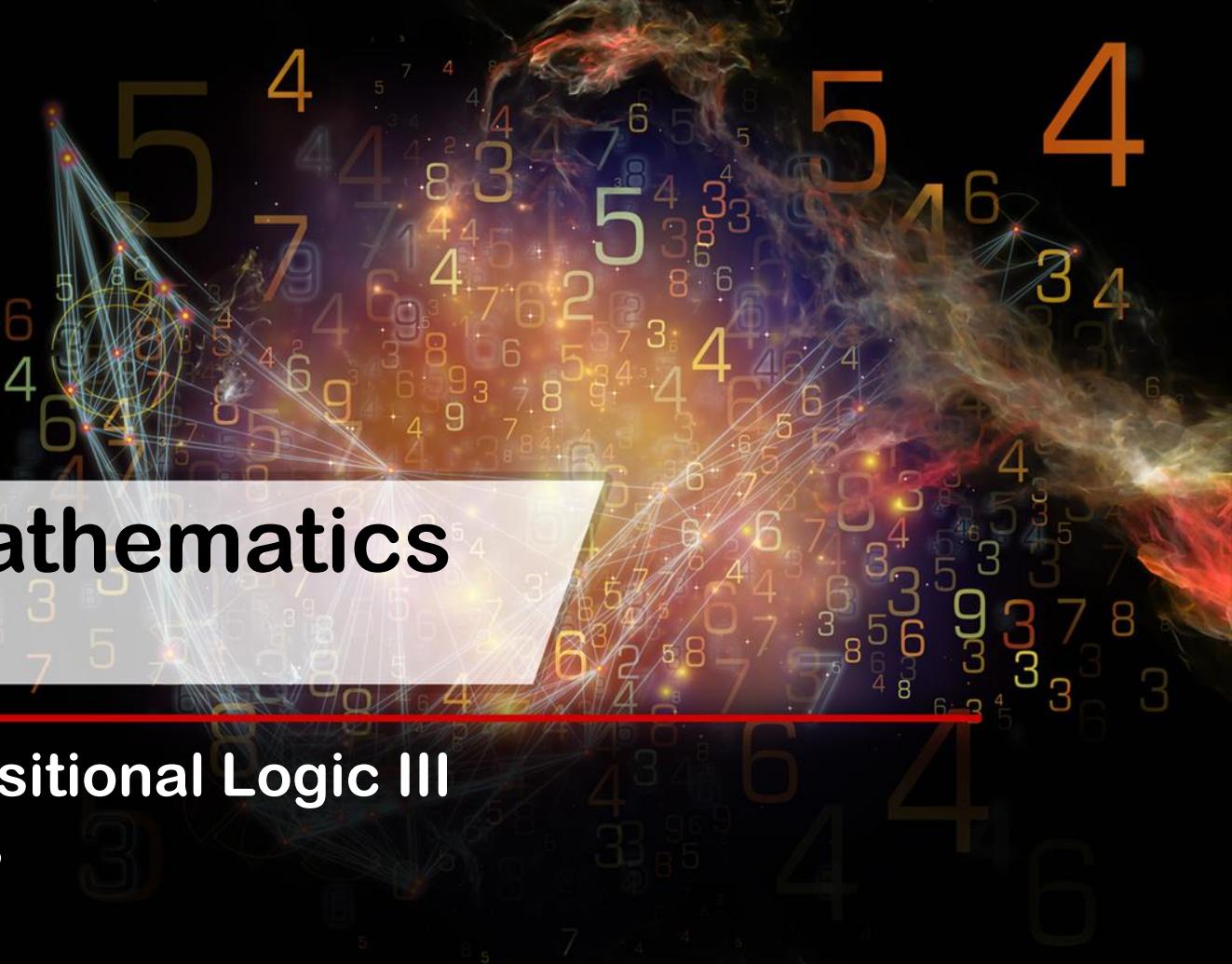


NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE

Discrete Mathematics

MH1812

Topic 2.3 - Propositional Logic III
Dr. Gary Greaves



Topic Overview



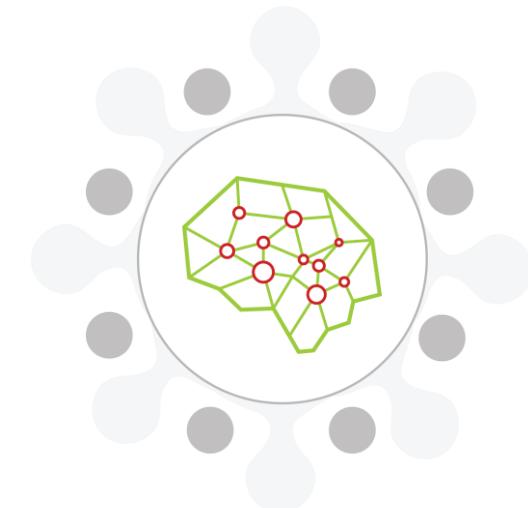
What's in store...

A

rguments

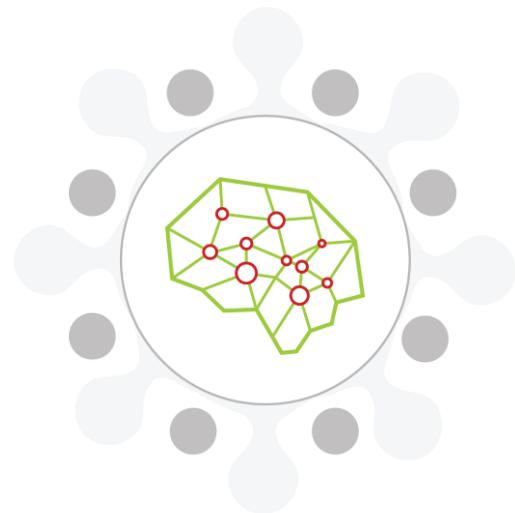
I

nference Rules



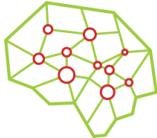
By the end of this lesson, you should be able to...

- Determine whether or not an argument is valid.
- Apply basic inference rules.

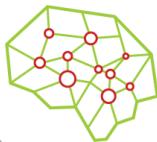


Arguments

Arguments: Valid Argument

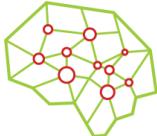


An **argument** is a sequence of statements. The last statement is called the **conclusion**. All the previous statements are called **premises** (or **assumptions/hypotheses**).



A **valid argument** is an argument where the conclusion is true if the premises are all true.

Arguments: Valid Argument



A **valid argument** is an argument where the conclusion is true if the premises are all true.



Example

- “If you pay up in full then I will deliver it”
- “You pay up in full”



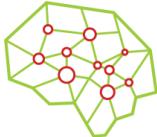
Premises

- “I will deliver it”



Conclusion

Arguments: Valid Argument



A **valid argument** is an argument where the conclusion is true if the premises are all true.

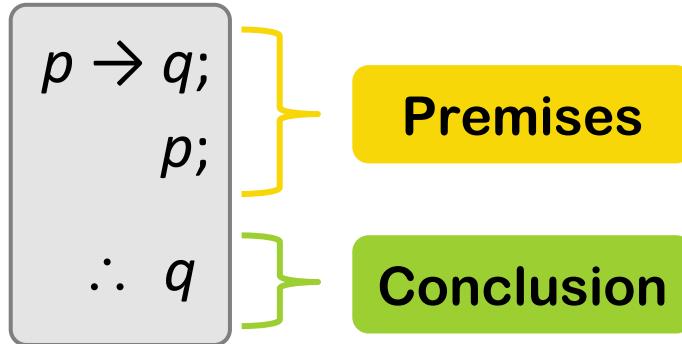
A series of statements form a valid argument if and only if “the conjunction of premises implying the conclusion” is a tautology.

((Premise) \wedge (Premise))

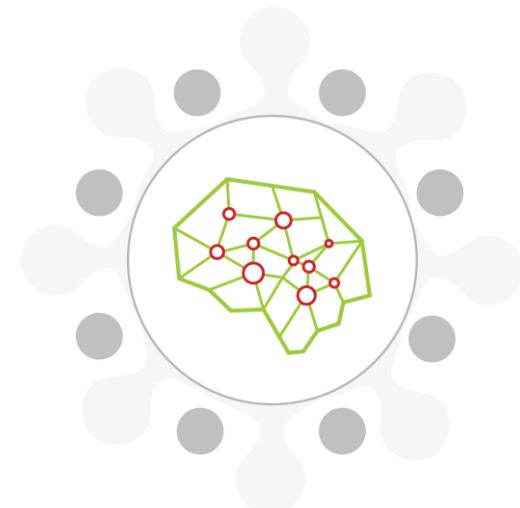


Conclusion

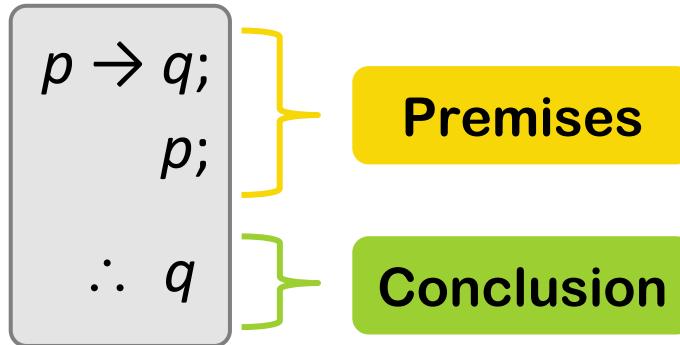
Arguments: A Valid Argument Template



- By definition, a valid argument satisfies: “If the premises are true, then the conclusion is true”.
- To check if the above argument is valid, we need to check that $((p \rightarrow q) \wedge p) \rightarrow q$ is a tautology.



Arguments: Valid Argument Template

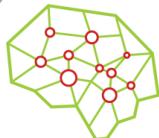


- **Critical rows** are rows with all premises true.
- If in all critical rows the conclusion is true, then the **argument is valid** (otherwise it is invalid).

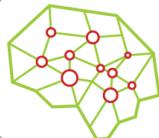
p	q	$p \rightarrow q$	$(p \rightarrow q) \wedge p$	$((p \rightarrow q) \wedge p) \rightarrow q$
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

A red diagonal watermark reading "No need to calculate" is overlaid across the bottom right of the table.

Arguments: Counterexample



If in all critical rows the conclusion is true, then the **argument is valid** (otherwise it is invalid).



A critical row with a false conclusion is a **counterexample**.

A counterexample:

- Invalidates the argument (i.e., makes the argument not valid)
- Indicates a situation where the conclusion does not follow from the premises

Arguments: Invalid Argument Example

- “If it is falling and it is directly above me then I’ll run”
- “It is falling”
- “It is not directly above me”
- “I will not run”



Premises



Conclusion

$$S = (f \wedge a \rightarrow r); \\ f; \\ \neg a; \\ \therefore \neg r$$



Premises



Conclusion

Arguments: Invalid Argument Example

$$\begin{aligned} S = & (f \wedge a \rightarrow r); \\ & f; \\ & \neg a; \\ \therefore & \neg r \end{aligned}$$

Premises

Conclusion

Counterexample

Critical rows

a	r	f	$\neg a$	$f \wedge a$	S	$\neg r$
TT	T	F	F	T	T	F
TT	F	F	F	F	T	F
TF	T	F	T	T	F	T
TF	F	F	F	F	T	T
FT	T	T	T	F	T	F
FT	F	T	F	F	T	F
FF	T	T	T	F	T	T
FF	F	T	F	F	T	T

Arguments: Fallacy



A **fallacy** is an error in reasoning that results in an invalid argument.

Arguments: Fallacy 1 (Converse Error)



Example

- If it is Christmas then it is a holiday.
- It is a holiday. Therefore, it is Christmas!

$$\begin{aligned} p \rightarrow q; \\ q; \\ \therefore p \end{aligned}$$

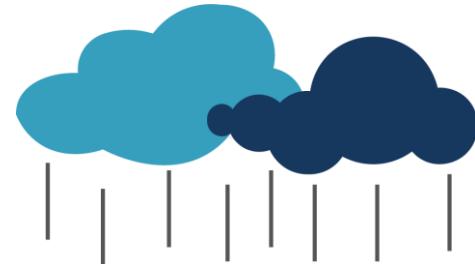


Arguments: Fallacy 2 (Inverse Error)



Example

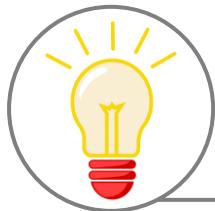
- If it is raining then I will stay at home.
- It is not raining. Therefore, I will not stay at home!



$$\begin{aligned} p \rightarrow q; \\ \neg p; \\ \therefore \neg q \end{aligned}$$

Arguments: Invalid Argument, Correct Conclusion

An argument may be invalid, but it may still draw a correct conclusion (e.g., by coincidence).



Example

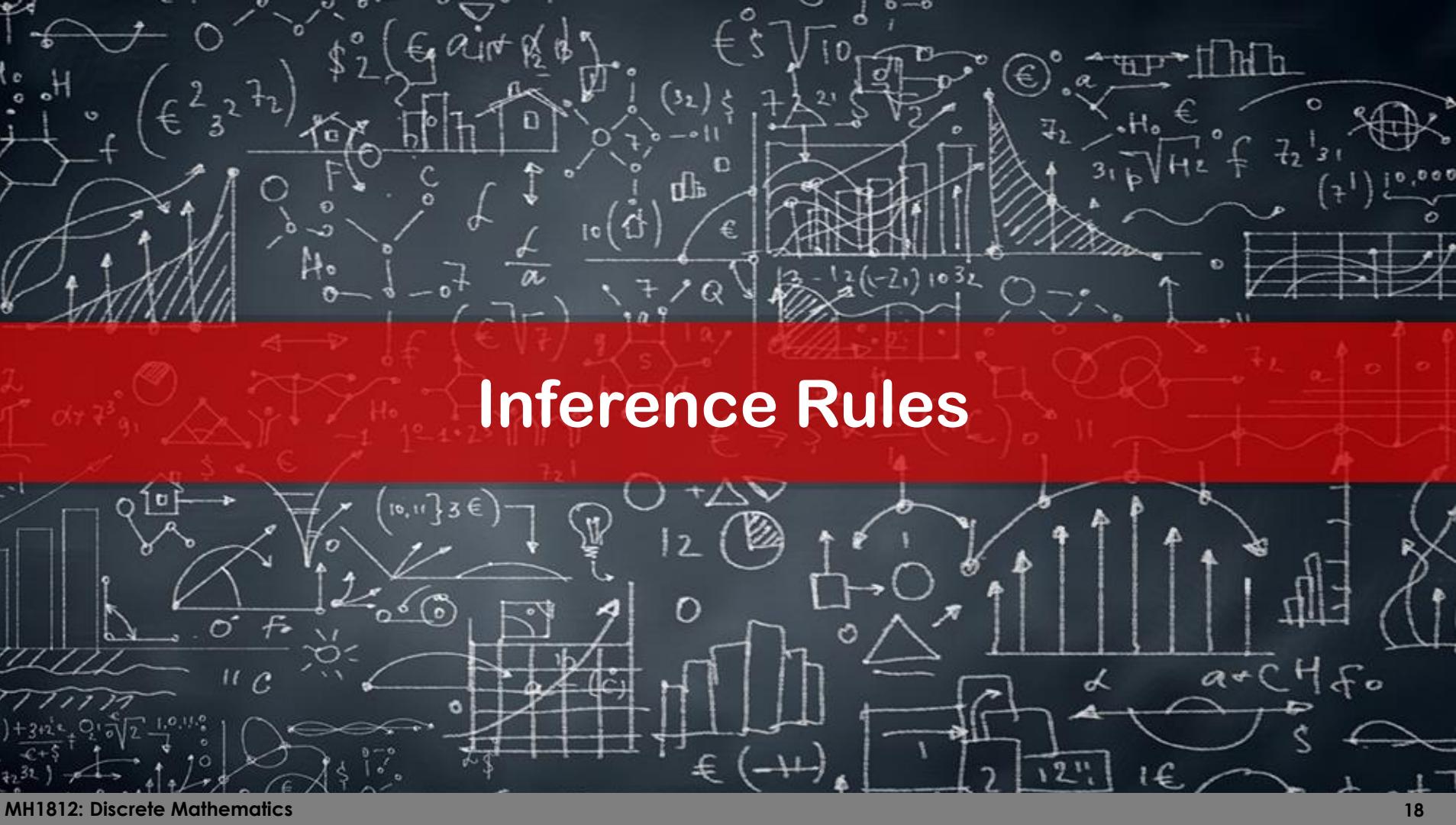
- If New York is a big city then New York has tall buildings.
- New York has tall buildings.
 - So, New York is a big city.

So, what happened?

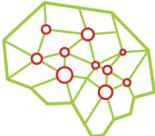
- We have just made an invalid argument,
i.e., converse error!
- But the conclusion is true (a fact true by itself).



Inference Rules



Inference Rules: Definition



A **rule of inference** is a logical construct which takes premises, analyses their syntax and returns a conclusion.

We already saw...

$$\begin{array}{l} p \rightarrow q; \\ p; \\ \therefore q \end{array}$$

Modus Ponens
(Method of Affirming)

$$\begin{array}{l} p \rightarrow q; \\ \neg q; \\ \therefore \neg p \end{array}$$

Modus Tollens
(Method of Denying)

Inference Rules: More Inference Rules

Conjunctive
Simplification
(Particularising)

$$\begin{array}{l} p \wedge q; \\ \therefore p \end{array}$$

Disjunctive
Syllogism
(Case Elimination)

$$\begin{array}{l} p \vee q; \\ \neg p; \\ \therefore q \end{array}$$

Conjunctive
Addition
(Specialising)

$$\begin{array}{l} p; \\ q; \\ \therefore p \wedge q \end{array}$$

Rule of
Contradiction

$$\begin{array}{l} \neg p \rightarrow C; \\ \therefore p \end{array}$$

Disjunctive
Addition
(Generalisation)

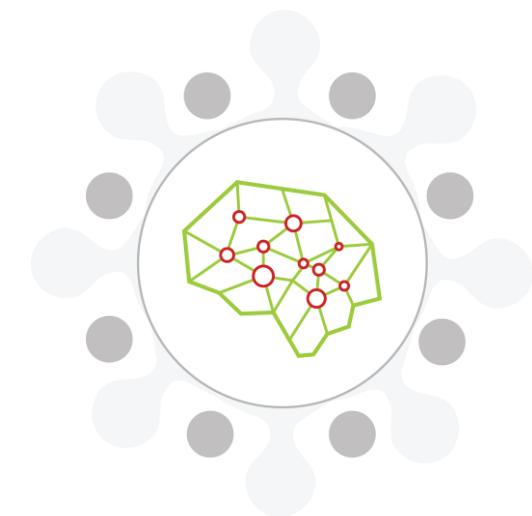
$$\begin{array}{l} p; \\ \therefore p \vee q \end{array}$$

Alternative
Rule of
Contradiction

$$\begin{array}{l} \neg p \rightarrow F; \\ \therefore p \end{array}$$

Inference Rules: Dilemma

Dilemma (case by case discussions)

$$\begin{aligned} p \vee q; \\ p \rightarrow r; \\ q \rightarrow r; \\ \therefore r \end{aligned}$$


Inference Rules: Hypothetical Syllogism

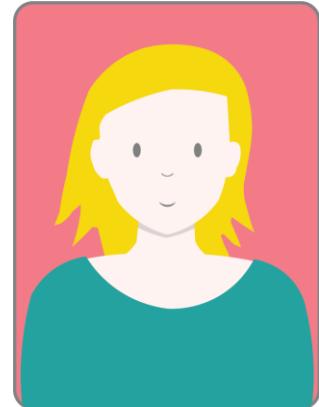
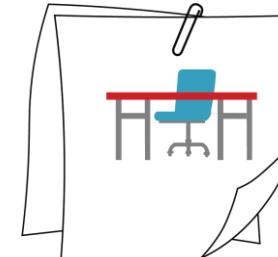
Hypothetical Syllogism

$$\begin{aligned} p &\rightarrow q; \\ q &\rightarrow r; \\ \therefore p &\rightarrow r \end{aligned}$$



Example

- If I do not wake up, then I cannot go to work.
- If I cannot go to work, then I will not get paid.
- Therefore, if I do not wake up, then I will not get paid.



Alice



Inference Rules: Proof Hypothetical Syllogism

$(p \rightarrow q) \wedge (q \rightarrow r)$	(Hypotheses; Assumed True)
$\equiv (p \rightarrow q) \wedge (\neg q \vee r)$	(Conversion Theorem)
$\equiv [(p \rightarrow q) \wedge \neg q] \vee [(p \rightarrow q) \wedge r]$	(Distributive)
$\equiv [(p \rightarrow q) \wedge \neg q] \vee (p \rightarrow q) \wedge [(p \rightarrow q) \wedge \neg q] \vee r$	(Distributive)
$\equiv (p \rightarrow q) \wedge [(p \rightarrow q) \wedge \neg q] \vee r$	(Recall absorption law: $a \vee (a \wedge b) \equiv a$, hence $[(p \rightarrow q) \wedge \neg q] \vee (p \rightarrow q) \equiv p \rightarrow q$)
$\equiv (p \rightarrow q) \wedge [(\neg p \vee q) \wedge \neg q] \vee r$	(Conversion)
$\equiv (p \rightarrow q) \wedge [(\neg p \wedge \neg q) \vee (q \wedge \neg q)] \vee r$	(Distributive)
$\equiv (p \rightarrow q) \wedge [(\neg p \wedge \neg q) \vee F] \vee r$	(Contradiction)
$\equiv (p \rightarrow q) \wedge [(\neg p \wedge \neg q) \vee r]$	(Unity)
$\equiv (p \rightarrow q) \wedge [(\neg p \vee r) \wedge (\neg q \vee r)]$	(Distributive)
$\equiv [(p \rightarrow q) \wedge (\neg q \vee r)] \wedge (\neg p \vee r)$	(Commutative; Associative)
$\therefore (\neg p \vee r) \equiv p \rightarrow r$	(Conjunctive Simplification; Conversion)

Topic Summary



Let's recap...

- Arguments:
 - Valid arguments
 - Invalid arguments
 - Counterexample
 - Fallacy
- Inference rules:
 - Derive conclusions from a bunch of information
 - Some basic inference rules

