Summary

Last time:

- Truth tables,
- Tautologies
- Logical equivalence

Today:

- Proofs in propositional logic
- Demos! ... if time permits ...
- Reading: Chap. 1.6, (1.7, 1.8)

Next time: - Predicate Logic - Reading: Chap. 1.4, 1.5
Checking a tautology or equivalence using the Truth Table method can be very time consuming when there are many propositional variables. E.g., if you have 20 atomic propositions $p_1, p_2, \ldots, p_{20}$, the truth table would have $2^{20} > 1,000,000$ rows!

Alternatively, we can:

- Use truth table method to establish a number of valid inference rules, or reasoning patterns
- Establish the final goal by applying a sequence of inference rules, or proof steps
Propositions: \( H_1, H_2, \ldots, H_n, T \)

**Theorem:** If \( H_1, H_2, \ldots, H_n \) are true, then \( T \) is also true.

- \( H_1, \ldots, H_n \): premises or hypotheses
- \( T \): conclusion or thesis

**Validity:** An inference rule is valid if \( H_1 \land \ldots \land H_n \rightarrow T \) is a tautology.

**Terminology:** Axiom, Inference Rule, Theorem, Lemma, Corollary, Proposition.

Inference rules:

- Usually written \([H_1, \ldots, H_k] \implies T\) or \([H_1, \ldots, H_k] \vdash T\)
- Notice: comma “,” often used for logical \( \land \) operation
Empty conjunction

For any list of statements $H_1, \ldots, H_k$, define the conjunction
$C_k = ((H_1 \land H_2) \land H_3 \ldots) \land H_k$

Notice: if $k = 1$, then $C_1 = H_1$ is just the first statement.

What if $k = 0$? How would you define an empty conjunction?

(A) $C_0 = False$; (B) $C_0 = True$; (C) $C_0 = H_0$; (D) $C_0$ is undefined
Empty conjunction

For any list of statements $H_1, \ldots, H_k$, define the conjunction $C_k = ((H_1 \land H_2) \land H_3 \ldots) \land H_k$

Notice: if $k = 1$, then $C_1 = H_1$ is just the first statement.

What if $k = 0$? How would you define an empty conjunction?

(A) $C_0 = False$; (B) $C_0 = True$; (C) $C_0 = H_0$; (D) $C_0$ is undefined

Answer: (B)

Justification:

- For every $k \geq 1$, we have $C_{k+1} = C_k \land H_{k+1}$
- For $k = 0$, $C_1 = C_0 \land H_1 = H_1$ if $C_0 = True$
- Same reason empty sum (sum of no numbers) is defined as zero
- Zero is the neutral element for $\land$. True is the neutral element for $\land$
Consider an inference rule \([H_1, \ldots, H_k] \implies P\) with no premises 
\((k = 0)\): 
\([\ ] \implies P\) or \(True \implies P\)

**Question:** The statement \(True \rightarrow P\) is logically equivalent to

(A) \(P\); (B) \(\neg P\); (C) \(P \rightarrow False\); (D) None of the above
Consider an inference rule \([H_1, \ldots, H_k] \implies P\) with no premises \((k = 0)\): 
\[
[] \implies P \text{ or } \text{True} \implies P
\]

**Question:** The statement \(\text{True} \rightarrow P\) is logically equivalent to

- (A) \(P\);  
- (B) \(\neg P\);  
- (C) \(P \rightarrow \text{False}\);  
- (D) None of the above

**Answer:** (A). Check using the Truth Table method

- So, inference rules with no premises simply assert that a statement \(P\) is true
- \(P\) is usually called an **axiom**
Implication $P \rightarrow Q \equiv \neg P \lor Q$.

Contrapositive of $P \rightarrow Q$:

- Contrapositive $\neg Q \rightarrow \neg P$
- Equivalent to $P \rightarrow Q$.
- Proof: truth table method, or
  
  $P \rightarrow Q \equiv \neg P \lor Q \equiv Q \lor \neg P \equiv \neg \neg Q \lor \neg P \equiv \neg Q \rightarrow \neg P$

Converse of $P \rightarrow Q$:

- Converse: $Q \rightarrow P$
- not equivalent to $P \rightarrow Q$
- Together with $P \rightarrow Q$, it yields the double implication $P \leftrightarrow Q$. 

Daniele Micciancio  CSE 20: Discrete Mathematics
What is the contrapositive of “If $x < 5$ then $x^2 < 25$”? 

A. If $x \geq 5$ then $x^2 \geq 25$
B. If $x^2 < 25$ then $x < 5$
C. If $x^2 \geq 25$ then $x \geq 5$
D. It is not true that “If $x < 5$ then $x^2 < 25$”
E. This is true only if $x \geq 0$ or $|x| < 5$. 

Remember: the contrapositive of $P \rightarrow Q$ is just a different (but equivalent) way to write an implication $P \rightarrow Q$. 

(B) is called the “converse”, and it is not equivalent to the contrapositive 

(D) is called the “negation” 

(E) is true, but not the correct answer. Contrapositive and converse are well defined regardless of the validity of $P \rightarrow Q$. 

Daniele Micciancio

CSE 20: Discrete Mathematics
What is the contrapositive of “If $x < 5$ then $x^2 < 25$”? 

- **A** If $x \geq 5$ then $x^2 \geq 25$
- **B** If $x^2 < 25$ then $x < 5$
- **C** If $x^2 \geq 25$ then $x \geq 5$
- **D** It is not true that “If $x < 5$ then $x^2 < 25$”
- **E** This is true only if $x \geq 0$ or $|x| < 5$.

Answer: **(C)**

- Remember: the contrapositive of is just a different (but equivalent) way to write an implication $P \rightarrow Q$.

**(B)** is called the “converse”, and it is not equivalent to the contrapositive

**(D)** is called the “negation”

**(E)** is true, but not the correct answer. Contrapositive and converse are well defined regardless of the validity of $P \rightarrow Q$. 

Daniele Micciancio  
CSE 20: Discrete Mathematics
Some common Inference Rules

- **Modus ponens:** \( P, (P \rightarrow Q) \implies Q \)
- **Moduls tollens:** \( \neg Q, (P \rightarrow Q) \implies \neg P \)
  - Same as modus ponens, using contrapositive \( \neg Q \rightarrow \neg P \)
- **AND-elimination**
  - left: \( P \land Q \implies P \)
  - right: \( P \land Q \implies Q \)
- **AND-introduction:** \( P, Q \implies P \land Q \)
- **OR-introduction**
  - left: \( P \implies P \lor Q \)
  - right: \( Q \implies P \lor Q \)
  - also called *weakening* or *addition*
- **OR-elimination:** \( P \rightarrow R, Q \rightarrow R, P \lor Q \implies R \)
  - Also called “proof by cases”
Assume you know:

- It is not sunny this afternoon and it is colder than yesterday
- We go swimming only if it is sunny
- If we do not go swimming, we go hiking
- If we go hiking, we will be home by sunset

Can you conclude that “We will go home by sunset”?

Atomic propositions:

- $s = "It is sunny this afternoon"
- $c = "It is colder than yesterday"
- $w = "We will go swimming"
- $h = "We will go hiking"
- $t = "We will be home by sunset"

Premises: $(\neg s \land c), (w \rightarrow s), \neg w \rightarrow h, h \rightarrow t$.

Conclusion: $t$
**Theorem:** If \((-s \land c), (w \rightarrow s), \neg w \rightarrow h\) and \(h \rightarrow t\) are true, then \(t\) is also true.

**Proof:**

1. \((-s \land c)\) (premise)
2. \((w \rightarrow s)\) (premise)
3. \((-w \rightarrow h)\) (premise)
4. \((h \rightarrow t)\) (premise)
Theorem: If \((\neg s \land c), (w \rightarrow s), \neg w \rightarrow h\) and \(h \rightarrow t\) are true, then \(t\) is also true.

Proof:

1. \((\neg s \land c)\) (premise)
2. \((w \rightarrow s)\) (premise)
3. \(\neg w \rightarrow h\) (premise)
4. \(h \rightarrow t\) (premise)
5. \(\neg s\) (AND-elim 1)
Using inference rules:

**Theorem:** If \((\neg s \land c)\), \((w \rightarrow s)\), \((\neg w \rightarrow h)\) and \((h \rightarrow t)\) are true, then \(t\) is also true.

**Proof:**

1. \((\neg s \land c)\) (premise)
2. \((w \rightarrow s)\) (premise)
3. \((\neg w \rightarrow h)\) (premise)
4. \((h \rightarrow t)\) (premise)
5. \((\neg s)\) (AND-elim 1)
6. \((\neg w)\) (modus tollens 2,5)
Theorem: If \((\neg s \land c)\), \((w \rightarrow s)\), \((\neg w \rightarrow h)\) and \((h \rightarrow t)\) are true, then \(t\) is also true.

Proof:

1. \((\neg s \land c)\) (premise)
2. \((w \rightarrow s)\) (premise)
3. \((\neg w \rightarrow h)\) (premise)
4. \((h \rightarrow t)\) (premise)
5. \((\neg s)\) (AND-elim 1)
6. \((\neg w)\) (modus tollens 2,5)
7. \((h)\) (modus ponens 3,6)
Using inference rules:

**Theorem:** If \((\neg s \land c), (w \rightarrow s), \neg w \rightarrow h\) and \(h \rightarrow t\) are true, then \(t\) is also true.

**Proof:**

1. \((\neg s \land c)\) (premise)
2. \((w \rightarrow s)\) (premise)
3. \(\neg w \rightarrow h\) (premise)
4. \(h \rightarrow t\) (premise)
5. \(\neg s\) (AND-elim 1)
6. \(\neg w\) (modus tollens 2,5)
7. \(h\) (modus ponens 3,6)
8. \(t\) (modus ponens 4,7)

So, \(t\) follows from the premises. If all premises are satisfied, we will go home by sunset.
Proof structure

Theorem:

- Premises: $H_1, \ldots, H_k$
- Conclusion: $T$.

Proof: a sequence of logical statements $S_1, \ldots, S_n$ (propositions) ending in $S_n = T$ such that each $S_i$ is

- either one of the theorem premises $S_i = H_j$, or
- an axiom, i.e., an inference rule with no premises
- it is obtained from previous statements $S_j$ (with $j < i$) using a valid inference rule $H_{j_1}, \ldots, H_{j_k} \implies S_i$

Meta-theorem: If all the inference rules used in the proof of a theorem are valid, then the theorem is also valid, i.e., $(H_1 \land \ldots \land H_k) \rightarrow T$ is a tautology.
Example: Frege system

- Logical connectives: $\neg$, $\rightarrow$. (Other connectives are defined in terms of $\neg$, $\rightarrow$.)
- Only one inference rule: $P, P \rightarrow Q \implies Q$ (modus ponens)
- Axioms
  - $A \rightarrow (B \rightarrow A)$
  - $\neg\neg A \rightarrow A$
  - $A \rightarrow \neg\neg A$
  - $(A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A)$
  - $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$

Completeness: all valid theorems (tautologies) can be proved using Frege system

There are several other classic proof systems for propositional calculus: Hilbert, Russell, Lukasievicz and Tarski, etc.
Can you define “∨” using only “→” and “¬”?

$P \lor Q$ is logically equivalent to

- (A) $\neg (Q \rightarrow P)$
- (B) $P \rightarrow \neg Q$
- (C) $\neg P \rightarrow Q$
- (D) None of the above
Can you define “∨” using only “→” and “¬”? 

\( P \lor Q \) is logically equivalent to 

- A. \( \neg (Q \rightarrow P) \) 
- B. \( P \rightarrow \neg Q \) 
- C. \( \neg P \rightarrow Q \) 
- D. None of the above 

Answer (C): \( \neg P \rightarrow Q \). Check using Truth Table method.
Can you define “∧” using only “→” and “¬”? 

$P \lor Q$ is logically equivalent to 

A. $\neg (Q \rightarrow P)$  
B. $P \rightarrow \neg Q$  
C. $\neg P \rightarrow Q$  
D. None of the above

Answer (C): $\neg P \rightarrow Q$. Check using Truth Table method.

What about $P \land Q$?
Can you define “∨” using only “→” and “¬”?

\( P \lor Q \) is logically equivalent to

- A. \( \neg (Q \rightarrow P) \)
- B. \( P \rightarrow \neg Q \)
- C. \( \neg P \rightarrow Q \)
- D. None of the above

Answer (C): \( \neg P \rightarrow Q \). Check using Truth Table method.

What about \( P \land Q \)?

Using De Morgan Law:

\[ P \land Q \equiv \neg (\neg P \lor \neg Q) \equiv \neg (P \rightarrow \neg Q). \]

Check using Truth Table method.
Formal proofs are seldom natural and intuitive. Especially so when statements use only $\neg$ and $\rightarrow$

Formal proofs are easy for a computer to check, but can be hard for a human to understand

Mathematicians usually describe their proofs informally, in a natural language like English.

Logic can be used to explain and justify reasoning patterns

Example:

- Formal: $[A, B] \rightarrow (A \land B)$
- English (informal): “In order to prove $(A \land B)$, I will first prove $A$ and then prove $B$.”
Natural deduction

Formal proof system, but somehow closer to informal “natural” proofs.

- Use all connectives $\land$, $\lor$, $\rightarrow$, $\neg$, $\leftrightarrow$
- Each connective is associated with introduction and elimination rules

Example: AND ($\land$)

- Introduction: $A, B \implies A \land B$.  
  “In order to prove $A \land B$ one needs to prove $A$ and $B$”
- Elimination: $A \land B \implies A$ and $A \land B \implies B$ “From $A \land B$, one can deduce both $A$ and $B$”

Also used by most modern automated theorem provers. Closely related to programming languages and type checking.
Demos: What do you want to prove today?

The Incredible Proof Machine!
- Fun educational project
- http://incredible.pm/

The LEAN theorem prover
- State of the art theorem prover under development at Microsoft Research
- https://leanprover.github.io/

You will **NOT** be tested on this