Lecture 01: Propositional Logic Basics

Jingjin Yu | Computer Science @ Rutgers

September 1, 2020

RUTGERS
THE STATE UNIVERSITY
OF NEW JERSEY

Algorithmic
Robotics &
Control Lab

Outline

- **Motivation**
- **Propositions**
- **Operations**
  - Negation, conjunctions, disjunctions
  - Conditional statements, bidirectional statements
- **Bit operations**
- **Some applications**

Motivation

- Propositional logic in computer science?
  - **Goal**: enable formal & rigorous reasoning for computing
  - **Why**: making sure programs are **correct**
  - **How**: proofs through logical inferences

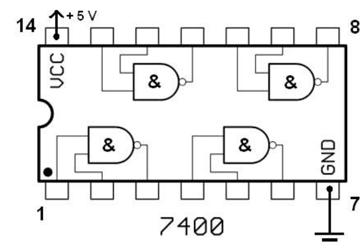
- The **basics** are important!
  - Definitions

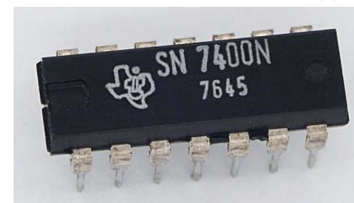
    May need further definitions!

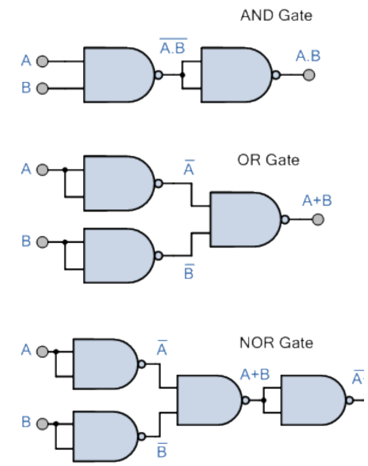
    - **Ex**: A triangle is polygon with three edges and three vertices
  - Axioms
    - **Ex**: For every two points A and B, there exists a line containing both
  - Theorems
    - **Ex**: Pythagorean theorem

Propositional Logic

- **General logic**
  - Logic (short for logical systems) provides meanings (true or false) to mathematical statements
  - Applications in computer science
    - Design of computer circuits
    - Computer programs
      - E.g. conditional statements
    - …







- **Propositional logic**
  - A relatively simple but foundational logical system, with which we begin our study of discrete math
  - Basic element: **propositions** (like numbers in arithmetic)

Proposition

- $\mathbf{Def}^n$: A **proposition** is a declarative sentence that is either $true$ or $false$, but not both.
- $\mathbf{Ex}$:
  - Propositions (true or false?)
    - Toronto is the capital of Canada.
    - $1 + 1 = 2$
    - $3 \times 7 = 20$
  - Non-propositions
    - What time is it?
    - Do not sleep while listening to the video!
    - $x + 1 = 2$
  - What about: $x^3 + y^3 = z^3$ for x, y, z as positive integers?

Proposition Variables

- Propositions may be represented using **variables**
  - **Ex**: $\boldsymbol{p}$ = "1 + 1 = 2"
  - **Ex**: $\boldsymbol{q}$ = "Toronto is the capital of Canada"
  - Generally, uses $p, q, r, \ldots$ to denote propositions

- True and false
  - Also, we often use $\boldsymbol{T}$ for $true$ and $\boldsymbol{F}$ for $false$
  - Note that $true$ and $false$ are also propositions
    - **Ex**: $r = T$

- With propositions, we can build **compound propositions**

Liar Paradox

- What about the sentence: *This current statement is* $false$.
  - If the statement is false, then the sentence is true?
  - If the statement is true, then the sentence if false?
  - So, is the sentence $true$ or $false$?

- This is mostly a language artifact
  - Think about it
  - Then read: https://en.wikipedia.org/wiki/Liar_paradox

Negations

- **Def^n**: Let p be a proposition, the **negation** of p, denoted $\neg p$ (not p), is the statement "it is not the case that p".
  - **Ex**: negate "Michael's PC runs Linux"
    - Negation: "**It is not the case** that Michael's PC runs Linux"
    - Simplification: "Michael's PC does not run Linux"
  - **Ex**: negate "Ada's phone has at least 256GB of memory."
    - Negation: "**It is not the case** that Ada's phone has at least 256GB of memory."
    - Simplification: "Ada's phone has less than 256GB of memory."

Truth Table: a First Look

- **Truth table** lists the truth values of a compound proposition based on the truth value(s) of its component proposition(s)
  - Negation, $\neg p$, is a compound proposition
    - It contains a single component proposition, p
    - Its truth table then has two columns $p, \neg p$
    - With a single component, it has 2^1 "data" rows

p	$\neg p$
T	F
F	T

  - For a compound proposition with k components, e.g., $P(p_1, \ldots, p_k)$, its truth table has 2^k data rows.
  - Truth tables we will see often have four data rows ($k = 2$)

Conjunction and Disjunctions

exclusive or

- $\mathbf{Def^n}$: Let p and q be propositions. The **conjunction** of p and q, denoted by $p \land q$, is the proposition "p and q". $p \land q$ is $true$ if and only if p is $true$ and q is $true$

p xor q

- $\mathbf{Def^n}$: Let p and q be propositions. The **disjunction** of p and q, denoted by $p \lor q$, is the proposition "p or q". $p \lor q$ is $false$ if and only if p is $false$ and q is $false$
  - "∨" is "inclusive or"
  - "or" in English is may be inclusive or exclusive.
  - "Exclusive or" in logic is denoted as "⊕"
    - Not frequently used

Truth Table and an Example for ∧ and ∨

- Truth table for ∧, ∨, and ⊕ directly from definitions

p	q	$p \wedge q$	$p \vee q$	$p \oplus q$
T	T	T	T	F
T	F	F	T	T
F	T	F	T	T
F	F	F	F	F

- **Ex**: p: "Tom is sleeping", q: "Tom wears a baseball hat".
  - Conjunction: Tom wears a baseball hat and is sleeping
  - Disjunction: Tom wears a baseball hat or Tom is sleeping
    - True if either or both of "Tom is sleeping" and "Tom wears a baseball hat" hold

Conditional Statements

- **Def^{n}**: Let p and q be propositions. The **conditional statement** "$p \to q$" is the proposition "if p, then q". $p \to q$ is $false$ if and only if p is $true$ and q is $false$
  - p is the **hypothesis**, q is the **consequence**

- **Ex**: "If I am elected, (then) I will lower taxes"
- **Ex**: "If you get 100%, (then) you will get an A"

- Equivalent statements that you may see
  - "p implies q"
  - "p only if q"
  - "q if p"
  - "q follows from p"

Related Statements and Truth Table

- Statements related to the conditional $p \rightarrow q$
  - Converse: $q \rightarrow p$
  - Contrapositive: $\neg q \rightarrow \neg p$
  - Inverse: $\neg p \rightarrow \neg q$

- The truth table

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

  - Observe that $p \rightarrow q$ and $\neg q \rightarrow \neg p$ are the same
  - Note that $p \rightarrow q$ is the same as $\neg p \vee q$

From Conditional to Related Statements

- **Ex**: "The home team wins whenever it is raining"
  - First, we need to break the conditional into components
    - p: "it rains", q: "the home team wins"
  - Converse
    - "If the home team wins, then it rains"
  - Contrapositive
    - "If the home team does not win, then it is not raining"
  - Inverse:
    - "If it is not raining, then the home team does not win"

Bidirectional Statement

- **Def$^{\boldsymbol{n}}$**: Let p and q be propositions. The **bidirectional statement** "$p \leftrightarrow q$" is the proposition "p if only if q". $p \leftrightarrow q$ is $true$ if and only if p and q take the same truth value
- Also known as "p is necessary and sufficient for q"
- Truth table

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

Evaluation of Compound Statements

- Precedence: $\neg > \wedge > \vee > \rightarrow > \leftrightarrow$
  - To avoid confusion, can always add parenthesis "()" to provide priority
- We may always use truth table to evaluate
- **Ex**: $(p \rightarrow \neg q) \rightarrow (p \wedge q)$

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

Bit Operations

- A bit is a symbol with two values, 0 or 1
  - Perfect for representing propositions
    - Set $0 = false, 1 = true$
- Computer bit operations allow logical operations using bits, which can be done in parallel

- **Ex:**

1011 0110	
0001 1101	
1011 1111	Bitwise OR
0001 0100	Bitwise AND
1010 1011	Bitwise XOR

Applications

- **Ex**: Modular representation of texts, e.g., English
  - "You can access internet from campus if you are a CS student or you are not a freshman"
    - p: "You are a CS student"
    - q: "You are not a freshman"
    - r: "You can access internet from campus"
    - The sentence is then "$(p \vee q) \rightarrow r$"
- **Ex**: Design and verification of logic circuits