Discrete Mathematics

Propositional Logic

Shin Hong

Textbook coverage

- Section 1.1 Propositional logic
- Section 1.2 Applications of propositional logic
- Section 1.3 Propositional equivalence

Logic

- Logic, or a logic system, is a set of rules to specify and derive a certain kind of statements
  - to achieve clarity and correctness in an argument

- A logic system comprises of the syntax and the semantics
  - syntax: symbolic structure of the statements
  - semantics: a mapping from symbolic structures to things that the logic system concerns

Propositional Logic

- A proposition is a declarative sentence that is either true or false
  - 1 + 1 = 2
  - *Vancouver is the capital of Canada*
  - ~~1 + 2 + 3~~
  - ~~x + 1 = 2~~

- A statement in the propositional logic consists one or multiple propositions connected with logical operators

- A propositional variable is a symbol that represents a propositional statement
  - a propositional variable has either true or false as its value
  - the value is definitive within a statement

Syntax

- Grammar

```
P := A | C
A := p | q | r | ... | True | False
C := ¬P | (P) | P ∨ P | P ∧ P | ...
```

- An atomic proposition is one that cannot be expressed in term of simpler terms

- A compound proposition is formed with one or more propositions and logical operators
  - logical operators (connectives): negation, disjunction, implication, etc.
    - E.g., The negation of p for a proposition p, denoted as $\neg p$, is the proposition that is true only when p is false

Evaluation (Semantics)

- A propositional statement with propositional variables may have different evaluations (truth values) depending on the values of each propositional variable
  - ex. $p \vee (q \wedge r)$

- An assignment (model or valuation) of a proposonal statement is a combination of truth values of the propositional variables
  - e.g., $\phi_1 = (p{:}\,\mathrm{T},\ q{:}\,\mathrm{T}, r{:}\,\mathrm{T})$ or $[\![p]\!]_{\phi_1} = \mathrm{T}$, $[\![q]\!]_{\phi_1} = \mathrm{T}$, $[\![r]\!]_{\phi_1} = \mathrm{T}$

    $\phi_2 = (p{:}\,\mathrm{F},\ q{:}\,\mathrm{T}, r{:}\,\mathrm{F})$ or $[\![p]\!]_{\phi_1} = \mathrm{F}$, $[\![q]\!]_{\phi_1} = \mathrm{T}$, $[\![r]\!]_{\phi_1} = \mathrm{F}$

$$\phi_1 \vDash p \vee (q \wedge r)$$
$$\phi_2 \nvDash p \vee (q \wedge r)$$

Implication (Conditional Statement)

- An implication is a logical connective such that $p \rightarrow q$ evaluates to *true* when q is true if p is true
  - used to state a conditional statement
    - examples
      - you get F if you do not take an exam
      - if you are in the Handong campus, you are in Pohang
      - $x < y \rightarrow x < y + 1$
      - $(2 + 3 = 4) \rightarrow (1 + 2 = 4)$
  - $p \rightarrow q$ is equivalent with $\neg p \vee q$
- The converse of $p \rightarrow q$ is $q \rightarrow p$.
- The inverse of $p \rightarrow q$ is $\neg p \rightarrow \neg q$.
- The contrapositive of $p \rightarrow q$ is $\neg q \rightarrow \neg p$.

Equivalence

- The condition that two propositions p and q evaluate to the same can be expressed as $(p \to q) \wedge (q \to p)$, or simply $p \leftrightarrow q$
  - have the same truth value for every assignment
  - a statement $p \leftrightarrow q$ refers as p if and only if q (or simply p iff q)

Example

Equivalence	*Name*
$p \wedge \mathbf{T} \equiv p$ $p \vee \mathbf{F} \equiv p$	Identity laws
$p \vee \mathbf{T} \equiv \mathbf{T}$ $p \wedge \mathbf{F} \equiv \mathbf{F}$	Domination laws
$p \vee p \equiv p$ $p \wedge p \equiv p$	Idempotent laws
$\neg(\neg p) \equiv p$	Double negation law
$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$	Commutative laws
$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	Associative laws
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	Distributive laws

- De Morgan's law:

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

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

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

Propositional Satisfiability

- A proposition p is **satisfiable** if there exists an assignment that makes p true

- A proposition p is **unsatisfiable** if p is not satisfiable
  - a unsatisfiable proposition is called as *contradiction*

- A proposition p is **valid** if p is true for all assignments
  - a valid proposition is called as tautology
  - e.g., if x = y, then x = y

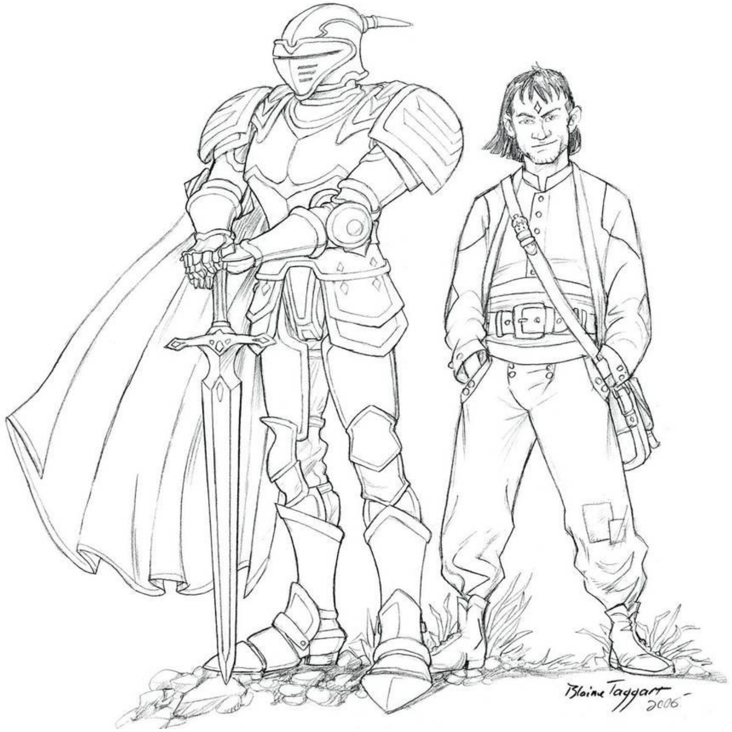
I just want to live while I am alive - Bon Jovi

Logic Puzzle - Knight or Knaves

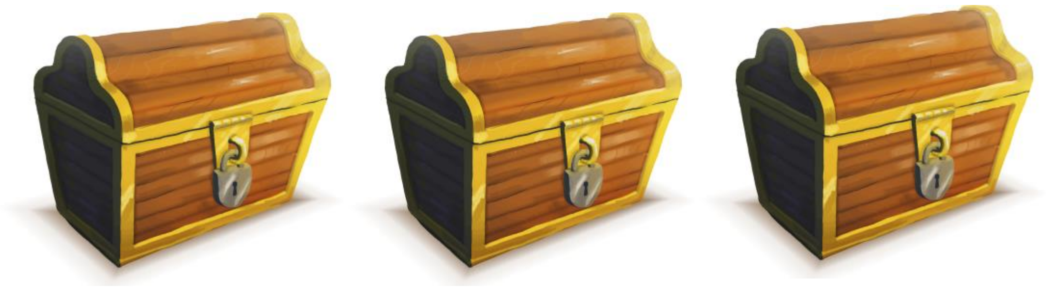
- An island has two kinds of inhabitants:
  knights, who always tell the truth, and
  knaves, who always lie.

- You met John and Paul in the island
  - John said "Paul is a knight."
  - Paul said "The two of us are of opposite types."

  What are the types of Jonh and Paul?



Logic Puzze: Treasure



- There are 3 trunks only one of which contains a treasure.
- Trunk 1 and Trunk 2 are inscribed with "This trunk is empty" and Trunk 3 is inscribed with "Treasure is in Trunk 2".
- You know that only one of the three inscriptions is true.
- Where's the treasure?

System Requirement Analysis

- Logic-based languages (formal languages) are powerful tools for specifying and analyzing software requirements rigorously

- Case of Lufthansa A320 Airbus accident at Warsaw in 1993
  - Requirement:
    Turn on reverse thrust when airplane is running on runway for landing
  - System design specification (simplified):
    - SET `REVERSE_THRUST` AS `ON` WHEN (`MODE` = `LANDING`) AND (`ALTITUDE` = `0`)
    - SET `MODE` AS `LANDING` WHEN (`VELOCITY` $\neq$ `0`) AND (`LANDING_GEAR_ANG` $\neq$ 0)