COMP9020
Foundations of Computer Science
2023 Term 1

- Textbook (R & W) - Ch. 1, Sec. 1.1-1.4
  Ch. 2, Sec. 2.1
- Problem set 1 + Quiz (due Tuesday week 2 at 5pm)
- Self-guided study: Exercises Ch. 1 (R & W), ...

COMP9020 23T1Course Convenor

Name:	Michael Thielscher	
Consults:	Wednesday 2pm-3pm	
Room:	K17 401I	(turn left from lift and dial 57129)
Research:	Artificial Intelligence, Robotics, General Game Playing	
Pastimes:	Fiction, Films, Food, Football	
Tutors:	Kevin Luong, Mark Raya	
Help Tute:	Tuesday 5pm-6pm	Room 202 Ainsworth (Bldg K-J17)
	Tuesday 7pm-8pm	Online ($\Rightarrow$ Moodle $\Rightarrow$ Help Tutorial)
Admin:	Michael Schofield	
Email:	cs9020@cse.unsw.edu.au for all course-related enquiries	

Course Aims

The course aims to increase your level of mathematical maturity to assist with the fundamental problem of **finding, formulating, and proving** properties of programs.

The actual content is taken from a list of subjects that constitute the basis of the tool box of every serious practitioner of computing:

- numbers, sets, formal languages — week 1
- logic, proofs, Boolean algebra — week 2
- function and relation theory — week 3–4
- graph theory — week 5
- **mid-term test** (no lectures) — week 6
- induction, recursion, order of growth — week 7
- structured counting, probability — week 8–10

NB

"universitas" (Lat.) = sum of all things, a whole

By acquiring knowledge and enhancing your problem-solving skills, you're preparing yourself for the future

Course Material

All course information is placed on the WebCMS3 course website

webcms3.cse.unsw.edu.au/COMP9020/23T1/

Need to login to access course materials.

Textbook:

- KA Ross and CR Wright: Discrete Mathematics

Supplementary textbook:

- E Lehman, FT Leighton, A Meyer: Mathematics for Computer Science

Lectures, Problem Sets, Quizzes

Lectures will:

- present theory
- demonstrate problem-solving methods

Lecture slides will be made available before lecture

Feel free to ask questions, but No Idle Chatting

The weekly **homework** and its assessment (**quizzes**) aim to:

- clarify any problems with lecture material
- work through exercises related to lecture topics

Homework (problem sets) made available before the lecture
Assessment (quizzes) opens on day of first lecture each week

Sample solutions posted a week later, **after** the quiz deadline
Do them yourself! and Don't fall behind!

Assessment Summary

1. online quizzes (weeks 2, 3, 4, 5, 7, 8, 9) — max. marks 14
2. online mid-term test (1 hour in week 6) — max. marks 26
3. online exam (2 hours in the exam period) — max. marks 60

NB

Your **overall score** for this course will be the **maximum** of

- weekly + mid-term + exam
- weekly + exam $* \frac{86}{60}$
- mid-term + exam $* \frac{74}{60}$
- exam $* \frac{100}{60}$

$\Rightarrow$ If you do better in the final exam, your weekly quizzes and/or mid-term test result will be ignored

$\Rightarrow$ The quizzes and mid-term test can only improve your final mark

NB

To pass the course, your overall score must be 50 or higher **and** your mark for the final exam must be 25 or higher.

The online assessment of your weekly homework:

- becomes available after the Wednesday lecture each week
- is due **Tuesday, 5:00pm** in the following week

You get your own individual questions for each quiz

- each quiz is worth 2 marks
- max. quiz mark $= 14$
  Week 1 Quiz is practice and won't count towards your mark for the weekly homework

Mid-term Test, Weekly Help Sessions

NB

Online test in week 6
(1 hour on Friday, 24 March, starting at 10:30am)

You get your own individual questions:

- some multiple-choice questions
- some descriptive/analytical questions with open answers

max. mid-term test marks = 26

A tutorial-style help session:

- aims to further explain homework/quiz solutions
- and help with any other questions related to course contents

Tuesday 5-6pm Room 202, Bldg K-J17 **or**
Tuesday 7-8pm Online
Starting in week 2. Attendance is entirely voluntary.

Notation for Numbers

Definition

Integers $\mathbb{Z} = \{\ldots -2, -1, 0, 1, 2, \ldots\}$
Reals $\mathbb{R}$
$\lfloor . \rfloor : \mathbb{R} \longrightarrow \mathbb{Z}$ — **floor** of x, the greatest integer $\leq x$
$\lceil . \rceil : \mathbb{R} \longrightarrow \mathbb{Z}$ — **ceiling** of x, the least integer $\geq x$

Example

$\lfloor \pi \rfloor = 3 = \lceil e \rceil \qquad \pi, e \in \mathbb{R};\ \lfloor \pi \rfloor, \lceil e \rceil \in \mathbb{Z}$

Simple properties

- $\lfloor -x \rfloor = -\lceil x \rceil$, hence $\lceil x \rceil = -\lfloor -x \rfloor$
- $\lfloor x+t \rfloor = \lfloor x \rfloor + t$ and $\lceil x+t \rceil = \lceil x \rceil + t$, for all $t \in \mathbb{Z}$

Fact

Let $k, m, n \in \mathbb{Z}$ such that $k > 0$ and $m \geq n$. The number of multiples of k in the interval $[n, m]$ is

$$\left\lfloor \frac{m}{k} \right\rfloor - \left\lfloor \frac{n-1}{k} \right\rfloor$$

Exercise

$\boxed{1.1.4}$

(b) $2\lfloor 0.6 \rfloor - \lfloor 1.2 \rfloor = -1$

$2\lceil 0.6 \rceil - \lceil 1.2 \rceil = 0$

(d) $\lceil \sqrt{3} \rceil - \lfloor \sqrt{3} \rfloor = 1$; the same for every non-integer

$\boxed{1.1.19(a)}$

Give x, y s.t. $\lfloor x \rfloor + \lfloor y \rfloor < \lfloor x + y \rfloor$

$\lfloor 3\pi \rfloor + \lfloor e \rfloor = 9 + 2 = 11 < 12 = \lfloor 9.42\ldots + 2.71\ldots \rfloor = \lfloor 3\pi + e \rfloor$

Exercise

$\boxed{1.1.4}$

(b) $2\lfloor 0.6\rfloor - \lfloor 1.2\rfloor = -1$

$2\lceil 0.6\rceil - \lceil 1.2\rceil = 0$

(d) $\left\lceil\sqrt{3}\right\rceil - \left\lfloor\sqrt{3}\right\rfloor = 1$; the same for every non-integer

$\boxed{1.1.19(a)}$

Give x, y s.t. $\lfloor x\rfloor + \lfloor y\rfloor < \lfloor x+y\rfloor$

$$\lfloor 3\pi\rfloor + \lfloor e\rfloor = 9 + 2 = 11 < 12 = \lfloor 9.42\ldots + 2.71\ldots\rfloor = \lfloor 3\pi + e\rfloor$$

Divisibility

Definition

Let $m, n \in \mathbb{Z}$.
'$m|n$' — m is a **divisor** of n, defined by $n = k \cdot m$ for some $k \in \mathbb{Z}$
Also stated as: 'n is divisible by m', 'm divides n', 'n multiple of m'

$m \nmid n$ — negation of $m|n$

Notion of divisibility applies to all integers (positive, negative, zero)

$1|m$, $-1|m$, $m|m$, $m|-m$, for every m
$n|0$ for every n; $0 \nmid n$ except $n = 0$

Definition

$r = n \bmod m$ — **remainder** when n divided by m,
defined by $n = k \cdot m + r$ for $0 \leq r < m$

$m|n$ if, and only if, $n \bmod m = 0$

Numbers > 1 divisible only by 1 and itself are called **prime**.
Greatest common divisor $\gcd(m, n)$
Numbers m, n s.t. $\gcd(m, n) = 1$ are said to be **relatively prime**.
Least common multiple $\mathrm{lcm}(m, n)$

NB

$\gcd(m, n)$ and $\mathrm{lcm}(m, n)$ are always taken as positive, even if m or n is negative.

$$\begin{aligned} \gcd(-4, 6) = \gcd(4, -6) = \gcd(-4, -6) = \gcd(4, 6) &= 2 \\ \mathrm{lcm}(-5, -5) = \ldots &= 5 \end{aligned}$$

Note also that $gcd(0, 0)$ is undefined (why?)

NB

Number theory (the study of prime numbers, divisibility etc.) is important in cryptography, for example.

Absolute Value

$$|x| = \begin{cases} x & \text{, if } x \geq 0 \\ -x & \text{, if } x < 0 \end{cases}$$

Fact

$\gcd(m, n) \cdot \text{lcm}(m, n) = |m| \cdot |n|$

Exercise

$\boxed{1.2.2}$ *True* or *False*. Explain briefly.
(a) $n|1$
(b) $n|n$
(c) $n|n^2$

$\boxed{1.2.7(b)}$ $\gcd(0, n) \stackrel{?}{=}$

$\boxed{1.2.12}$ Can two even integers be relatively prime?

$\boxed{1.2.9}$ Let m, n be positive integers.
(a) What can you say about m and n if $\operatorname{lcm}(m, n) = m \cdot n$?
(b) What if $\operatorname{lcm}(m, n) = n$?

Exercise

$\boxed{1.2.2}$ *True* or *False*. Explain briefly.
(a) $n|1$ — only if $n = 1$ (for $n \in \mathbb{Z}$ also $n = -1$)
(b) $n|n$ — always
(c) $n|n^2$ — always

$\boxed{1.2.7(b)}$ $\gcd(0, n) = |n| \quad$ (provided $n \neq 0$)

$\boxed{1.2.12}$ Can two even integers be relatively prime? No. (why?)

$\boxed{1.2.9}$ Let m, n be positive integers.
(a) What can you say about m and n if $\operatorname{lcm}(m, n) = m \cdot n$?
They must be relatively prime since always $\operatorname{lcm}(m, n) = \frac{mn}{\gcd(m,n)}$
(b) What if $\operatorname{lcm}(m, n) = n$?
m must be a divisor of n

Euclid's gcd Algorithm

$$f(m,n) = \begin{cases} m & \text{if } m = n \\ f(m-n,n) & \text{if } m > n \\ f(m,n-m) & \text{if } m < n \end{cases}$$

Fact

For $m > 0, n > 0$ the algorithm always terminates. (Proof?)

Fact

For $m, n \in \mathbb{Z}$, if $m > n$ then $\gcd(m,n) = \gcd(m-n,n)$

Proof.

For all $d \in \mathbb{Z}$, ($d|m$ and $d|n$) if, and only if, ($d|m-n$ and $d|n$):

"$\Rightarrow$": if $d|m$ and $d|n$ then $m = a \cdot d$ and $n = b \cdot d$, for some a, b

then $m - n = (a-b) \cdot d$, hence $d|m-n$

"$\Leftarrow$": if $d|m-n$ and $d|n$ then $\ldots$ $d|m$ (why?)

Sets

A set is defined by the collection of its elements.
Sets are typically described by:
(a) Explicit enumeration of their elements

$$
\begin{aligned}
S_1 &= \{a, b, c\} = \{a, a, b, b, b, c\} \\
&= \{b, c, a\} = \dots \quad \text{three elements} \\
S_2 &= \{a, \{a\}\} \quad \text{two elements} \\
S_3 &= \{a, b, \{a, b\}\} \quad \text{three elements} \\
S_4 &= \{\} \quad \text{zero elements} \\
S_5 &= \{\{\{\}\}\} \quad \text{one element} \\
S_6 &= \{\, \{\}, \{\{\}\} \,\} \quad \text{two elements}
\end{aligned}
$$

$x \in S$ — object x **is an element of** (or: **belongs to**) set S

(b) Specifying the properties their elements must satisfy; the elements are taken from some 'universal' domain. A typical description involves a **logical** property $P(x)$

$$S = \{\, x : x \in X \text{ and } P(x) \,\} = \{\, x \in X : P(x) \,\}$$

We distinguish between an element and the set comprising this single element. Thus always $a \neq \{a\}$.

Subsets

$S \subseteq T$ — S is a **subset** of T; includes the case of $T \subseteq T$

- all elements of S are also elements of T

$S \subset T$ — a **proper** subset: $S \subseteq T$ and $S \neq T$

Empty set $\{\,\}$ (alternative notation: $\emptyset$)

- $\emptyset \subseteq X$ for all sets X

NB

An element of a set and a subset of that set are two different concepts

$$a \in \{a, b\}, \quad a \not\subseteq \{a, b\}; \qquad \{a\} \subseteq \{a, b\}, \quad \{a\} \notin \{a, b\}$$

Cardinality, Power Set

Number of elements in a set X (various notations):

$$|X| = \#(X) = \mathrm{card}(X)$$

Power set $\mathrm{Pow}(X) = \{ A : A \subseteq X \}$

Fact

Always $|Pow(X)| = 2^{|X|}$.

$|\emptyset| = 0 \qquad \mathrm{Pow}(\emptyset) = \{\emptyset\} \qquad |\mathrm{Pow}(\emptyset)| = 1$

$\mathrm{Pow}(\mathrm{Pow}(\emptyset)) = \{\emptyset, \{\emptyset\}\} \qquad |\mathrm{Pow}(\mathrm{Pow}(\emptyset))| = 2 \quad \ldots$

$|\{a\}| = 1 \qquad \mathrm{Pow}(\{a\}) = \{\emptyset, \{a\}\} \qquad |\mathrm{Pow}(\{a\})| = 2 \quad \ldots$

Note the **type**: $\mathrm{Pow}(X)$ is always a **set** of **sets**

Sets of Numbers

Natural numbers $\mathbb{N} = \{0, 1, 2, \ldots\}$
Positive integers $\mathbb{P} = \{1, 2, \ldots\}$
Common notation $\mathbb{N}_{>0} = \mathbb{Z}_{>0} = \mathbb{N} \setminus \{0\}$

Integers $\mathbb{Z} = \{\ldots, -n, -(n-1), \ldots, -1, 0, 1, 2, \ldots\}$
Rational numbers (fractions) $\mathbb{Q} = \left\{ \frac{m}{n} : m, n \in \mathbb{Z}, n \neq 0 \right\}$
Real numbers (decimal or binary expansions) $\mathbb{R}$
$r = a_1 a_2 \ldots a_k \,.\, b_1 b_2 \ldots$

In $\mathbb{P} \subset \mathbb{N} \subset \mathbb{Z}$ different symbols denote different numbers.
In $\mathbb{Q}$ and $\mathbb{R}$ the standard representation is not necessarily unique.

Exercise

$\boxed{1.3.2}$ Find the cardinalities of sets

1. $|\{\ \frac{1}{n} : n \in \{1,2,3,4\}\ \}| \stackrel{?}{=}$
2. $|\{\ n^2 - n : n \in \{0,1,\ldots,4\}\ \}| \stackrel{?}{=}$
3. $|\{\ \frac{1}{n^2} : n \in \mathbb{P} \text{ and } 2|n \text{ and } n < 11\ \}| \stackrel{?}{=}$
4. $|\{\ 2+(-1)^n : n \in \mathbb{N}\ \}| \stackrel{?}{=}$

Exercise

$\boxed{1.3.2}$ Find the cardinalities of sets

1. $|\{\ \frac{1}{n} : n \in \{1, 2, 3, 4\}\ \}| = 4$ — four 'indices', no repetitions of values
2. $|\{\ n^2 - n : n \in \{0, 1, \ldots, 4\}\ \}| = 4$ — one 'repetition' of value
3. $|\{\ \frac{1}{n^2} : n \in \mathbb{P} \text{ and } 2|n \text{ and } n < 11\ \}| = 5$
4. $|\{\ 2 + (-1)^n : n \in \mathbb{N}\ \}| = 2$ — what are the two elements?

NB

Proper ways to introduce reals include Dedekind cuts and Cauchy sequences, neither of which will be discussed here. Natural numbers etc. are either axiomatised or constructed from sets ($0 \stackrel{\text{def}}{=} \{\}$, $n+1 \stackrel{\text{def}}{=} n \cup \{n\}$)

NB

If we need to emphasise that an object (expression, formula) is defined through an equality we use the symbol $\stackrel{\text{def}}{=}$. It denotes that the object on the left is defined by the formula/expression given on the right.

Number sets and their containments

$$\mathbb{P} \subset \mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R}$$

Derived sets of positive numbers

$$\mathbb{P} = \mathbb{N}_{>0} = \mathbb{Z}_{>0} = \{n \in \mathbb{Z} : n \geq 1\} \subset \mathbb{Q}_{>0} = \{r : r = \tfrac{k}{l} > 0\} \subset \mathbb{R}_{>0}$$

Derived sets of integers

$$2\mathbb{Z} = \{\, 2x : x \in \mathbb{Z} \,\} \qquad \text{the even numbers}$$

$$3\mathbb{Z} + 1 = \{\, 3x + 1 : x \in \mathbb{Z} \,\}$$

Intervals

Intervals of numbers (applies to any type)

$$[a,b] = \{x | a \leq x \leq b\}; \quad (a,b) = \{x | a < x < b\}$$

$$[a,b] \supseteq [a,b),\ (a,b] \supseteq (a,b)$$

NB

$(a,a) = (a,a] = [a,a) = \emptyset$; however $[a,a] = \{a\}$.

Intervals of $\mathbb{P}, \mathbb{N}, \mathbb{Z}$ are finite: if $m \leq n$

$$[m,n] = \{m, m+1, \ldots, n\} \qquad |[m,n]| = n - m + 1$$

Exercise

$\boxed{1.3.10}$ Number of elements in the sets

1. $\{-1, 1\}$
2. $[-1, 1]$
3. $(-1, 1)$

Exercise

$\boxed{1.3.10}$ Number of elements in the sets

1. $\{-1, 1\}$ — 2
2. $[-1, 1]$ — 3 (if over $\mathbb{Z}$); ∞ (if over $\mathbb{Q}$ or $\mathbb{R}$)
3. $(-1, 1)$ — 1 (if over $\mathbb{Z}$); ∞ (if over $\mathbb{Q}$ or $\mathbb{R}$)

Set Operations

Union $A \cup B$; Intersection $A \cap B$

Note that there is a correspondence between set operations and logical operators (to be discussed in Week 2):
One can match set A with that subset of the universal domain, where the property a holds, then match B with the subset where b holds. Then
$A \cup B \Leftrightarrow a$ or b; $A \cap B \Leftrightarrow a$ and b

We say that A, B are **disjoint** if $A \cap B = \emptyset$

NB

$A \cup B = B \Leftrightarrow A \subseteq B$ $A \cap B = B \Leftrightarrow A \supseteq B$

Other set operations

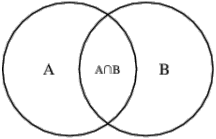
- $A \setminus B$ — **difference**, set difference, relative complement
  It corresponds (logically) to a but not b
- $A \oplus B$ — **symmetric difference**

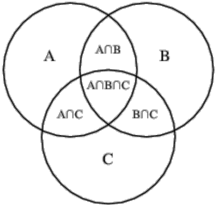
$$A \oplus B \stackrel{\text{def}}{=} (A \setminus B) \cup (B \setminus A)$$

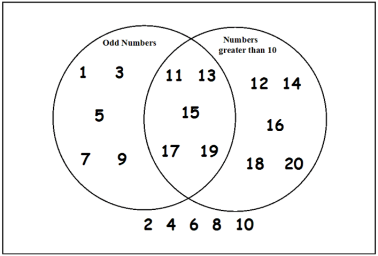
  It corresponds to a and not b or b and not a; also known as **xor (exclusive or)**
- A^c — set **complement** w.r.t. the 'universe'
  It corresponds to 'not a'

Venn Diagrams

p23–26: are a simple graphical tool to reason about the algebraic properties of set operations.







Laws of Set Operations

Commutativity	$A \cup B = B \cup A$
	$A \cap B = B \cap A$
Associativity	$(A \cup B) \cup C = A \cup (B \cup C)$
	$(A \cap B) \cap C = A \cap (B \cap C)$
Distribution	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
Idempotence	$A \cup A = A$
	$A \cap A = A$
Identity	$A \cup \emptyset = A$
	$A \cap \emptyset = \emptyset$
Double Complementation	$(A^c)^c = A$
De Morgan laws	$(A \cup B)^c = A^c \cap B^c$
	$(A \cap B)^c = A^c \cup B^c$

Exercise

$\boxed{1.4.4}$ $\Sigma = \{a, b\}$
(d) All subsets of Σ ?
(e) $|\mathrm{Pow}(\Sigma)| \stackrel{?}{=}$

$\boxed{1.4.7}$ $A \oplus A \stackrel{?}{=}, \quad A \oplus \emptyset \stackrel{?}{=}$

$\boxed{1.4.8}$ Relate the cardinalities $|A \cup B|$, $|A \cap B|$, $|A \setminus B|$, $|A \oplus B|$, $|A|$, $|B|$

Exercise

$\boxed{1.4.4}$ $\Sigma = \{a, b\}$
(d) All subsets of Σ ? $\quad \emptyset, \{a\}, \{b\}, \{a, b\}$
(e) $|\mathrm{Pow}(\Sigma)| = 4$

$\boxed{1.4.7}$ $A \oplus A = \emptyset, \quad A \oplus \emptyset = A$ for all A

$\boxed{1.4.8}$ Relate the cardinalities:
$|A \cup B| = |A| + |B| - |A \cap B|$
hence $|A \cup B| + |A \cap B| = |A| + |B|$
$|A \setminus B| = |A| - |A \cap B|$
$|A \oplus B| = |A| + |B| - 2|A \cap B|$

Cartesian Product

$S \times T \stackrel{\text{def}}{=} \{\ (s,t) : s \in S,\ \ t \in T\ \}$ where (s,t) is an **ordered** pair

$\bigtimes_{i=1}^{n} S_i \stackrel{\text{def}}{=} \{\ (s_1,\ldots,s_n) : s_k \in S_k,\ \text{for } 1 \leq k \leq n\ \}$

$S^2 = S \times S, \quad S^3 = S \times S \times S, \ldots, \quad S^n = \times_1^n S, \ldots$

$\emptyset \times S = \emptyset$, for every S
$|S \times T| = |S| \cdot |T|, \quad |\bigtimes_{i=1}^{n} S_i| = \prod_{i=1}^{n} |S_i|$

Functions

We deal with functions as a set-theoretic concept, it being a special kind of correspondence (between two sets)

$f : S \longrightarrow T$ describes pairing of the sets: it means that f assigns to every element $s \in S$ a unique element $t \in T$

To emphasise that a specific element is sent, we can write $f : x \mapsto y$, which means the same as $f(x) = y$

Formal Languages

Σ — **alphabet**, a finite, nonempty set

Examples (of various alphabets and their intended uses)

$\Sigma = \{a, b, \ldots, z\}$ for single words (in lower case)
$\Sigma = \{\sqcup, -, a, b, \ldots, z\}$ for composite terms
$\Sigma = \{0, 1\}$ for binary integers
$\Sigma = \{0, 1, \ldots, 9\}$ for decimal integers

The above cases all have a natural ordering; this is not required in general, thus the set of all Chinese characters forms a (formal) alphabet.

Definition

word — any finite string of symbols from Σ
empty word — λ

Example

$\omega = aba$, $\omega = 01101\ldots1$, etc.

$\text{length}(\omega)$ — $\#$ of symbols in ω
$\text{length}(aaa) = 3, \text{length}(\lambda) = 0$
The only operation on words (discussed here) is **concatenation**, written as juxtaposition $\nu\omega, \omega\nu\omega, ab\omega, \omega b\nu, \ldots$

NB

$\lambda\omega = \omega = \omega\lambda$
$\text{length}(\nu\omega) = \text{length}(\nu) + \text{length}(\omega)$

Notation: Σ^k — set of all words of length k
We often identify $\Sigma^0 = \{\lambda\}$, $\Sigma^1 = \Sigma$
Σ^* — set of all words (of all lengths)
Σ^+ — set of all nonempty words (of any positive length)

$$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots; \quad \Sigma^{\leq n} = \bigcup_{i=0}^{n} \Sigma^i$$

$$\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \ldots = \Sigma^* \setminus \{\lambda\}$$

A **language** is a subset of Σ^*. Typically, only the subsets that can be formed (or described) according to certain rules are of interest. Such a collection of 'descriptive/formative' rules is called a **grammar**.

Examples: Programming languages, Database query languages

Examples

$\boxed{1.3.10}$ Number of elements in the sets (cont'd)

(e) Σ^* where $\Sigma = \{a, b, c\}$ — $|\Sigma^*| = \infty$

(f) $\{\ \omega \in \Sigma^* : \text{length}(\omega) \leq 4\ \}$ where $\Sigma = \{a, b, c\}$

$|\Sigma^{\leq 4}| = 3^0 + 3^1 + \ldots + 3^4 = \frac{3^5 - 1}{3 - 1} = \frac{243 - 1}{2} = 121$

Elementary Logic

Exercise

Claim:
A *necessary* condition for the program to terminate is to input a positive number.

Suppose you want to formally verify this claim. Which would be the correct logical way of formalising and proving this?

- $Terminates \Rightarrow Positive\_Input$ correct
- $Positive\_Input \Rightarrow Terminates$

Elementary Logic

Exercise

Claim:
A *necessary* condition for the program to terminate is to input a positive number.

Suppose you want to formally verify this claim. Which would be the correct logical way of formalising and proving this?

- $Terminates \Rightarrow Positive\_Input$ **correct**
- $Positive\_Input \Rightarrow Terminates$

Proofs

A **mathematical proof** of a proposition p is a chain of logical deductions leading to p from a base set of axioms.

Example

Proposition: Every group of 6 people includes a group of 3 who each have met each other or a group of 3 who have not met a single other person in that group.
Proof: by case analysis.

But what are propositions, logical deductions, and axioms? And what is a sound case analysis?

The Real World vs Symbols

$$
\begin{tikzcd}
\text{symbols} \arrow[r, "\text{symbolic manipulation}"] \arrow[d, "\text{relationship}"] & \text{symbols} \arrow[d, "\text{relationship}"] \\
\text{world} \arrow[r, "\text{physical operation}"] & \text{world}
\end{tikzcd}
$$

NB

"Essentially, all models are wrong. But some are useful."

(George Box)

The main relationship between symbols and the world of concern in logic is that of a *sentence of a language* being *true* in the world. A sentence of a natural language (like English, Cantonese, Warlpiri) is *declarative*, or a **proposition**, if it can be meaningfully be said to be either true or false.

Examples

- Richard Nixon was president of Ecuador.
- A square root of 16 is 4.
- Euclid's program gets stuck in an infinite loop if you input 0.
- Whatever list of numbers you give as input to this program, it outputs the same list but in increasing order.
- $x^n + y^n = z^n$ has no nontrivial integer solutions for $n > 2$.

The following are *not* declarative sentences of English:

- Gubble gimble goo
- For Pete's sake, take out the garbage!
- Did you watch MediaWatch last week?
- Please waive the prerequisites for this subject for me.

Declarative sentences in natural languages can be *compound* sentences, built out of other sentences.
Propositional Logic is a formal representation of some constructions for which the truth value of the compound sentence can be determined from the truth value of its components.

- Lists L and M contain the same elements *and* M is ordered.
- Either you have a passport *or* you cannot travel abroad.
- *It is not the case that* this program always halts.

Not all constructions of natural language are truth-functional:

- *Trump believes that* Iran is developing nukes.
- This program always halts *because* it contains no loops.
- The disk crashed *after* I saved my file.

NB

Various **modal logics** extend classical propositional logic to represent, and reason about, these and other constructions.

Propositional Logic

symbol	text
$\wedge$	"and", "but", ";", ":"
$\vee$	"or", "either ... or ..."
$\neg$	"not", "it is not the case that"

Truth tables:

A	B	$A \wedge B$
F	F	F
F	T	F
T	F	F
T	T	T

A	B	$A \vee B$
F	F	F
F	T	T
T	F	T
T	T	T

A	$\neg A$
F	T
T	F

Applications I: Program Logic

Example

```
if x > 0 or (x <= 0 and y > 100):
```

Let $p \stackrel{\text{def}}{=} (\texttt{x > 0})$ and $q \stackrel{\text{def}}{=} (\texttt{y > 100})$

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

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

This is equivalent to $p \vee q$. Hence the code can be simplified to

```
if x > 0 or y > 100:
```

Somewhat more controversially, consider the following constructions:

- if A then B
- A only if B
- B if A
- A implies B
- it follows from A that B
- whenever A, B
- A is a sufficient condition for B
- B is a necessary condition for A

Each has the property that if true, and A is true, then B is true.

Example

If you are from England *then* you are from the UK.

We can *approximate* the English meaning of these by "not (A and not B)", written $A \Rightarrow B$, which has the following truth table:

A	B	A $\Rightarrow$ B
F	F	T
F	T	T
T	F	F
T	T	T

While only an approximation to the English, 100+ years of experience have shown this to be adequate for capturing *mathematical reasoning*.
(Moral: mathematical reasoning does not need all the features of English.)

Converse of $A \Rightarrow B$ is the proposition $B \Rightarrow A$

Exercise

$\boxed{\text{LLM: Problem 3.2}}$

$$
\begin{aligned}
p &= \text{“you get an HD on your final exam”} \\
q &= \text{“you do every exercise in the book”} \\
r &= \text{“you get an HD in the course”}
\end{aligned}
$$

Translate into logical notation:

(a) You get an HD in the course although you do not do every exercise in the book.

(c) To get an HD in the course, you must get an HD on the exam.

(d) You get an HD on your exam, but you don't do every exercise in this book; nevertheless, you get an HD in this course.

Exercise

LLM: Problem 3.2

$$\begin{aligned} p &= \text{“you get an HD on your final exam”} \\ q &= \text{“you do every exercise in the book”} \\ r &= \text{“you get an HD in the course”} \end{aligned}$$

Translate into logical notation:

(a) You get an HD in the course although you do not do every exercise in the book. $r \wedge \neg q$

(c) To get an HD in the course, you must get an HD on the exam. $r \Rightarrow p$

(d) You get an HD on your exam, but you don't do every exercise in this book; nevertheless, you get an HD in this course. $p \wedge \neg q \wedge r$

Unless

A *unless* B can be approximated as $\neg B \Rightarrow A$

E.g.
I go swimming unless it rains = If it is not raining I go swimming.

Correctness of the translation is perhaps easier to see in:
I don't go swimming unless the sun shines = If the sun does not shine then I don't go swimming.

Note that "I go swimming unless it rains, but sometimes I swim even though it is raining" makes sense, so the translation of "A unless B" should not imply $B \Rightarrow \neg A$.

Just in case

A just in case B usually means *A if, and only if, B*; written $A \Leftrightarrow B$

The program terminates just in case the input is a positive number.
= The program terminates if, and only if, the input is positive.

I will go swimming just in case I won't play soccer.
= If I play soccer I will not go swimming and vice versa.

It has the following truth table:

A	B	A $\Leftrightarrow$ B
F	F	T
F	T	F
T	F	F
T	T	T

Same as $(A \Rightarrow B) \wedge (B \Rightarrow A)$

The Formal Language of Propositional Logic

Let $Prop = \{p, q, r, \ldots\}$ be a set of basic propositional letters. Consider the *alphabet*

$$\Sigma = Prop \cup \{\top, \bot, \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow, (,)\}$$

The set of **formulae of propositional logic** is the smallest set of words over Σ such that

- $\top$, $\bot$ and all elements of $Prop$ are formulae
- If ϕ is a formula, then so is $\neg\phi$
- If ϕ and ψ are formulae, then so are $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \Rightarrow \psi)$, and $(\phi \Leftrightarrow \psi)$.

Convention: we often drop parentheses when there is no ambiguity. $\neg$ binds more tightly than $\wedge$ and $\vee$, which in turn bind more tightly than $\Rightarrow$ and $\Leftrightarrow$.

Finally... Some Supplementary Exercises

Exercise

$\boxed{\text{1.8.2(b) (supp)}}$ When is $(A \setminus B) \setminus C = A \setminus (B \setminus C)$?

$\boxed{\text{1.8.9 (supp)}}$ How many third powers are $\leq 1,000,000$ and end in 9? (Solve without calculator!)

Finally… Some Supplementary Exercises

Exercise

$\boxed{\text{1.8.2(b) (supp)}}$ When is $(A \setminus B) \setminus C = A \setminus (B \setminus C)$?

From Venn diagram

$(A \setminus B) \setminus C = A \cap B^c \cap C^c$; $A \setminus (B \setminus C) = (A \cap B^c) \cup (A \cap C)$.

Equality would require that $A \cap C \subseteq A \cap B^c \cap C^c$; however, these two sets are disjoint, thus $A \cap C = \emptyset$ is a necessary condition for the equality.

One verifies that $A \cap C = \emptyset$ is also a sufficient condition and that, in this case, both set expressions simplify to $A \setminus B$.

$\boxed{\text{1.8.9 (supp)}}$ How many third powers are $\leq 1,000,000$ and end in 9? (Solve without calculator!)

$n^3 = 9 \pmod{10}$ only when $n = 9 \pmod{10}$, and $n^3 \leq 1,000,000$ when $n \leq 100$. Hence all such n are $9, 19, \ldots, 99$.

Try the same question for n^4.

Quiz Rules

Practice Quiz for Homework Week 1 due Tues, 21 Feb, 5pm

Do ...

- use your own best judgement to understand & solve questions
- email us if you think Moodle is wrong (question or answer)
- discuss quizzes on the forum only **after** the deadline

Do not ...

- post specific questions about the quiz **before** the deadline
- ask us to check your answers before you submit
- agonise too much about a question that you find too difficult

NB

1. Homework and quizzes are for you to demonstrate your ability to understand and solve problems (like an exam)
2. They give you feedback on how well you have understood the contents (to prepare you for the exam)

Summary

- Notation for numbers
  $\lfloor m \rfloor$, $\lceil m \rceil$, $m|n$, $n \bmod m$, $|a|$, $[a,b]$, (a,b), gcd, lcm
- Sets and set operations
  $|A|$, $\in$, $\cup$, $\cap$, $\setminus$, $\oplus$, A^{c}, $\mathrm{Pow}(A)$, $\subseteq$, $\subset$, $\times$
- Formal languages: alphabets and words
  λ, Σ^*, Σ^+, Σ^1, $\Sigma^2, \ldots,$ $\Sigma^{\leq k}$
- Language of propositional logic
  $\wedge$, $\vee$, $\neg$, $\Rightarrow$, $\Leftrightarrow$, $\top$, $\perp$, truth tables

Coming up ...

- Ch. 2, Sec. 2.2-2.5 (Propositional calculus, Proofs)
- Ch. 10, Sec. 10.1-10.5 (Boolean algebra)