



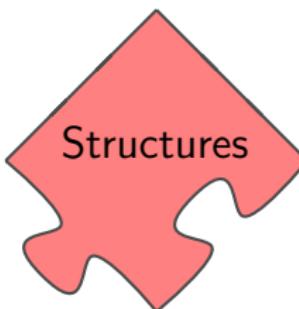
**UNSW**  
SYDNEY

# **COMP9020**

## Foundations of Computer Science

### Lecture 3: Sets and Formal Languages

# Topic 1: Structures



		[LLM]	[RW]
Week 2	Sets and Formal Languages; Set Theory	4.1, 4.2	Ch. 1
Week 3	Relations	4.4	Ch. 3
Week 4	Functions	4.3, 13.7	Ch. 3
Week 5	Graph Theory	Ch. 11, 12	Ch. 6

# Structures in Computer Science

## Sets:

- Sets are the building blocks of nearly all mathematical structures
- Data structures based around sets can be a space-efficient storage system
- Set theory is a good introduction to formal reasoning (logic)

## Formal languages:

- Formal languages are essential for compilers and programming language design
- Formal languages provide a good introduction to recursive structures (recursion and induction)

# Structures in Computer Science

## Relations:

- Relations are the building blocks of nearly all structures used in Computer Science
- Databases are collections of relations
- Any ordering is a relation
- Common data structures (e.g. graphs) are relations
- Functions/procedures/programs compute relations between their input and output

## Functions:

- Functions, methods, procedures in programming
- Computer programs “are” functions
- Graphical transformations
- Algorithmic analysis

# Structures in Computer Science

## Graphs:

- Route planning in navigation systems, robotics
- Optimisation, e.g. timetables, utilisation of network structures, bandwidth allocation
- Compilers using “graph colouring” to assign registers to program variables
- Circuit layout ([Untangle game](#))
- Determining the significance of a web page (Google’s pagerank algorithm)
- Modelling the spread of a virus in a computer network or news in social network

# Outline

Introduction to Sets

Defining Sets

Set Operations

Formal Languages

# Outline

Introduction to Sets

Defining Sets

Set Operations

Formal Languages

# Sets

## Definition

A **set** is a collection of objects (**elements**). If  $x$  is an element of  $A$  we write  $x \in A$ .

## NB

- *Elements are taken from a **universe**,  $\mathcal{U}$ , – but this can be quite complex. e.g. numbers, and sets of numbers, and sets of sets of numbers, etc.*
- *Not all “well-defined” universes are possible. e.g.*
  - *No “set of all sets” (Cantor’s paradox)*
  - *No “sets which do not contain themselves” (Russell’s paradox)*

## Sets

- A set is defined by the collection of its elements. Order and multiplicity of elements is not considered.
- We distinguish between an element and the set comprising this single element. Thus always  $a \neq \{a\}$ .
- Set  $\emptyset = \{\}$  is empty (no elements);
- Set  $\{\{\}\}$  is nonempty — it has one element.

# Subsets

## Definition

For sets  $S$  and  $T$ , we say  $S$  is a **subset** of  $T$ , written  $S \subseteq T$ , if every element of  $S$  is an element of  $T$ .

## NB

- $S \subseteq T$  includes the case of  $S = T$
- $S \subset T$  — a **proper** subset:  $S \subseteq T$  and  $S \neq T$
- $\emptyset \subseteq S$  for all sets  $S$
- $S \subseteq \mathcal{U}$  for all sets  $S$
- $\mathbb{N}_{>0} \subset \mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R}$
- An element of a set; and a subset of that set are two different concepts

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

# Outline

Introduction to Sets

Defining Sets

Set Operations

Formal Languages

## Defining sets

Sets are typically described by:

- ① Explicit enumeration of their elements

$$\begin{aligned}S_1 &= \{a, b, c\} = \{a, a, b, b, b, c\} \\&= \{b, c, a\} = \dots \text{ three elements}\end{aligned}$$

$$S_2 = \{a, \{a\}\} \text{ two elements}$$

$$S_3 = \{a, b, \{a, b\}\} \text{ three elements}$$

$$S_4 = \{\} \text{ zero elements}$$

$$S_5 = \{\{\{\}\}\} \text{ one element}$$

$$S_6 = \{\{\}, \{\{\}\}\} \text{ two elements}$$

# Defining sets

## ② Defining a subset of the universal set $\mathcal{U}$ . Including:

- Specifying the properties their elements must satisfy. A typical description involves a **logical** property  $P(x)$ . For example, with  $\mathcal{U} = \mathbb{N}$  and  $P(x) = "x \text{ is even}"$ :

$$\{x : x \in \mathbb{N} \text{ and } x \text{ is even}\} = \{0, 2, 4, \dots\}$$

- Derived sets of integers

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

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

- Using interval notation.

# Intervals

Intervals of numbers (applies to any type)

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

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

$$(-\infty, b] = \{x : x \leq b\}; \quad (-\infty, b) = \{x : x < b\}$$

$$[a, \infty) = \{x : a \leq x\}; \quad (a, \infty) = \{x : a < x\}$$

## NB

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

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

$$[m, n] = \{m, m + 1, \dots, n\}$$

# Examples

## Examples

- $[1, 5] = \{1, 2, 3, 4, 5\}$  (when  $\mathcal{U} = \mathbb{Z}$ )
- $[1, 5] = \{1, 1.1, 1.01, 1.001, \dots, 2, \dots, \pi, e, \dots\}$  (when  $\mathcal{U} = \mathbb{R}$ )
- Number of multiples of  $k$  between  $n$  and  $m$  (inclusive) in  $[n, m]$ :

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

- $0 \leq (m \% n) < n$  ( $m \% n \in [0, n)$ )

# Defining sets

## ③ Constructions from other, already defined, sets

- Union ( $\cup$ ), intersection ( $\cap$ ), complement ( $\cdot^c$ ), set difference ( $\setminus$ ), symmetric difference ( $\oplus$ )
- Power set  $\text{Pow}(X) = \{ A : A \subseteq X \}$
- Cartesian product ( $\times$ )

# Outline

Introduction to Sets

Defining Sets

Set Operations

Formal Languages

# Basic Set Operations

## Definition

$A \cup B$  – **union** ( $a$  or  $b$ ):

$$A \cup B = \{x : x \in A \text{ or } x \in B\}.$$

$A \cap B$  – **intersection** ( $a$  and  $b$ ):

$$A \cap B = \{x : x \in A \text{ and } x \in B\}.$$

$A^c$  – **complement** (with respect to a universal set  $\mathcal{U}$ ):

$$A^c = \{x : x \in \mathcal{U} \text{ and } x \notin A\}.$$

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

# Basic Set Operations

Other set operations

## Definition

$A \setminus B$  – **set difference**, relative complement ( $a$  but not  $b$ ):

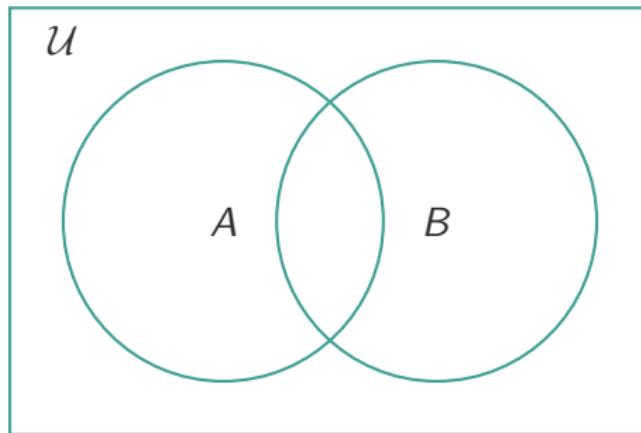
$$A \setminus B = A \cap B^c$$

$A \oplus B$  – **symmetric difference** ( $a$  and not  $b$  or  $b$  and not  $a$ ; also known as  $a$  or  $b$  exclusively;  $a$  xor  $b$ ):

$$A \oplus B = (A \setminus B) \cup (B \setminus A)$$

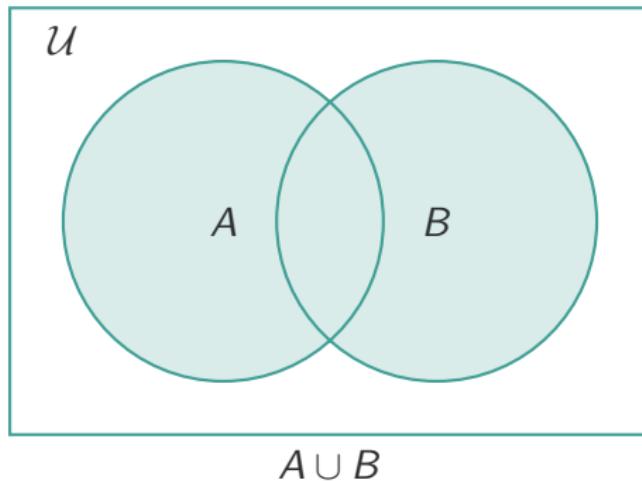
## Venn Diagrams

A **Venn Diagram** is a simple graphical approach to visualize the basic set operations.



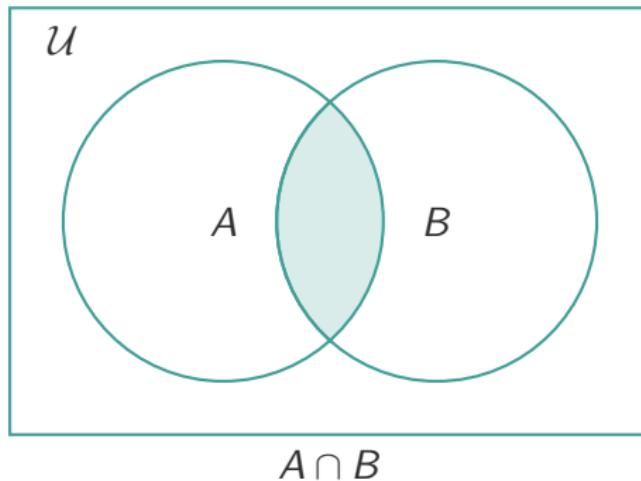
# Venn Diagrams

A **Venn Diagram** is a simple graphical approach to visualize the basic set operations.



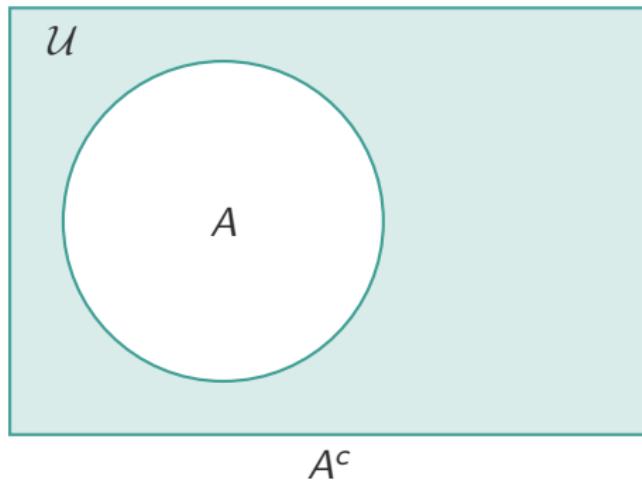
# Venn Diagrams

A **Venn Diagram** is a simple graphical approach to visualize the basic set operations.



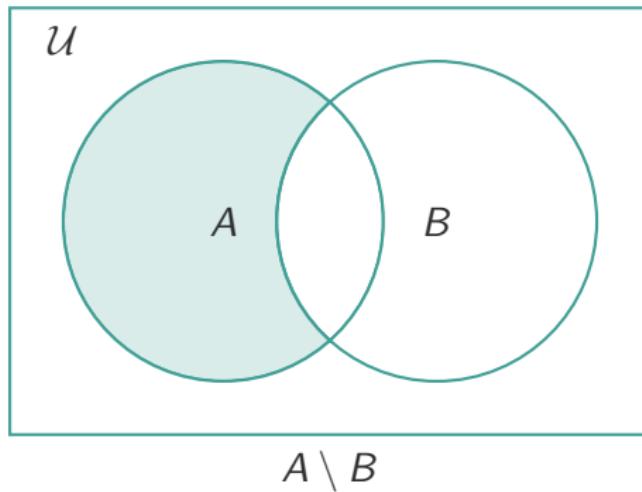
# Venn Diagrams

A **Venn Diagram** is a simple graphical approach to visualize the basic set operations.



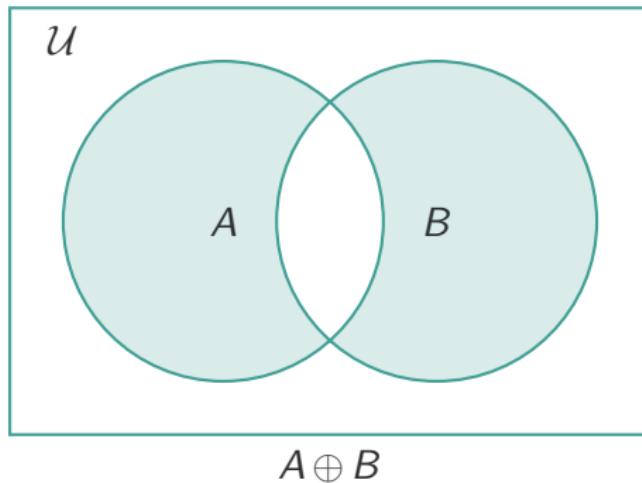
# Venn Diagrams

A **Venn Diagram** is a simple graphical approach to visualize the basic set operations.



# Venn Diagrams

A **Venn Diagram** is a simple graphical approach to visualize the basic set operations.



# Set Operations and Subset

## Fact

$$A \cup B = B \quad \text{iff} \quad A \cap B = A \quad \text{iff} \quad A \subseteq B$$

There is a correspondence between set operations and logical operators (to be discussed in Week 7).

# Exercises

## Exercises

RW: 1.4.7 (a)	$A \oplus A =$
---------------	----------------

RW: 1.4.7 (b)	$A \oplus \emptyset =$
---------------	------------------------

# Power set

## Definition

The **power set** of a set  $X$ ,  $\text{Pow}(X)$ , is the set of all subsets of  $X$

## Example

$$\text{Pow}(\{a, b\}) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$$

# Cardinality

## Definition

The **cardinality** of a set  $X$  (various notation) is the number of elements in that set.

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

## Fact

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

# Exercises

## Exercises

- $|\emptyset| \stackrel{?}{=} \dots$
- $\text{Pow}(\emptyset) \stackrel{?}{=} \dots$
- $|\text{Pow}(\emptyset)| \stackrel{?}{=} \dots$
- $\text{Pow}(\text{Pow}(\emptyset)) \stackrel{?}{=} \dots$
- $|\text{Pow}(\text{Pow}(\emptyset))| \stackrel{?}{=} \dots$
- $|\{a\}| \stackrel{?}{=} \dots$
- $\text{Pow}(\{a\}) \stackrel{?}{=} \dots$
- $|\text{Pow}(\{a\})| \stackrel{?}{=} \dots$
- $|[m, n]| \stackrel{?}{=} \dots$

# Exercises

## Exercises

RW: 1.3.2 Find the cardinalities of sets

$$(a) \quad |\left\{ \frac{1}{n} : n \in [1, 4] \right\}| = ?$$

$$(b) \quad |\left\{ n^2 - n : n \in [0, 4] \right\}| = ?$$

$$(c) \quad |\left\{ \frac{1}{n^2} : n \in \mathbb{N}_{>0} \text{ and } 2|n \text{ and } n < 11 \right\}| = ?$$

$$(d) \quad |\left\{ 2 + (-1)^n : n \in \mathbb{N} \right\}| = ?$$

# Exercises

## Exercises

RW: 1.4.8 Relate the cardinalities to  $|A \cap B|$ ,  $|A|$ ,  $|B|$

- $|A \cup B|$
- $|A \setminus B|$
- $|A \oplus B|$

# Cartesian Product

## Definition

The **Cartesian product** of two sets  $S$  and  $T$  is the set of **ordered pairs**:

$$S \times T \stackrel{\text{def}}{=} \{ (s, t) : s \in S, t \in T \}$$

The **Cartesian product** of a collection of  $n$  sets  $S_1, S_2, \dots, S_n$  is the set of **ordered  $n$ -tuples**:

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

When all the  $S_i$  are equal:

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

# Cartesian product

## Fact

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

# Examples

## Examples

Let  $A = \{0, 1\}$  and  $B = \{a, b\}$

$$\begin{aligned}A \times B &= \{(0, a), (0, b), (1, a), (1, b)\} \\&= \{(0, a), (1, a), (0, b), (1, b)\}\end{aligned}$$

$$B \times A = \{(a, 0), (b, 0), (a, 1), (b, 1)\} \neq A \times B$$

$$A^2 = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$$

$$\begin{aligned}A^3 &= \{(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), \\&\quad (1, 0, 0), (1, 0, 1), (1, 1, 0), (1, 1, 1)\}.\end{aligned}$$

## Exercise

### Exercise

Let  $A, B, C$  be sets.

Is  $A \times (B \times C) = (A \times B) \times C$ ?

# Outline

Introduction to Sets

Defining Sets

Set Operations

Formal Languages

## Formal Languages: Symbols

$\Sigma$  — **alphabet**, a finite, nonempty set

### Examples (of various alphabets and their intended uses)

$\Sigma = \{a, b, \dots, z\}$  for single words (in lower case)

$\Sigma = \{0, 1\}$  for binary integers

$\Sigma = \{0, 1, \dots, 9\}$  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.

# Formal Languages: Words

## Definition

A **word** is a finite string (sequence) of symbols from  $\Sigma$ .

The **empty word**,  $\lambda$ , is the unique word with no symbols.

## Examples

$w = aba$ ,  $w = 01101\dots 1$ , etc.

$\text{length}(w)$  — # of symbols in  $w$

$\text{length}(aaa) = 3$ ,  $\text{length}(\lambda) = 0$

The only operation on words (discussed here) is **concatenation**, written as juxtaposition  $vw$ ,  $wvw$ ,  $abw$ ,  $wbv$ , ...

## NB

$$\lambda w = w = w\lambda$$

$$\text{length}(vw) = \text{length}(v) + \text{length}(w)$$

# Examples

## Examples

Let  $w = abb$ ,  $v = ab$ ,  $u = ba$

- $vw = ababb$
- $ww = abbabb = vubb$
- $w\lambda v = abbab$
- $\text{length}(vw) = \text{length}(ababb) = 5$

# Formal Languages: Sets of words

## Definition

- $\Sigma^k$  or  $\Sigma^{=k}$ : The set of all words of length  $k$
- $\Sigma^{\leq k}$ : The set of all words of length at most  $k$
- $\Sigma^*$ : The set of all finite words
- $\Sigma^+$ : The set of all nonempty words

We often identify  $\Sigma^1 = \Sigma$

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

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

# Formal Languages: Languages

## Definition

A **language** is a subset of  $\Sigma^*$ .

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

## Example (Decimal numbers)

The “language” of all numbers written in decimal to at most two decimal places can be described as follows:

- $\Sigma = \{-, ., 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- Consider all words  $w \in \Sigma^*$  which satisfy the following:
  - $w$  contains at most one instance of  $-$ , and if it contains an instance then it is the first symbol.
  - $w$  contains at most one instance of  $.$ , and if it contains an instance then it is preceded by a symbol in  $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ , and followed by either one or two symbols in that set.
  - $w$  contains at least one symbol from  $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

## NB

*According to these rules 123, 123.0 and 123.00 are all (distinct) words in this language.*

# Examples

## Example (HTML documents)

Take

$$\Sigma = \{<\text{html}>, </\text{html}>, <\text{head}>, </\text{head}>, <\text{body}>, \dots\}.$$

The (language of) **valid HTML documents** is loosely described as follows:

- Starts with “`<html>`”
- Next symbol is “`<head>`”
- Followed by zero or more symbols from the set of HeadItems (defined elsewhere)
- Followed by “`</head>`”
- Followed by “`<body>`”
- Followed by zero or more symbols from the set of BodyItems (defined elsewhere)
- Followed by “`</body>`”
- Followed by “`</html>`”

# Exercises

## Exercises

RW: 1.3.10 Number of elements in the sets

(e)  $\Sigma^*$  where  $\Sigma = \{a, b, c\}$ ?

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

# Set Operations for Languages

Languages are sets, so the standard set operations ( $\cap$ ,  $\cup$ ,  $\setminus$ ,  $\oplus$ , etc) can be used to build new languages.

Two set operations that apply to languages uniquely:

- Concatenation (written as juxtaposition):  
$$XY = \{xy : x \in X \text{ and } y \in Y\}$$
- Kleene star:  $X^*$  is the set of words that are made up by concatenating 0 or more words in  $X$ 
  - $X^0 = \{\lambda\}$ ;  $X^{i+1} = XX^i$
  - $X^* = X^0 \cup X^1 \cup X^2 \cup \dots$

## NB

*The set of all finite words over  $\Sigma$  is the Kleene star of  $\Sigma$  (hence notation).*

# Set Operations for Languages

## Example

Let  $A = \{aa, bb\}$  and  $B = \{\lambda, c\}$  be languages over  $\Sigma = \{a, b, c\}$ .

- $A \cup B = \{\lambda, c, aa, bb\}$
- $AB = \{aa, bb, aac, bbc\}$
- $AA = \{aaaa, aabb, bbaa, bbbb\}$
- $A^* = \{\lambda, aa, bb, aaaa, aabb, bbaa, bbbb, aaaaaa, \dots\}$
- $B^* = \{\lambda, c, cc, ccc, cccc, \dots\}$
- $\{\lambda\}^* = \{\lambda\}$
- $\emptyset^* = \{\lambda\}$