

COMP9020 Week 2

Sets, Relations, and Functions

- Textbook (R & W) - Ch. 1, Sec. 1.3-1.5, 1.7; Ch. 3., Sec. 3.1

Summary of topics

- Applications in Computer Science
- Introduction to Sets
- Formal Languages
- Introduction to Relations
- Introduction to Functions

Summary of topics

- Applications in Computer Science
- Introduction to Sets
- Formal Languages
- Introduction to Relations
- Introduction to Functions

Applications of Sets, Formal Languages, Relations, and Functions

- Sets are the building blocks of nearly all mathematical structures
- 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
- Formal languages are essential for compilers and programming language design

Summary of topics

- Applications in Computer Science
- Introduction to Sets
- Formal Languages
- Introduction to Relations
- Introduction to Functions

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.
- There is only one empty set; only one set consisting of a single a ; only one set of all natural numbers.

Defining sets

Sets are typically described by:

(a) Explicit enumeration of their elements

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

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

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

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

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

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

Defining sets

(b) Defining a subset of an existing “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 $P(x) = “x \text{ is even}”$:

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

- Using interval notation. For example, with $\mathcal{U} = \mathbb{Z}$:

$$[1, 5] = \{1, 2, 3, 4, 5\}$$

- Derived sets of integers

$$2\mathbb{Z} = \{ 2x : x \in \mathbb{Z} \}$$

the even numbers

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

Sidenote: Subsets

NB

- $S \subseteq T$ — S is a **subset** of T ; includes the case of $T \subseteq T$
- $S \subset T$ — a **proper subset**: $S \subseteq T$ and $S \neq T$
- $\emptyset \subseteq S$ 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

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

Defining sets

(c) 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)

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}):

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

不相交的

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

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 without B

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

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

A union B without A and B

Set Operations

There is a correspondence between set operations and logical operators (to be discussed in Week 6).

NB

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

Exercises

Exercises

1.4.4 (d) All subsets of $\{a, b\}$:

1.4.7 (a) $A \oplus A = 0/$

1.4.7 (b) $A \oplus \emptyset = A$



Exercises

Exercises

1.4.4 (d) All subsets of $\{a, b\}$: ?

1.4.7 (a) $A \oplus A =$

1.4.7 (b) $A \oplus \emptyset =$

Exercises

Exercises

1.4.4 (d) All subsets of $\{a, b\}$: ?

1.4.7 (a) $A \oplus A = ?$

1.4.7 (b) $A \oplus \emptyset =$

Exercises

Exercises

1.4.4 (d) All subsets of $\{a, b\}$: ?

1.4.7 (a) $A \oplus A = ?$

1.4.7 (b) $A \oplus \emptyset = ?$

Cardinality

基数

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

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

Fact

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

There are only 2 status for each element in X when we do not consider orders of a set, which is True or False. So the number of possibilities is $2^{|X|}$

Exercises

Exercises

- $|\emptyset| = 0$
- $\text{Pow}(\emptyset) = 0/$
- $|\text{Pow}(\emptyset)| = 1$
- $\text{Pow}(\text{Pow}(\emptyset)) = 0/$
- $|\text{Pow}(\text{Pow}(\emptyset))| = 1$
- $|\{a\}| = 2$
- $\text{Pow}(\{a\}) = \{\{a\}, 0/\}$
- $|\text{Pow}(\{a\})| = 2$
- $|[m, n]| = N - M$

Exercises

Exercises

- $|\emptyset| = ?$
- $\text{Pow}(\emptyset) =$ 0/
- $|\text{Pow}(\emptyset)| =$ 1
- $\text{Pow}(\text{Pow}(\emptyset)) =$
- $|\text{Pow}(\text{Pow}(\emptyset))| =$
- $|\{a\}| =$
- $\text{Pow}(\{a\}) =$
- $|\text{Pow}(\{a\})| =$
- $|[m, n]| =$

Exercises

Exercises

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

Exercises

Exercises

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

Exercises

Exercises

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

Exercises

Exercises

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

Exercises

Exercises

- $|\emptyset| = ?$ 0
- $\text{Pow}(\emptyset) = ?$ 0/
- $|\text{Pow}(\emptyset)| = ?$ 1
- $\text{Pow}(\text{Pow}(\emptyset)) = ?$ $\{\{0/\}, 0/\}$
- $|\text{Pow}(\text{Pow}(\emptyset))| = ?$ 2
- $|\{a\}| = ?$ 1
- $\text{Pow}(\{a\}) = ?$ $\{0/, a\}$
- $|\text{Pow}(\{a\})| = ?$ 2
- $|[m, n]| = N - m + 1$

Exercises

Exercises

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

Exercises

1.3.2 Find the cardinalities of sets

① $|\{ \frac{1}{n} : n \in [1, 4] \}| = 4$

② $|\{ n^2 - n : n \in [0, 4] \}| = 4$

③ $|\{ \frac{1}{n^2} : n \in \mathbb{N}_{>0} \text{ and } 2|n \text{ and } n < 11 \}| = 5$

④ $|\{ 2 + (-1)^n : n \in \mathbb{N} \}| = 2$

Exercises

1.3.2 Find the cardinalities of sets

① $|\{ \frac{1}{n} : n \in [1, 4] \}| = ?$

② $|\{ n^2 - n : n \in [0, 4] \}| =$

③ $|\{ \frac{1}{n^2} : n \in \mathbb{N}_{>0} \text{ and } 2|n \text{ and } n < 11 \}| =$

④ $|\{ 2 + (-1)^n : n \in \mathbb{N} \}| =$

Exercises

1.3.2 Find the cardinalities of sets

① $|\{ \frac{1}{n} : n \in [1, 4] \}| = ?$

② $|\{ n^2 - n : n \in [0, 4] \}| = ?$

③ $|\{ \frac{1}{n^2} : n \in \mathbb{N}_{>0} \text{ and } 2|n \text{ and } n < 11 \}| =$

④ $|\{ 2 + (-1)^n : n \in \mathbb{N} \}| =$

Exercises

1.3.2 Find the cardinalities of sets

① $|\{ \frac{1}{n} : n \in [1, 4] \}| = ?$

② $|\{ n^2 - n : n \in [0, 4] \}| = ?$

③ $|\{ \frac{1}{n^2} : n \in \mathbb{N}_{>0} \text{ and } 2|n \text{ and } n < 11 \}| = ?$

④ $|\{ 2 + (-1)^n : n \in \mathbb{N} \}| =$

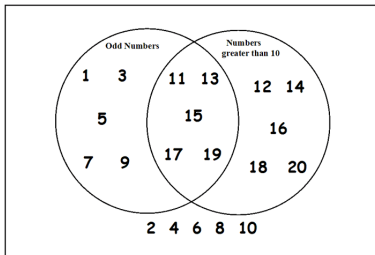
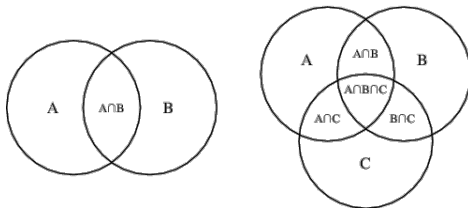
Exercises

1.3.2 Find the cardinalities of sets

- ① $|\{ \frac{1}{n} : n \in [1, 4] \}| = ?$
- ② $|\{ n^2 - n : n \in [0, 4] \}| = ?$
- ③ $|\{ \frac{1}{n^2} : n \in \mathbb{N}_{>0} \text{ and } 2|n \text{ and } n < 11 \}| = ?$
- ④ $|\{ 2 + (-1)^n : n \in \mathbb{N} \}| = ?$

Venn Diagrams

A simple graphical approach to reason about the algebraic properties of set operations.



Exercises

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

- $|A \cup B| = |A| + |B| - |A \cap B|$
hence $|A \cup B| + |A \cap B| = |A| + |B|$
- $|A \setminus B| = |A| - |A \cap B|$
- $|A \oplus B| = |A| + |B| - 2|A \cap B|$

Exercises

1.4.8 Relate the cardinalities $|A \cup B|$, $|A \cap B|$, $|A \setminus B|$, $|A \oplus B|$, $|A|$, $|B|$
?

Cartesian Product

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

有序的数对

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

$S^2 = S \times S, \quad S^3 = S \times S \times S, \dots, \quad S^n = \times_{i=1}^n S, \dots$

$\emptyset \times S = \emptyset$, for every S

$|S \times T| = |S| \cdot |T|, \quad |\times_{i=1}^n S_i| = \prod_{i=1}^n |S_i|$

Examples

Examples

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

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

$$B \times A =$$

$$A^2 =$$

$$A^3 =$$

Examples

Examples

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

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

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

$$A^2 =$$

$$A^3 =$$

Examples

Examples

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

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

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

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

$$A^3 =$$

Examples

Examples

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

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

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

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

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

Summary of topics

- Applications in Computer Science
- Introduction to Sets
- **Formal Languages**
- Introduction to Relations
- Introduction to Functions

Formal Languages: Symbols

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

Examples (of various alphabets and their intended uses)

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

$\Sigma = \{\sqcup, -, a, b, \dots, z\}$ for composite terms

$\Sigma = \{0, 1\}$ 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

word — any finite string of symbols from Σ

empty word — λ

Example

$w = aba$, $w = 01101 \dots 1$, 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 , ww , abw , wbv , \dots

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: Languages

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)

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

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

Definition

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**.

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.

Example (HTML documents)

Take $\Sigma = \{ \text{“<html>”, “</html>”, “<head>”, “</head>”, “<body>”, ...} \}$.

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

1.3.10 Number of elements in the sets (cont'd)

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

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

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

Exercises

1.3.10 Number of elements in the sets (cont'd)

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

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

Exercises

1.3.10 Number of elements in the sets (cont'd)

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

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

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

Set Operations for Languages

Example

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

- $A \cup B = \{\lambda, c, aa, bb\}$
- $AB = \{aa, bb, aac, bbc\}$
- $AA = \{aaaa, aabb, bbaa, bbbb\}$

Set Operations for Languages

Example

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

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

Set Operations for Languages

Example

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

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

Set Operations for Languages

Example

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

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

Set Operations for Languages

Example

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

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

Set Operations for Languages

Example

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

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

Summary of topics

- Applications in Computer Science
- Introduction to Sets
- Formal Languages
- Introduction to Relations
- Introduction to Functions

Relations and Functions

Relations are an abstraction used to capture the idea that the objects from certain domains (often the same domain for several objects) are *related*. These objects may

- influence one another (each other for binary relations; self(?) for unary)
- share some common properties
- correspond to each other precisely when some constraints are satisfied

Function is more concrete

Functions capture the idea of transforming *inputs* into *outputs*.

In general, functions and relations formalise the concept of interaction among objects from various domains; however, there must be a specified domain for each type of objects.

Relations

Definition

An **n-ary relation** is a subset of the cartesian product of n sets.

$$R \subseteq S_1 \times S_2 \times \dots \times S_n$$

To show tuples related by R we write:

!!

$$(x_1, x_2, \dots, x_n) \in R \quad \text{or} \quad R(x_1, x_2, \dots, x_n)$$

If $n = 2$ we have a **binary** relation $R \subseteq S \times T$ and to show pairs related by R we write:

$$(x, y) \in R \quad \text{or} \quad R(x, y) \quad \text{or} \quad xRy$$

Examples

Examples

- Equality: $=$
- Inequality: $\leq, \geq, <, >, \neq$
- Divides relation: $|$
- Element of: \in
- Subset, superset: $\subseteq, \subset, \supseteq, \supset$
- Congruence modulo n : $m = p \pmod{n}$

同余式

Database Examples

Example (Course enrolments)

S = set of CSE students

(S can be a subset of the set of all students)

C = set of CSE courses

(likewise)

E = enrolments = $\{ (s, c) : s \text{ takes } c \}$

E is a relation here

$$E \subseteq S \times C$$

In practice, almost always there are various 'onto' (nonemptiness) and 1-1 (uniqueness) constraints on database relations.

Example (Class schedule)

C = CSE courses

T = starting time (hour & day)

R = lecture rooms

S = schedule =

$$\{ (c, t, r) : c \text{ is at } t \text{ in } r \} \subseteq C \times T \times R$$

Example (sport stats)

$$R \subseteq \text{competitions} \times \text{results} \times \text{years} \times \text{athletes}$$

Defining Relations

Just as with sets R can be defined by

- explicit enumeration of interrelated k -tuples (ordered pairs in case of binary relations);
- properties that identify relevant tuples within the entire $S_1 \times S_2 \times \dots \times S_k$;
- construction from other relations.

Relation R as Correspondence From S to T

Given $R \subseteq S \times T$, $A \subseteq S$, and $B \subseteq T$.

- Relational image of A , $R(A)$:

$$R(A) \stackrel{\text{def}}{=} \{t \in T : (s, t) \in R \text{ for some } s \in A\}$$

- **Converse relation** $R^{\leftarrow} \subseteq T \times S$:

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

For symmetric relation, the converse relation is the same as itself

- Relational pre-image of B , $R^{\leftarrow}(B)$:

$$R^{\leftarrow}(B) \stackrel{\text{def}}{=} \{s \in S : (s, t) \in R \text{ for some } t \in B\}$$

Observe that $(R^{\leftarrow})^{\leftarrow} = R$.

Exercises

Exercises

Let $A = \{1, 2\}$, $B = \{2, 3\}$, $C = \{3, 4\}$, $X = [1, 4]$

- $|$ on X : **divide relation on $X \times X$**
- \in on $X \times \{A, B, C\}$:
- \subseteq^{\leftarrow} on $\{A, B, C, X\}$:
- $< (2)$ (on X):

Exercises

Exercises

Let $A = \{1, 2\}$, $B = \{2, 3\}$, $C = \{3, 4\}$, $X = [1, 4]$

- $|$ on X : ?
- \in on $X \times \{A, B, C\}$:
- \subseteq^{\leftarrow} on $\{A, B, C, X\}$:
- $< (2)$ (on X):

Exercises

Exercises

Let $A = \{1, 2\}$, $B = \{2, 3\}$, $C = \{3, 4\}$, $X = [1, 4]$

- $|$ on X : ?
- \in on $X \times \{A, B, C\}$: ?
- \subseteq^{\leftarrow} on $\{A, B, C, X\}$:
- $< (2)$ (on X):

Exercises

Exercises

Let $A = \{1, 2\}$, $B = \{2, 3\}$, $C = \{3, 4\}$, $X = [1, 4]$

- $|$ on X : ?
- \in on $X \times \{A, B, C\}$: ?
- \subseteq^{\leftarrow} on $\{A, B, C, X\}$: ?
- $< (2)$ (on X):

Exercises

the set of tuples represent a relation

Exercises

Let $A = \{1, 2\}$, $B = \{2, 3\}$, $C = \{3, 4\}$, $X = [1, 4]$

- $|$ on X :
 $\{(1, 1), (1, 2), (1, 3), (1, 4), (2, 2), (2, 4), (3, 3), (4, 4)\}$
- \in on $X \times \{A, B, C\}$:
 $\{(1, A), (2, A), (2, B), (3, B), (3, C), (4, D)\}$
- \subseteq^{\leftarrow} on $\{A, B, C, X\}$:
 $\{(A, A), (X, A), (B, B), (X, B), (C, C), (X, C), (X, X)\}$

• $< (2)$ (on X):

On \rightarrow 

Summary of topics

- Applications in Computer Science
- Introduction to Sets
- Formal Languages
- Introduction to Relations
- **Introduction to Functions**

Functions

Definition

A **function**, $f : S \rightarrow T$, is a binary relation $f \subseteq S \times T$ such that for all $s \in S$ there is *exactly one* $t \in T$ such that $(s, t) \in f$.

We write $f(s)$ for the unique element related to s .

偏函数

A **partial function** $f : S \rightharpoonup T$ is a binary relation $f \subseteq S \times T$ such that for all $s \in S$ there is *at most one* $t \in T$ such that $(s, t) \in f$. That is, it is a function $f : S' \rightarrow T$ for $S' \subseteq S$

Functions

$f : S \longrightarrow T$ describes pairing of the sets: it means that f assigns to every element $s \in S$ a unique element $t \in T$. To emphasise where a specific element is sent, we can write $f : x \mapsto y$, which means the same as $f(x) = y$

		Symbol	
S	domain of f	$\text{Dom}(f)$	(inputs)
T	co-domain of f	$\text{Codom}(f)$	(possible outputs)
$f(S)$	image of f	$\text{Im}(f)$	(actual outputs)
$= \{ f(x) : x \in \text{Dom}(f) \}$			

Important!

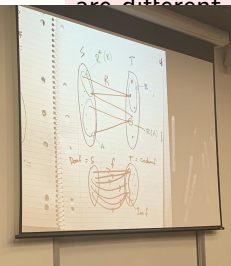
The domain and co-domain are critical aspects of a function's definition.

$$f : \mathbb{N} \rightarrow \mathbb{Z} \quad \text{given by} \quad f(x) \mapsto x^2$$

and

$$g : \mathbb{N} \rightarrow \mathbb{N} \quad \text{given by} \quad g(x) \mapsto x^2$$

are different functions even though they have the same behaviour!



Composition of Functions

Composition of functions is described as

$$g \circ f : x \mapsto g(f(x)), \quad \text{requiring } \text{Im}(f) \subseteq \text{Dom}(g)$$

Composition is associative

$$h \circ (g \circ f) = (h \circ g) \circ f, \quad \text{can write } h \circ g \circ f$$

Composition of Functions

If a function maps a set into itself, i.e. when $\text{Dom}(f) = \text{Codom}(f)$ (and thus $\text{Im}(f) \subseteq \text{Dom}(f)$), the function can be composed with itself — **iterated**

$$f \circ f, f \circ f \circ f, \dots, \quad \text{also written } \underline{f^2, f^3, \dots}$$

Identity function on S

$$\text{Id}_S(x) = x, x \in S; \text{Dom}(\text{Id}_S) = \text{Codom}(\text{Id}_S) = \text{Im}(\text{Id}_S) = S$$

$$\text{For } g : S \longrightarrow T \quad g \circ \text{Id}_S = g, \text{Id}_T \circ g = g$$

Extension: Composition of Binary Relations

If $R_1 \subseteq S \times T$ and $R_2 \subseteq T \times U$ then the composition of R_1 and R_2 is the relation:

$$R_1; R_2 := \{(a, c) : \text{there is a } b \in T \text{ such that} \\ (a, b) \in R_1 \text{ and } (b, c) \in R_2\}.$$

Note that if $f : S \rightarrow T$ and $g : T \rightarrow U$ are functions then $f; g = g \circ f$.

Exercises

Exercises

Let $f, g : \mathbb{Z} \rightarrow \mathbb{Z}$ be given by $f(n) = n^2 + 3$ and $g(n) = 5n - 11$.
What is:

- $f \circ g(n) =$
- $g \circ f(n) =$
- $g^2(n) =$

Exercises

Exercises

Let $f, g : \mathbb{Z} \rightarrow \mathbb{Z}$ be given by $f(n) = n^2 + 3$ and $g(n) = 5n - 11$.
What is:

- $f \circ g(n) = ?$
- $g \circ f(n) =$
- $g^2(n) =$

Exercises

Exercises

Let $f, g : \mathbb{Z} \rightarrow \mathbb{Z}$ be given by $f(n) = n^2 + 3$ and $g(n) = 5n - 11$.
What is:

- $f \circ g(n) = ?$
- $g \circ f(n) = ?$
- $g^2(n) =$

Exercises

Exercises

Let $f, g : \mathbb{Z} \rightarrow \mathbb{Z}$ be given by $f(n) = n^2 + 3$ and $g(n) = 5n - 11$.
What is:

- $f \circ g(n) = ?$
- $g \circ f(n) = ?$
- $g^2(n) = ?$

Composition of linear function is still linear

Properties of Functions

满射的

Function is called **surjective** or **onto** if every element of the codomain is mapped to by at least one x in the domain, i.e.

$$\text{Im}(f) = \text{Codom}(f)$$

Examples (of functions that are surjective)

- $f : \mathbb{N} \longrightarrow \mathbb{N}$ with $f(x) \mapsto x$
- Floor, ceiling

Examples (of functions that are not surjective)

- $f : \mathbb{N} \longrightarrow \mathbb{N}$ with $f(x) \mapsto x^2$
- $f : \{a, \dots, e\}^* \longrightarrow \{a, \dots, e\}^*$ with $f(w) \mapsto awe$

Injective Functions

Function is called **injective** or **1-1 (one-to-one)** if different x implies different $f(x)$, i.e. 单射的

$$\text{If } f(x) = f(y) \text{ then } x = y$$

Examples (of functions that are injective)

- $f : \mathbb{N} \longrightarrow \mathbb{N}$ with $f(x) \mapsto x$
- set complement (for a fixed universe)

Examples (of functions that are not injective)

- absolute value, floor, ceiling
- length of a word

Function is **bijjective** if it is both surjective and injective.

双射的

Converse of a function

Question

f^{\leftarrow} is a relation; when is it a function?