Sets and functions

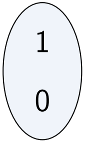
Contents

Motivation

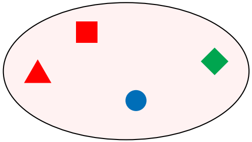
- The course teaches **functional programming**
- So it helps to know what **functions** are
- To do this we need to understand **sets**
- **Sums** and **products** of sets are similar to things we will see

Sets

A **set** is a collection of things, called its **elements**



$\{0, 1\}$



$\{\bullet, \blacklozenge, \blacktriangle, \blacksquare\}$

- The order does not matter, so $\{\bullet, \blacklozenge\} = \{\blacklozenge, \bullet\}$
- Each element is listed once, so we write $\{\bullet, \blacklozenge\}$ instead of $\{\bullet, \blacklozenge, \bullet\}$

Sets

Finite sets can be listed, for example:

- A **singleton** set: $\{*\}$ or $\{\bullet\}$
- The **Booleans**: $\mathbb{B} = \{\mathit{True}, \mathit{False}\}$

Sometimes it is easier to use **ranges**:

- $\{0, 1, \ldots, 9\} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- $\{5, 6, \ldots, 12321\}$

If a set A is finite, then we write $|A|$ for the **number of elements**

Sets

Some important **infinite sets** are:

- The **natural numbers**: $\mathbb{N} = \{0, 1, 2, \ldots\}$
- The **integers**: $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$
- The **real numbers**: $\mathbb{R}$

Another important infinite set is the set of **strings**:

- These are finite lists of elements in some set X
- For example, $[1, 1, 2, 3, 5]$ and $[24, 1, 0]$ are lists with elements in $\mathbb{N}$

The set of strings with elements in $X = \{\blacklozenge\}$ is infinite:

$$[\,], \quad [\blacklozenge], \quad [\blacklozenge, \blacklozenge], \quad [\blacklozenge, \blacklozenge, \blacklozenge], \ldots$$

Combining sets: products

The **product** of two sets A and B

- is denoted by $A \times B$
- has as elements **pairs** (a, b) with $a \in A$ and $b \in B$

Example

$$\mathbb{B} \times \{\bullet, \blacklozenge, \blacktriangle\} = \{(True, \bullet), (False, \bullet), (True, \blacklozenge), (False, \blacklozenge), (True, \blacktriangle), (False, \blacktriangle)\}$$

Remark

- It is called a **product** because $|A \times B| = |A| \times |B|$
- We can take products of more than two sets. For example, $A \times B \times C \times D$ has elements of the form (a, b, c, d)

Combining sets: union

The **union** of two sets A and B

- is denoted by $A \cup B$
- has as elements x such that x is in A or in B (or both)

Example

- $\mathbb{B} \cup \{1, 2, 3, 4\} = \{\mathit{True}, \mathit{False}, 1, 2, 3, 4\}$
- $\{●, ◆, ▲\} \cup \{●, ◆, ■\} = \{●, ◆, ▲, ■\}$

Remark

- The equality $|A \cup B| = |A| \cup |B|$ is not always true
- Unions extend to several sets, i.e. $A \cup B \cup C \cup D$

Combining sets: sum

The **sum** or **disjoint union** of two sets A and B

- is denoted by $A + B$
- has as elements pairs (a, left) with a in A, and (b, right) with b in B

Example

- $\mathbb{B} + \mathbb{B} = \{(\mathit{False}, \mathit{left}), (\mathit{True}, \mathit{left}), (\mathit{False}, \mathit{right}), (\mathit{True}, \mathit{right})\}$
- $\{\bullet, \blacklozenge, \blacktriangle\} + \{\bullet, \blacklozenge, \blacksquare\} =$
  $\{(\bullet, \mathit{left}), (\blacklozenge, \mathit{left}), (\blacktriangle, \mathit{left}), (\bullet, \mathit{right}), (\blacklozenge, \mathit{right}), (\blacksquare, \mathit{right})\}$

Remark

- It is called a **sum** because $|A + B| = |A| + |B|$
- Sums extend to several sets, i.e. $A + B + C + D$

Functions

A **function** f consists of

- a set A called the **domain**
- a set B called the **codomain**
- an assignment from each element of A to an element of B

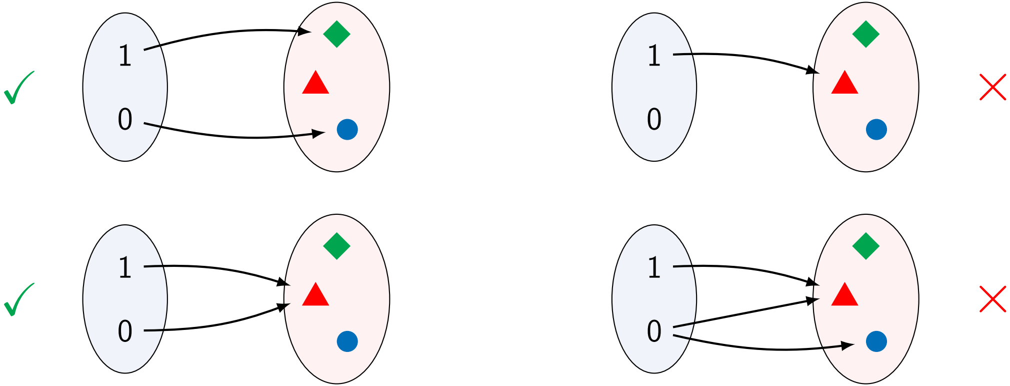
$$f :: \overset{\text{domain}}{A} \to \overset{\text{codomain}}{B}$$

$$f(\underset{\text{input}}{a}) = \underset{\text{output}}{b}$$

Functions: examples and non-examples

A **function** f consists of

- a set A called the **domain**
- a set B called the **codomain**
- an assignment from each element of A to an element of B



Functions: how to define them?

A function with a **finite domain** can be defined explicitly

Example Define $f : \mathbb{B} \to \mathbb{Z}$ by $f(False) = -4$ and $f(True) = 4$

A function with a **infinite domain** can be defined using an explanation

Example

- $minus :: \mathbb{Z} \to \mathbb{Z}$ is defined as $minus(x) = -x$
- $isPos :: \mathbb{Z} \to \mathbb{B}$ is defined as $isPos(y) = \begin{cases} True & \text{if } 0 < y \\ False & \text{otherwise} \end{cases}$

Here x and y are **variables** standing for elements of $\mathbb{Z}$

Polymorphic functions

Example The function *minus* can be defined for many different sets

- $minus :: \mathbb{Z} \to \mathbb{Z}$
- $minus :: \mathbb{R} \to \mathbb{R}$
- $minus :: \{-1, 0, 1\} \to \{-1, 0, 1\}$

A **polymorphic function** is a function that can be defined for many different sets

Example Some other polymorphic functions are

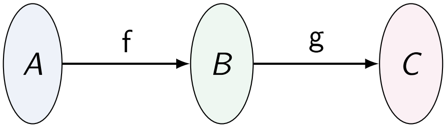
- $id :: A \to A$ defined by $id(a) = a$
- $zero :: B \to \mathbb{Z}$ defined by $zero(b) = 0$
- $proj_\ell :: A \times B \to A$ defined by $proj_\ell(a, b) = a$ (called **left projection**)
- $inj_\ell :: A \to A + B$ defined by $inj_\ell(a) = (a, left)$ (called **left injection**)

Function composition

If $f :: A \to B$ and $g :: B \to C$ then we can combine f and g to a new function

$$g \cdot f :: A \to C$$

$$(g \cdot f)(a) = g(f(a))$$



Sets of functions

Everything can be an element of a set, even functions!

Given sets A and B, the **set of functions** from A to B

- is denoted by $A \to B$
- has as elements all the functions from A to B

Remark The following are the same

- $f :: A \to B$
- f is an element of $A \to B$

Functions to functions

We extend $\to$ to more than two sets via **right associativity**, so

$$\begin{aligned} A \to B \to C &= A \to (B \to C) \\ &\neq (A \to B) \to C \end{aligned}$$

Example

- If $h :: A \to B \to C \to D$
- then $h(a) :: B \to C \to D$
- so $h(a)(b) :: C \to D$
- so $h(a)(b)(c)$ is an element of D

Remark We think of $f :: A \to B \to C$ as a function taking inputs from A and B

Combining sets (of functions)

Example

- $pair :: A \to B \to (A \times B)$ is defined by $pair(a)(b) = (a, b)$
- $compose :: (B \to C) \to (A \to B) \to A \to C$ can be defined as $compose(g)(f)(a) = g(f(a))$

Challenge Try to define a sensible polymorphic function for

$$(A + B) \to (A \to C) \to (B \to C) \to C$$

Mathematics versus functional programming

	Mathematicians	Programmers
▶	$f :: A \to B$	`f :: A -> B`
▶	$f(a) = b$	`f a = b`
▶	Infinite constructions are okay	Only finite memory
▶	Functions assign an output to every input	Functions may crash or get stuck in a loop
▶	Functions are defined by their association of input and output	One definition of a function may be better than another definition of the same function