Marius Furter
Institut für Mathematik
Universität Zürich

FS 2025

# Logic and Foundation with Haskell

## Exercise sheet 6

Recall the definition of the `Set` type from the last exercise class:

```
data Set a = Set [a] deriving Show
```

In this sheet, we will use this type to implement relations.

**Definition 1.** A *relation* $R\colon A \to B$ is a subset $R \subseteq A \times B$ of the Cartesian product.

Thinking of a relation as a set of pairs, we define

```
type Rel a b = Set (a,b)
```

The `type` keyword defines `Rel a b` as a synonym for `Set (a,b)`. Hence, you can use all the code from the previous sheet to manipulate relations.

**Exercise 1.** Using the newly defined type, define the relation $d\colon [1,\ldots,50] \to [1,\ldots,50]$ where $(a,b) \in d$ if and only if $a$ divides $b$.

**Definition 2.** The *domain* and *range* of a relation $R\colon A \to B$ are defined as

$$\begin{aligned} \mathrm{dom}(R) &:= \{a \in A \mid \exists b \in B : (a,b) \in R\}, \\ \mathrm{ran}(R) &:= \{b \in B \mid \exists a \in A : (a,b) \in R\}. \end{aligned}$$

**Exercise 2.** Write functions that compute the domain and range of a relation, both returning sets.

**Definition 3.** A relation $R\colon A \to A$ is said to be
(i) *reflexive* if $(a,a) \in R$, for all $a \in A$,
(ii) *symmetric* if $(a,b) \in R$ implies $(b,a) \in R$, for all $a, b \in A$,
(iii) *transitive* if $(a,b) \in R$ and $(b,c) \in R$ imply $(a,c) \in R$, for all $a,b,c \in A$.

**Exercise 3.** Write functions that check whether a relation is reflexive, symmetric, and transitive.

**Definition 4.** Given relations $R\colon A \to B$ and $S\colon B \to C$, their composite $S \circ R\colon A \to C$ is defined as

$$S \circ R := \{(a,c) \in A \times C \mid \exists b \in B : (a,b) \in R \wedge (b,c) \in S\}.$$

**Exercise 4.** Write a function that implements relation composition.