CHAPTER 7 · FUNCTIONS

7.1 Introduction to Functions

What a function is · equality · examples · well-defined rules

Why functions?

Much of mathematics - and computing - is the study of rules that turn an input into a single output.

Arithmetic rules	$\mathrm{div}(n, d)$, $n \bmod d$, $\lvert x \rvert$, $\lfloor x \rfloor$
Relationships	fatherOf(x), capitalOf(country)
Computation	truthTable(p), hash(key), length(string)

input → **rule** → one output

In this course we focus on discrete functions - inputs and outputs drawn from discrete sets such as integers, strings, or finite sets.

7.1 Introduction to Functions

1. **What is a function**
2. Equality of functions
3. Examples of functions
4. Checking well-defined functions

Function, image, range, and inverse image

A function ***f* : *X* → *Y*** is a relation from the **domain** X to the **co-domain** Y satisfying two rules:

1. every element of X is related to some element of Y, and
2. no element of X is related to more than one element of Y.

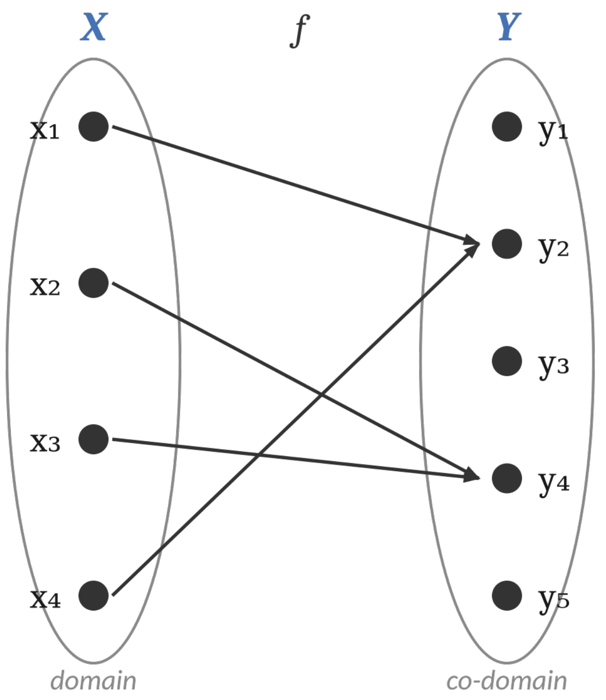
So each x in X has a **unique** image $f(x)$ in Y.

image / value of f at x: $f(x)$ - the unique output f sends x to.

range of f: $\{ y \in Y \mid y = f(x) \text{ for some } x \text{ in } X \}$ - outputs actually hit.

preimage of y: any x with f(x) = y.

inverse image of y: $\{ x \in X \mid f(x) = y \}$ - all such x.



range ⊆ co-domain: y_1, y_3, y_5 receive no arrow, yet f is still a function.

Notation and arrow diagrams

When X and Y are finite we can draw f as an arrow diagram: list X on the left, Y on the right, and draw one arrow out of each element of X.

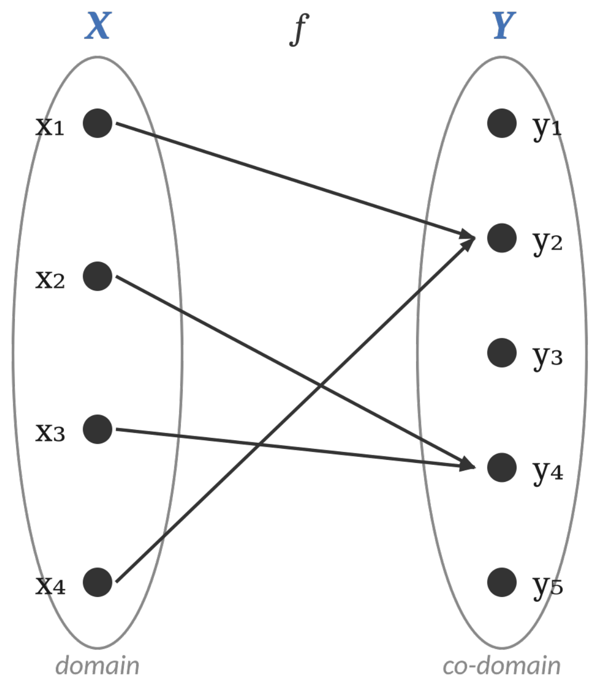
$f : X \to Y$ *"f is a function from X to Y"*

$f(x) = y$ *y is the image of x; f sends x to y*

$x \mapsto y$ *the "maps to" arrow (between elements)*

Two rules, in diagram language:

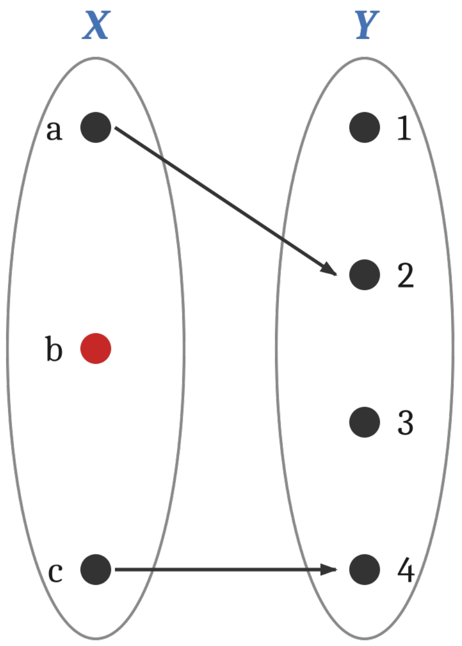
- every left element has an arrow leaving it (rule 1)
- no left element has two arrows leaving it (rule 2)



A right-hand element may receive zero, one, or many arrows - only the left side is constrained.

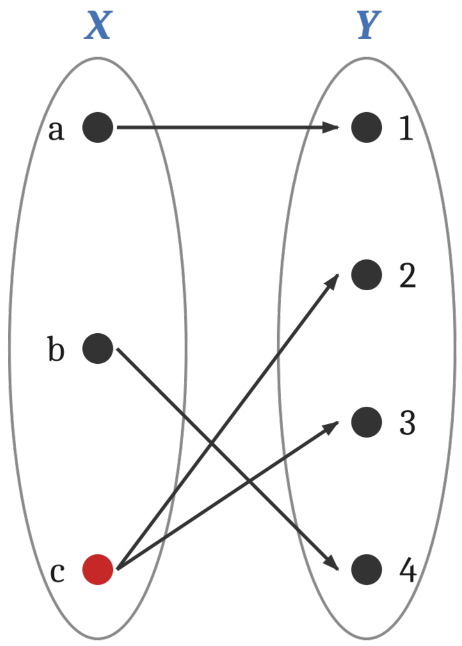
Example 7.1.1 - which arrow diagrams are functions?

X = {a, b, c}, Y = {1, 2, 3, 4}. Only one of these is a function.



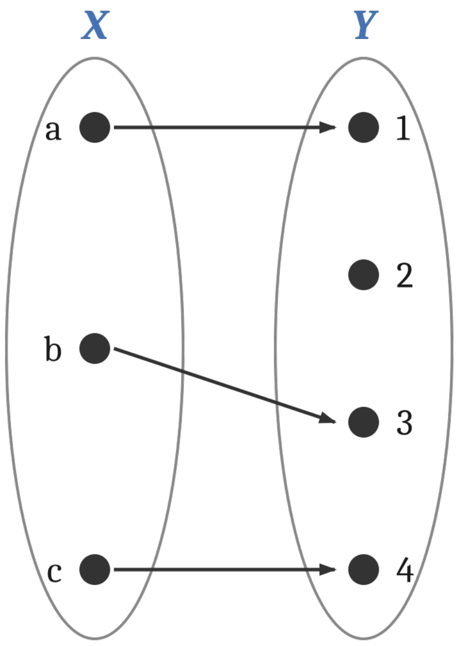
(a) **not a function**

b has no arrow (rule 1 fails)



(b) **not a function**

c has two arrows (rule 2 fails)



(c) **a function**

each input → exactly one output

Example 7.1.2 - reading a function off its diagram

Let X = {a, b, c} and Y = {1, 2, 3, 4}, with f given by the arrow diagram.

domain = {a, b, c}   co-domain = {1, 2, 3, 4}

f(a) = 2,   f(b) = 4,   f(c) = 2

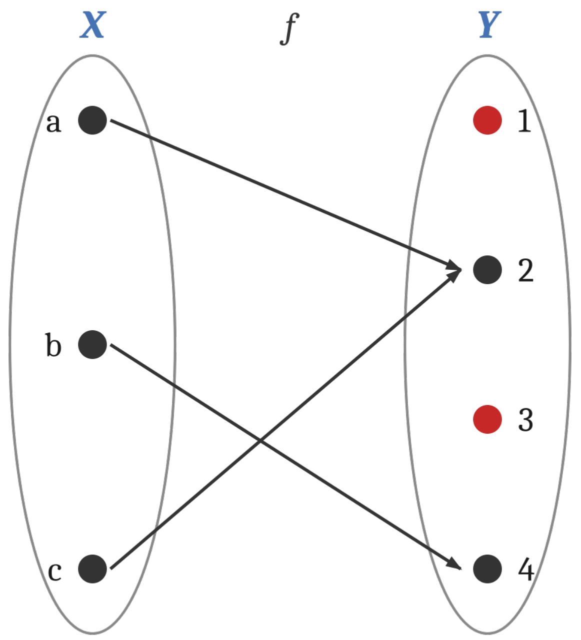
range = {2, 4}

inverse image of 2 = {a, c}

inverse image of 4 = {b}

inverse image of 1 = ∅   *(nothing maps to 1)*

f as ordered pairs = {(a, 2), (b, 4), (c, 2)}



Two arrows reach 2 (from a and c) - allowed. ***Nothing reaches 1 or 3*** *- also allowed.*

7.1 Introduction to Functions

1. What is a function
2. **Equality of functions**
3. Examples of functions
4. Checking well-defined functions

Equality of functions

Two functions are equal exactly when they have the same domain, the same co-domain, and agree on every input.

Theorem 7.1.1 (a test for function equality)

If $F: X \to Y$ and $G: X \to Y$ are functions, then $\mathbf{F = G \;\Leftrightarrow\; F(x) = G(x) \text{ for all } x \in X.}$

Why it holds (idea):

A function is its set of ordered pairs. Saying $(x, y) \in F$ is the same as saying $y = F(x)$. So F and G contain exactly the same pairs precisely when F(x) and G(x) match at every x - i.e. F = G as sets of pairs.

Consequence: the same rule written two different ways still gives *one* function - we just check it agrees at every input.

Example 7.1.3(a) - two rules, one function

On $J_3 = \{0, 1, 2\}$ define $f(x) = (x^2 + x + 1) \bmod 3$ and $g(x) = (x + 2)^2 \bmod 3$. **Does f = g?**

x	$x^2 + x + 1$	f(x)	$(x + 2)^2$	g(x)
0	1	1	4	1
1	3	0	9	0
2	7	1	16	1

The **f** and **g** columns are identical: **1, 0, 1**. By Theorem 7.1.1, **f = g** - even though the two rules look nothing alike.

Example 7.1.3(b) - F + G = G + F

For $F, G : R \to R$ define F + G and G + F by $(F + G)(x) = F(x) + G(x)$ and $(G + F)(x) = G(x) + F(x)$. **Is F + G = G + F ?**

$$\mathbf{(F + G)(x)} = F(x) + G(x) \quad \textit{by definition of F + G}$$

$$= G(x) + F(x) \quad \textit{addition of reals is commutative}$$

$$= (G + F)(x) \quad \textit{by definition of G + F}$$

This holds for every $x \in R$, so by Theorem 7.1.1, **F + G = G + F.**

Note. The commutativity comes from the real numbers, not from the functions. Where the underlying operation is not commutative, the built functions need not be equal.

7.1 Introduction to Functions

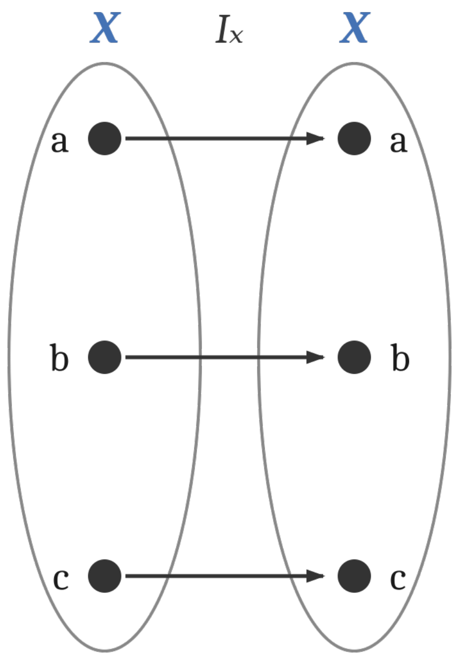
The identity function

For any set X, the **identity function** $I_X : X \to X$ is defined by

$$I_X(x) = x \quad \textit{for all } x \textit{ in } X.$$

It sends each element straight to itself — a machine that passes its input to the output unchanged. Its range equals its domain (all of X), and it is both one-to-one and onto.

Why we care: I_X is the “do nothing” function. It will return in 7.3 as the identity for composition: $f \circ I_X = f$, and $f^{-1} \circ f = I_X$.



A sequence is a function

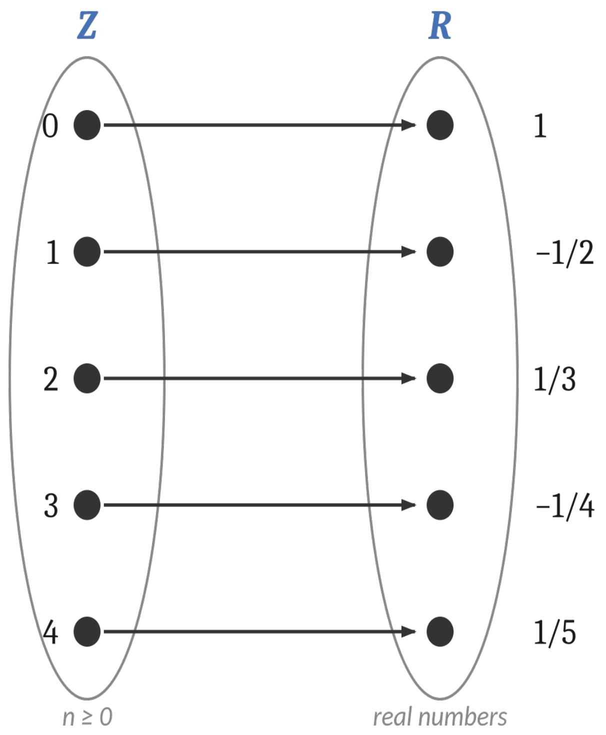
Formally…an infinite **sequence** is a function defined on the set of integers greater than or equal to a fixed integer.

The sequence

$$1,\ -1/2,\ 1/3,\ -1/4,\ 1/5,\ \ldots$$

is the function $f : Z_{\geq 0} \to R$ given by $f(n) = (-1)^n / (n + 1)$.

The same sequence, many functions. We may also index from 1: define $g : Z^+ \to R$ by $g(n) = (-1)^{n+1} / n$. Then $g(1) = 1$, $g(2) = -1/2$, … and $g(n + 1) = f(n)$: the two functions list the very same numbers, only the starting index differs.



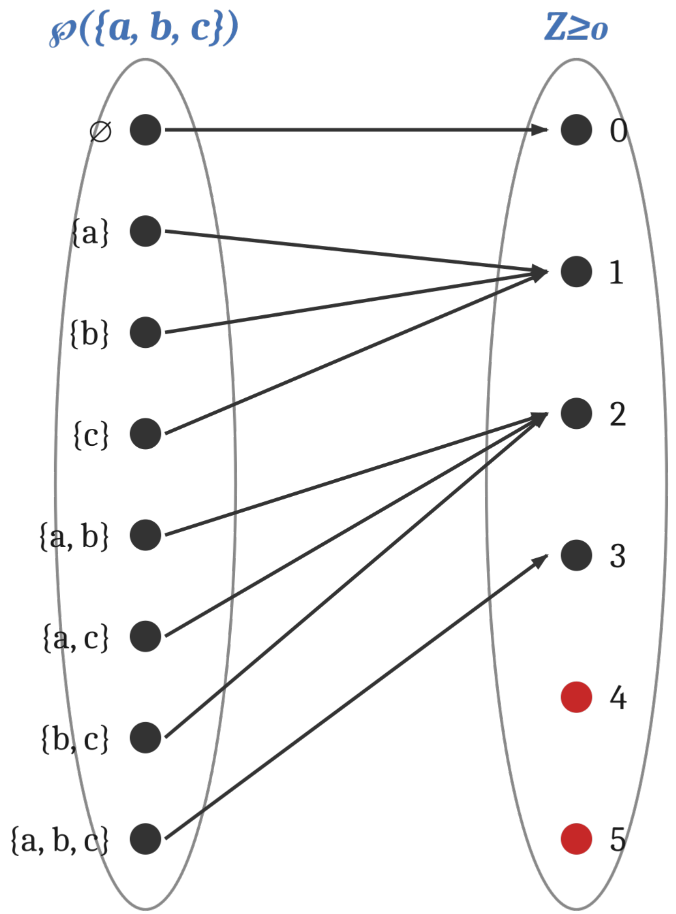
Example 7.1.6 — a function on a power set

Let P({a, b, c}) be the set of all subsets of {a, b, c}. Define

$$F : P(\{a, b, c\}) \to Z_{\geq 0}$$
$$F(X) = \text{the number of elements in } X.$$

Each subset maps to its size: ∅ → 0; the singletons → 1; the pairs → 2; {a, b, c} → 3.

Range vs co-domain again: the co-domain is all of $Z_{\geq 0}$, but the range is only {0, 1, 2, 3}. The values 4, 5, … are never reached — a 3-element set has no subset larger than 3.

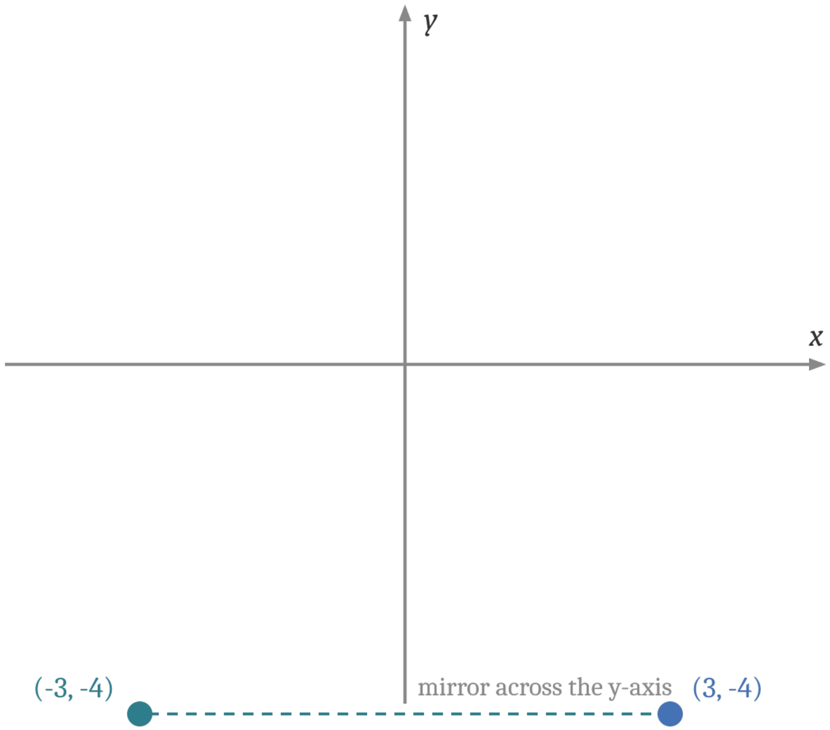


Example 7.1.7 — functions on a Cartesian product

Define M : R × R → R and R : R × R → R × R by, for all (a, b) in R × R,

$$M(a, b) = ab \qquad R(a, b) = (-a, b)$$

M multiplies the pair; **R** reflects the point across the vertical (y) axis.



$M(-1, -1) = 1$

$M(½, ½) = ¼$

$M(\sqrt{2}, \sqrt{2}) = 2$

$R(2, 5) = (-2, 5)$

$R(-2, 5) = (2, 5)$

$R(3, -4) = (-3, -4)$

Note. *We write M(a, b), dropping one set of parentheses around the input pair.*

A function on strings

Let S be the set of all strings of a's and b's (ε = the empty string). Define

$$g : S \to Z_{\geq 0} \qquad g(s) = \text{the number of a's in s.}$$

$g(\varepsilon) = 0$ *the empty string has no a's*

$g(bb) = 0$ *no a's*

$g(ababb) = 2$ *two a's*

$g(bbbaa) = 2$ *two a's*

Reading it as a function: the domain is the (infinite) set of strings, the co-domain is $Z_{\geq 0}$, and every string has exactly one a-count — so g is well defined. Strings are the basic objects of automata and formal-language theory.

Example 7.1.8 — the logarithmic function

For a base $b > 0$ with $b \neq 1$: $\log_b x$ = the exponent to which b must be raised to obtain x. $\log_b x = y \Leftrightarrow b^y = x$.

The logarithmic function with base b maps $R^+ \to R$. Evaluate:

$\mathbf{\log_3 9 = 2}$	*because $3^2 = 9$*
$\mathbf{\log_2 (1/2) = -1}$	*because $2^{-1} = 1/2$*
$\mathbf{\log_{10} 1 = 0}$	*because $10^0 = 1$*
$\mathbf{\log_2 (2^m) = m}$	*the exponent giving 2^m is m*
$\mathbf{2 \wedge (\log_2 m) = m}$	*$\log_2 m$ is the exponent giving m ($m > 0$)*

Main idea: "log" is just a name for "the exponent." Every logarithm question is secretly an exponent question. In 7.2 we will see $\log_b$ is the inverse of the exponential function b^x.

Example 7.1.11 — a Boolean function

A Boolean function takes tuples of 0's and 1's to {0, 1}. The three-place function

$$f : \{0, 1\}^3 \to \{0, 1\}$$

$$f(x_1, x_2, x_3) = (x_1 + x_2 + x_3) \text{ mod } 2$$

This is the **parity** function: the output is 1 exactly when an odd number of inputs are 1.

Domain. $\{0, 1\}^3$ has $2^3 = 8$ triples — the eight rows of the table.

Why it matters. every logic circuit / truth table is a Boolean function. This connects 7.1 back to the digital-logic work in Chapter 2.

x_1	x_2	x_3	$f = (x_1+x_2+x_3)$ mod 2
1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	1
0	0	0	0

7.1 Introduction to Functions

1. What is a function
2. Equality of functions
3. Examples of functions
4. **Checking well-defined functions**

When a rule fails to define a function

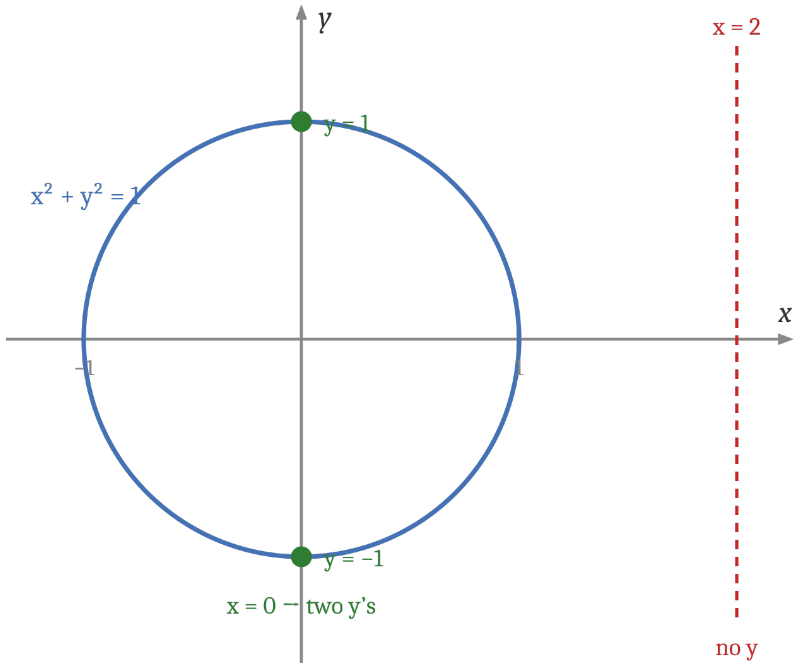
Try to define $f : R \to R$ by: f(x) is the real number y with $x^2 + y^2 = 1$.

This rule is not well defined — it breaks both function rules:

x = 2: no real y solves $4 + y^2 = 1$. *(rule 1 fails: an input with no image)*

x = 0: both y = 1 and y = −1 solve $0 + y^2 = 1$. *(rule 2 fails: two images)*

Main idea: a rule is **well defined** only if it gives every input exactly one unambiguous output. A rule that does not is simply not a function.



The vertical-line view: at x = 0 the line meets the circle twice; at x = 2 it misses entirely.

Example 7.1.12 — the same input, two names

Q is the set of rational numbers. Suppose we try to define

$$f : Q \to Z \qquad f(m\,/\,n) = m$$
(for all integers m, n with n ≠ 0)

Is f well defined? No — and here is the contradiction:

The same rational number has many names: $\mathbf{1/2 \;=\; 3/6.}$

But the formula reads off the numerator, so it gives different answers for the one number:

$$f(1/2) = 1 \quad \text{but} \quad f(3/6) = 3 \quad \Rightarrow \quad f(1/2) \neq f(3/6).$$

Contradiction: 1/2 and 3/6 are the *same* input, so a function must give them the same output. The rule does not — so f is not a function. (“well-defined function” is really redundant: being well defined is what makes it a function.)

Is it a function? — everyday relations

Take the relation from People to People. For each, ask the two rules: is everyone related to someone, and to only one?

Relation y = ___(x)	Function?	Why
FatherOf(x)	Yes	every person has exactly one (biological) father
MotherOf(x)	Yes	every person has exactly one (biological) mother
SonOf(x)	No	a person may have no son, or several — both rules can fail
BrotherOf(x)	No	a person may have zero or many brothers
WifeOf(x)	No	an unmarried person has none; the relation is not total

The test is always the same: total (everyone has an output) and single-valued (only one). Fail either, and it is a relation but not a function.

7.1 in one screen

A function $f : X \to Y$	is a relation in which every x in X has exactly one image f(x) in Y (total + single-valued).
Image, range, co-domain	image = the output f(x); range = all images; co-domain = the declared target. Range ⊆ co-domain, and the inclusion is often strict.
Preimage / inverse image	the inverse image of y is every x with f(x) = y; it may be empty, one, or many.
Equality (Thm 7.1.1)	same domain, same co-domain, and f(x) = g(x) for every x — then f = g.
Well defined	the rule must give one unambiguous output per input; if not, it is not a function.
Functions are everywhere	numbers, sequences, sets, pairs, strings, logarithms, circuits — all are functions.

Recommended exercises

Epp, 4th edition — Section 7.1 (p. 393)

2, 6, 7, 9, 18, 19, 20, 41, 42, 43

What they drill: reading domain / co-domain / range / inverse images off a diagram (2–9); equality of functions (18–20); and checking whether a rule is well defined (41–43).

Next: 7.2 — one-to-one and onto functions, bijections, inverses, and composition.