

Functions

- Some relations relate inputs with outputs produced by operating given rules over the inputs.

Inputs
(domain)

A

f

B

Outputs
(codomain)

a

Rules

$f(a)$

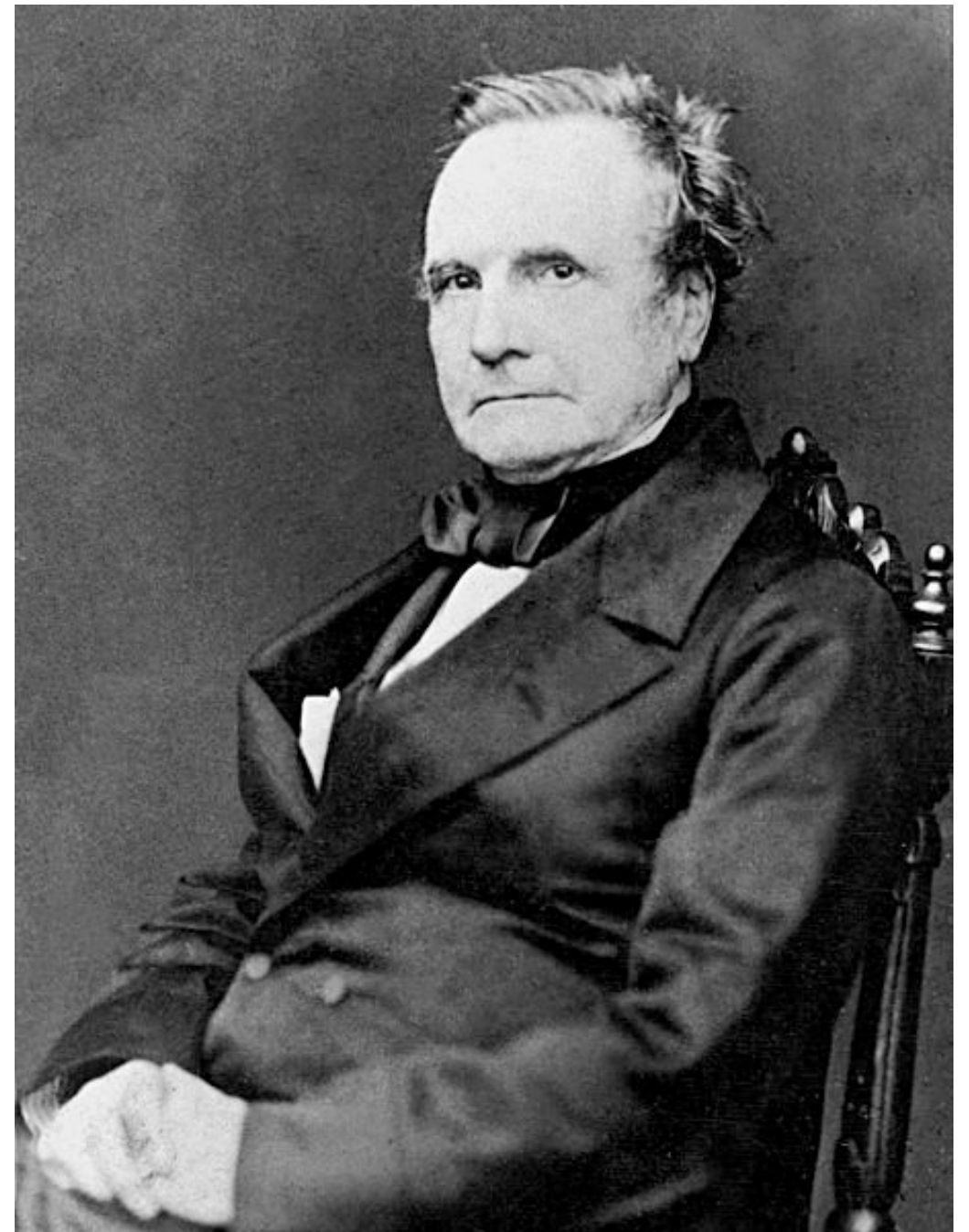
Functions

- Some relations relate inputs with outputs produced by operating given rules over the inputs.

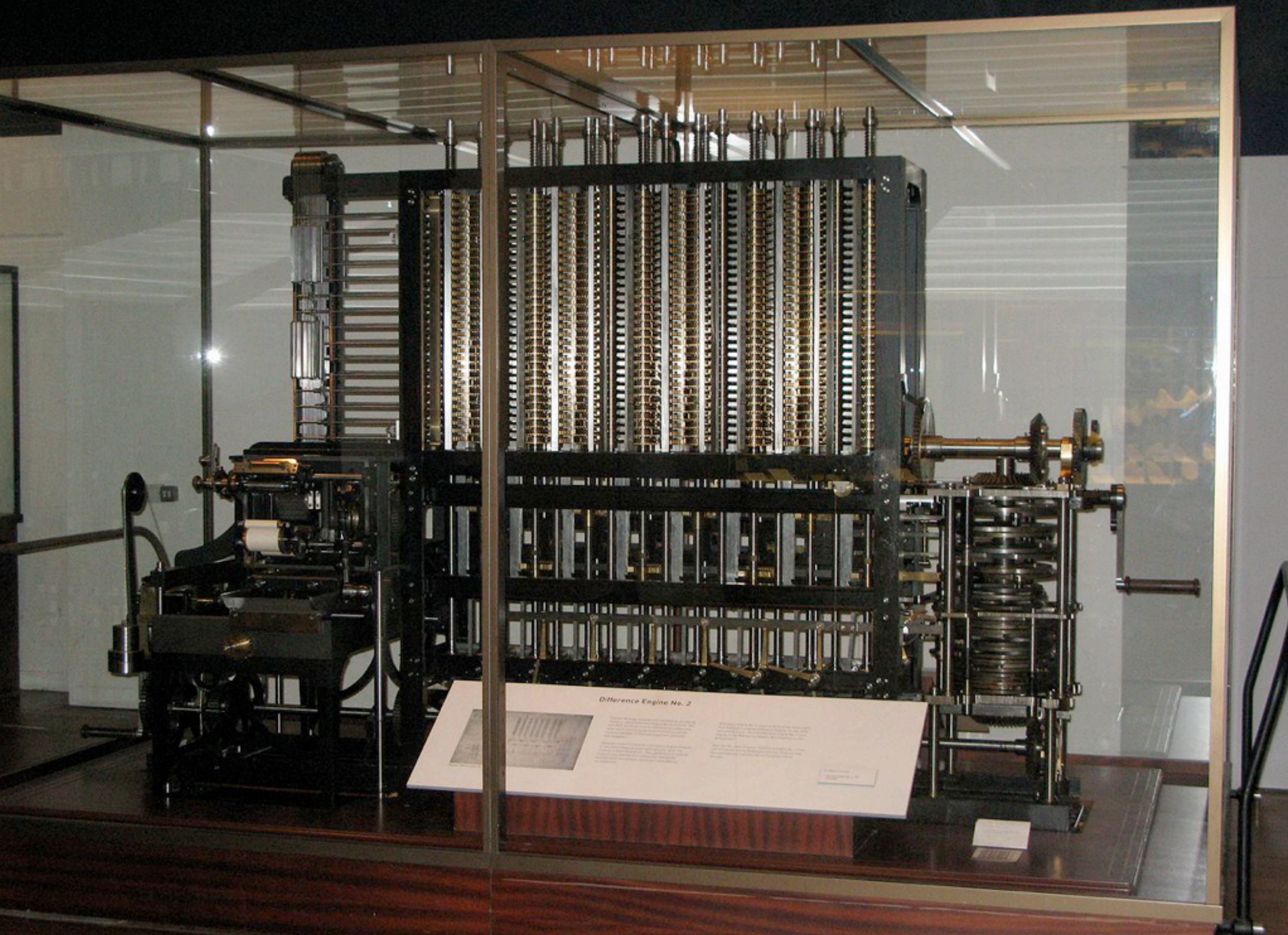


Charles Babbage

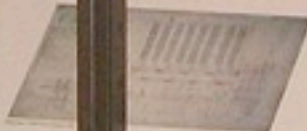
- Charles Babbage, (1791 – 1871) was an English mathematician, philosopher, inventor and mechanical engineer, who is best remembered now for originating the concept of a programmable computer.
- Babbage began working in 1822 on what he called the difference engine, made to compute values of polynomial functions. It was created to calculate a series of values automatically.



José Fiad



Difference Engine No. 2



The Difference Engine No. 2 was designed by Charles Babbage in 1842. It was a mechanical calculator designed to calculate polynomial functions, specifically the binomial coefficients, using the method of finite differences. The machine was intended to be a general-purpose calculator, capable of calculating any polynomial function. It was the second of two designs by Babbage, the first being the Difference Engine No. 1. The machine was never built, but its design was a major milestone in the history of computing.

The machine was designed to calculate the binomial coefficients, which are the numbers that appear in Pascal's triangle. It was designed to calculate these coefficients for any power of x , and for any value of x . The machine was designed to be a general-purpose calculator, capable of calculating any polynomial function. It was the second of two designs by Babbage, the first being the Difference Engine No. 1. The machine was never built, but its design was a major milestone in the history of computing.

Charles Babbage
1842

Charles Babbage
1842

Functions

- A function between two sets A and B is a relation $f \subseteq A \times B$ such that:
 - for every $a \in A$, there is some $b \in B$ such that $(a, b) \in f$
 - for every $a \in A$, if $(a, b_1) \in f$ and $(a, b_2) \in f$ then $b_1 = b_2$
- In other words, for every $a \in A$, there is one and only one $b \in B$ such that $(a, b) \in f$ – therefore we write $b = f(a)$ and $f: A \rightarrow B$

Functions

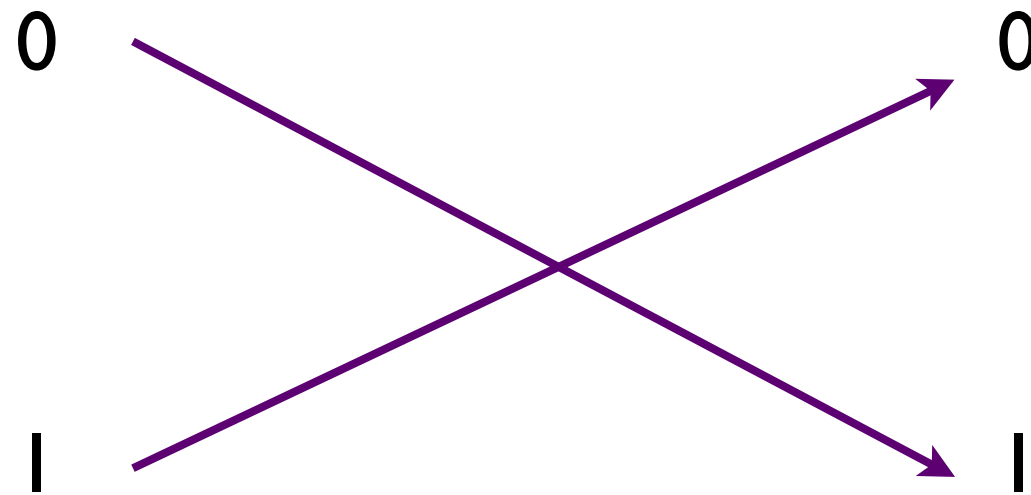
- Is this relation a function $\{0, 1\} \rightarrow \{0, 1\}$?



Yes

Functions

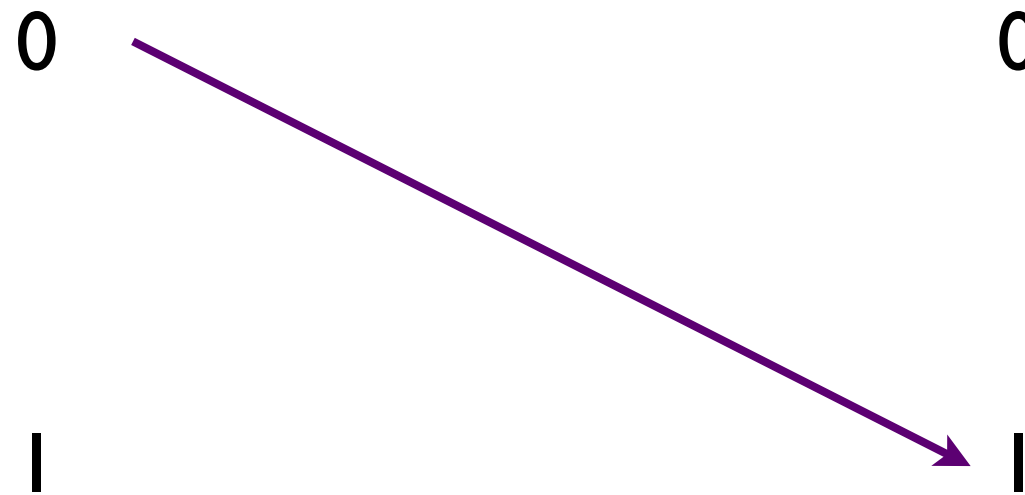
- Is this relation a function $\{0, 1\} \rightarrow \{0, 1\}$?



Yes

Functions

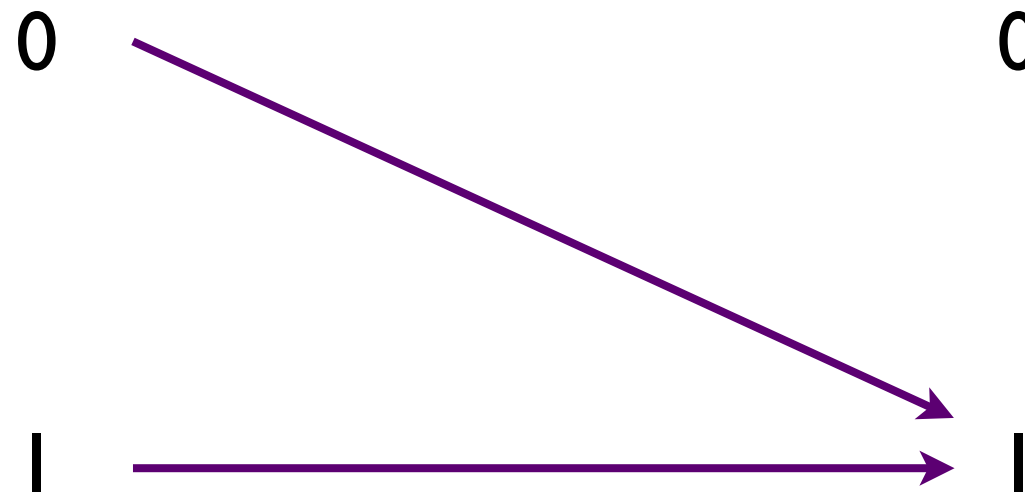
- Is this relation a function $\{0,1\} \rightarrow \{0,1\}$?



No

Functions

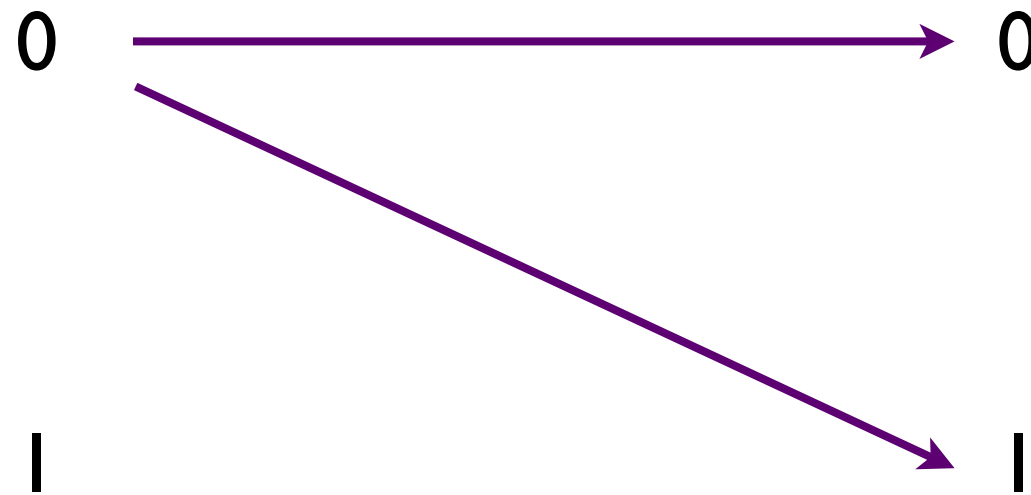
- Is this relation a function $\{0, 1\} \rightarrow \{0, 1\}$?



Yes

Functions

- Which of these relations are functions $\{0,1\} \rightarrow \{0,1\}$?



No

Terminology

- Let $f: A \rightarrow B$ be a function:
 - for every $a \in A$, $f(a)$ is the *image* of a .
 - A is the *domain* and B is the *codomain* of f .
 - The set $f(A) = \{ f(a) \mid a \in A \}$ is the *image* of A .
We have $f(A) \subseteq B$.

Invertible functions

- A function $f: A \rightarrow B$ is *invertible* if f^{-1} is a function (which is then called its *inverse*).
- If $f: A \rightarrow B$ is invertible then, for every $a \in A$,

$$f^{-1}(f(a)) = a$$

Example

- When data is transmitted errors can be introduced.
- For example, $(1, 0, 0, 1, 1, 0, 0, 1)$ is sent and $(1, 0, 1, 1, 1, 0, 0, 1)$ is received.
- How can the receiver know that there has been an error, and ask for the data to be resent?

Example

- Instead of sending a string of length 8 send one of length 9, where the last digit on the string is the sum of the previous 8 digits modulo 2.
- For example, instead of (1, 0, 0, 1, 1, 0, 0, 1), (1, 0, 0, 1, 1, 0, 0, 1, 0) is sent.
- The coding function is *bit_parity*: $\{0, 1\}^n \rightarrow \{0, 1\}^{n+1}$
- $\text{bit_parity}(b_1, b_2, \dots, b_n) = (b_1, b_2, \dots, b_n, b_{n+1})$ where

$$b_{n+1} = (b_1 + b_2 + \dots + b_n) \% 2$$

Example

But not the other way around

- The receiver checks that the last digit is correct. If it is not, the receiver knows there has been an error.
- If $(1, 0, 1, 1, 1, 0, 0, 1, 0)$ is received, there was an error.
- If there is no error, the receiver applies the inverse of *bit_parity* to decode the message:
 - $\text{bit_parity}^{-1}(b_1, b_2, \dots, b_n, b_{n+1}) = (b_1, b_2, \dots, b_n)$

Injective functions

- A function $f: A \rightarrow B$ is *injective* if, for every $a_1, a_2 \in A$,

$$f(a_1) = f(a_2) \text{ implies } a_1 = a_2$$

- We also say that f is *one-to-one*, or *1-1*.
- Question: is $f(x) = x^2 + 2$ an injective function?
- Better question: is $f(x) = x^2 + 2$ an injective function *over* \mathbb{Z} ?

Surjective functions

- A function $f: A \rightarrow B$ is *surjective* if

for every $b \in B$, there exists $a \in A$ such that $b = f(a)$

- We also say that f is *onto*.
- Question: is $f(x) = x^2 + 2$ an surjective function over \mathbb{Z} ?

Bijjective functions

- A function $f: A \rightarrow B$ is *bijjective* (or a bijection) if it is both injective and surjective.
- Theorem I:
 - A function $f: A \rightarrow B$ is invertible iff it is bijjective.
- Exercises:
 - If f is surjective then $f(A) = B$
 - If $f: A \rightarrow B$ is injective then f defines a bijection between A and $f(A)$.