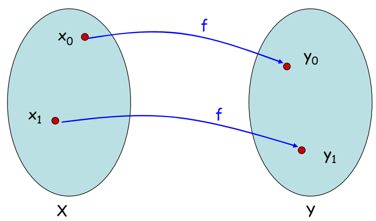
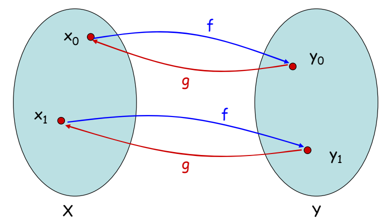


Inverse

A function f: X → Y is called invertible if and only if there exists a function g : Y → X such that

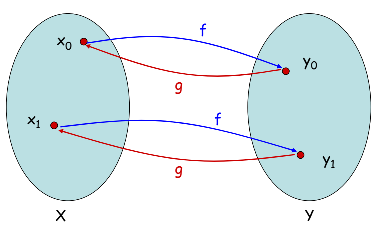
$$y = f(x) \leftrightarrow x = g(y) \text{ for all } x \in X \text{ and for all } y \in Y.$$

We call g the inverse of f and write $g = f^{-1}$.



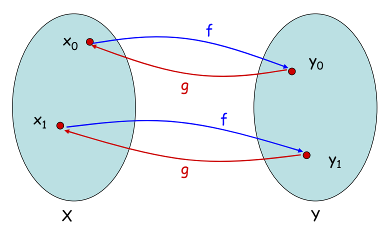
Theorem: A function f: X → Y is invertible if and only if it is a bijection, and if f is invertible, then the inverse is unique.

bijection → invertible



Theorem: A function f: X → Y is invertible if and only if it is a bijection, and if f is invertible, then the inverse is unique.

bijection → invertible



Bijection

Surjective: Everything has an incoming arrow (onto)

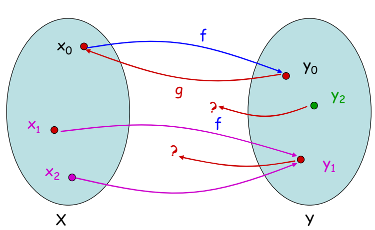
AND

Injective: Nothing has more than one incoming arrow (one-to-one)

We can construct an inverse: if f(x) = y, then g(y) = x.

Theorem: A function f: X → Y is invertible if and only if it is a bijection, and if f is invertible, then the inverse is unique.

invertible → bijection
¬bijection → ¬invertible



Not Bijection

Not Surjective: Something does not have an incoming arrow

OR

Not Injective: Something has more than one incoming arrow

In either case, we cannot construct an inverse

Back to combinatorics...

The Bijection Principle: If A and B are finite sets and there is a bijection from A to B, then |A| = |B|.

One way to count the number of elements in a set A is to show that there is a bijection from A to some other set B and count the number of elements in B.

How many subsets are there (including the empty set) of a set with n elements?

$$\sum_{k=0}^{n} C(n, k) = 2^n$$

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n$$

How many subsets are there (including the empty set) of a set with 4 elements?

$$\sum_{k=0}^{4} C(n, k) = 2^4$$

$$\sum_{k=0}^{4} \binom{n}{k} = 2^4$$

There is a one-to-one correspondence between subsets and binary numbers of length 4, and we know that there are 16 such numbers.

So by the Bijection Principle, there are 16 subsets.

n = 4

1	2	3	4
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Combinations with Repetition

Consider 7 kinds of bills: $1, $2, $5, $10, $20, $50, $100

Problem: Suppose I have a bag with lots of bills in it, and I pull out 5 bills. How many different combinations could there be?

Equivalent problem: Suppose I have 5 blank bills. How many ways can I print denominations on them?

Equivalent problem: Suppose I have 7 empty bins, labeled with the denominations. I'm going to place 5 blank bills in them, to be printed later. How many ways can I distribute the 5 bills?

Equivalent problem: How many sets can I form by selecting 5 items, with repetition allowed, from a set of 7 items?

Equivalent problem: How many integer solutions are there to the equation

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 = 5 \quad \text{where } 0 \le x_i \le 5 \text{ ?}$$

There is a one-to-one correspondence between solutions to one problem and solutions to another.

Without answering either question, why do these two questions have the same answer?

- How many sets of size 2 can be made choosing elements from {1, 2, 3, 4, 5, 6, 7, 8, 9}?
- How many sets of size 7 can be made choosing elements from {1, 2, 3, 4, 5, 6, 7, 8, 9}?

Because every solution to one of the questions also specifies a solution to the other.

{3, 8} ⇔ {1, 2, 4, 5, 6, 7, 9}
{5, 7} ⇔ {1, 2, 3, 4, 6, 8, 9}

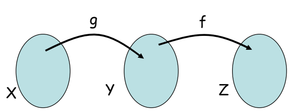
$$\binom{n}{r} = \binom{n}{n-r}$$

Functions

successor(n) = n + 1

sqr(x) = x^2

sqr(successor(n)) = $(n+1)^2$

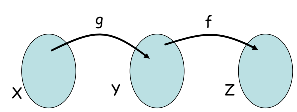


Functions

successor(n) = n + 1

sqr(x) = x^2

sqr(successor(n)) = $(n+1)^2$

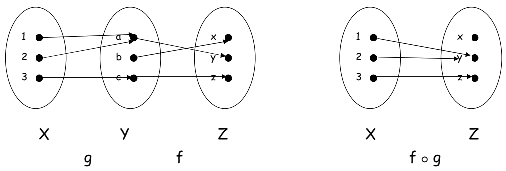


Let g be a function $g : X \to Y$ and f be a function $f : Y \to Z$.

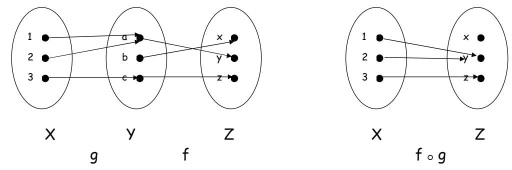
A new function (denoted by $f \circ g$) is defined by the following rule:

For all $x \in X$, $(f \circ g)(x) = f(g(x))$.

This is called the **composition of f and g**.



Is g ∘ f defined?



Is g ∘ f defined?

No, the co-domain of f is not the domain of g.

f : Y → Z   g : X → Y

f(x) = 2x + 3

g(x) = 3x + 2

f ∘ g(x) = f(g(x)) = f(3x + 2) = 6x + 7

g ∘ f(x) = g(f(x)) = g(2x + 3) = 6x + 11

So composition is not commutative.

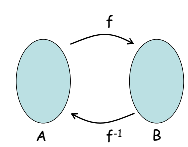
Suppose A is a set. The identity function on A is the function $\iota_A : A \to A$ such that for all $x \in A$, $\iota_A(x) = x$.

So if f: A → B is a bijection, then

$f^{-1} \circ f$ is ι_A

and

$f \circ f^{-1}$ is ι_B



More Theorems

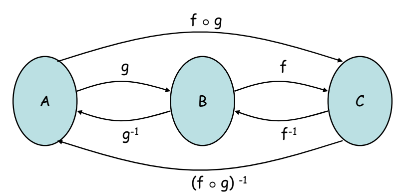
Theorem: The composition of functions is associative.
That is, suppose that functions f, g, and h are such that
h : A→B, g : B→C and f : C→D. Then f ∘ (g ∘ h) = (f ∘ g) ∘ h.

Theorem: If g : A→B and f : B→C are bijections, then
f ∘ g is a bijection and $(f \circ g)^{-1} = g^{-1} \circ f^{-1}$

Theorem: If g : A→B and f : B→C are bijections, then
f ∘ g is a bijection and $(f \circ g)^{-1} = g^{-1} \circ f^{-1}$

(f ∘ g) : A→C
(f ∘ g)(x) = f(g(x))
(f ∘ g) is a bijection is a iff every element of C has precisely one pre-image

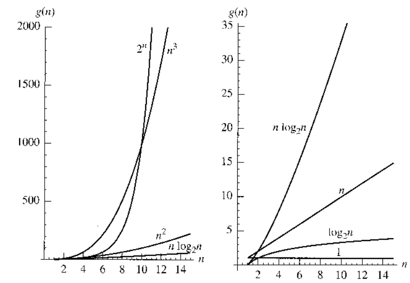
If y ∈ C, then y has precisely one pre-image under f in B, call it y'
Since y' ∈ B, y' has precisely one pre-image under g in A, call it y''
y'' is the unique pre-image of y under (f ∘ g)



Functions

When you study the analysis of algorithms, the following functions will be interesting, because they can often be used to characterize the growth of running time based on the growth of the size of the data set.

$f(n) = 1$
$f(n) = \log n$
$f(n) = n$
$f(n) = n \log n$
$f(n) = n^2$
$f(n) = 2^n$
$f(n) = n!$



	n = 10,000
$\log_2(n)$	13 nanoseconds
n	0.00001 seconds
$n\log_2(n)$	0.00013 seconds
n^2	0.1 seconds
n^3	16 min. 40 seconds
2^n	~6 x 102993 years

	n = 10,000	n = 100,000
$\log_2(n)$	13 nanoseconds	17 nanoseconds
n	0.00001 seconds	0.0001 seconds
$n\log_2(n)$	0.00013 seconds	0.00166 seconds
n^2	0.1 seconds	10.0 seconds
n^3	16 min. 40 seconds	~11 days 14 hours
2^n	~6 x 102993 years	too long to contemplate