Discrete Structures for Computer Science

William Garrison
bill@cs.pitt.edu
6311 Sennott Square

Lecture #9: Set Identities and Functions

University of Pittsburgh

Based on materials developed by Dr. Adam Lee

Today's Topics

Set identities

- Methods of proof
- Relationships to logical equivalences

Functions

- Important definitions
- Relationships to sets, relations
- Specific functions of particular importance

Set identities help us manipulate complex expressions

Recall from last lecture that set operations bear a striking resemblance to logical operations

- Disjunction ($\vee$) and set union ($\cup$)
- Conjunction ($\wedge$) and set intersection ($\cap$)
- Negation ($\neg$) and complement ($\overline{}$)

Just as logical equivalences helped us manipulate logical expressions, set identities help us simplify and understand complex set definitions.

Some important set identities

Identity	*Name*
	Identity laws
	Domination laws
	Idempotent laws
	Complementation law
	Commutative laws
	Associative laws

We don't have commutative or associative laws for set difference!

Some important set identities

Identity	*Name*
	Distributive laws
	DeMorgan's laws
	Absorption laws
	Complement laws

There are many ways to prove set identities

Today, we'll discuss four common methods:

1. Membership tables
2. Logical argument
3. Using set builder notation
4. Applying other known set identities

Membership tables allow us to write proofs like we did using truth tables!

The membership table for an expression has columns for sub-expressions and rows to indicate the ways in which an arbitrary element may or may not be included.

Example: A membership table for set intersection

A	B	A ∩ B
1	1	1
1	0	0
0	1	0
0	0	0

An element is in A ∩ B iff it is in both A and B

Prove that $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

A	B	C	B∪C	A ∩ (B ∪ C)	A ∩ B	A ∩ C	(A ∩ B) ∪ (A ∩ C)
1	1	1					
1	1	0					
1	0	1					
1	0	0					
0	1	1					
0	1	0					
0	0	1					
0	0	0					

Since the appropriate columns of the membership table are the same, we can conclude that $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$. ❑

Sometimes, it's easier to make a logical argument about a set identity

Recall: $A = B$ iff $A \subseteq B$ and $B \subseteq A$

As a result, we can prove a set identity by arguing that each side of the equality is a subset of the other.

Example: Prove that $\overline{A \cap B} = \overline{A} \cup \overline{B}$

1. First prove that $\overline{A \cap B} \subseteq \overline{A} \cup \overline{B}$
2. Then prove that $\overline{A} \cup \overline{B} \subseteq \overline{A \cap B}$

Let's see how this is done…

Prove that $\overline{A \cap B} = \bar{A} \cup \bar{B}$

Prove that $\overline{A \cap B} = \overline{A} \cup \overline{B}$

Since we have shown $\overline{A \cap B} \subseteq \overline{A} \cup \overline{B}$ and $\overline{A} \cup \overline{B} \subseteq \overline{A \cap B}$, we have shown that $\overline{A \cap B} = \overline{A} \cup \overline{B}$

Note: Differences between $\subseteq$ and $\in$

Recall that $A \subseteq B$ if A is a subset of B, whereas $a \in A$ means that a is an element of A.

Examples:

- Is $\{1\} \in \{1, 2, 3\}$?
- Is $\{1\} \subseteq \{1, 2, 3\}$?
- Is $1 \in \{1, 2, 3\}$?
- Is $\{2, 3\} \subseteq \{1, \{2, 3\}, \{4, 5\}\}$?
- Is $\{2, 3\} \in \{1, \{2, 3\}, \{4, 5\}\}$?
- Is $\emptyset \in \{1, 2, 3\}$?
- Is $\emptyset \subseteq \{1, 2, 3\}$?

Be careful when computing power sets

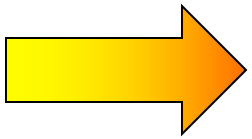
Question: What is P({1, 2, {1, 2}})?

Note: The set {1, 2, {1, 2}} has three elements

- 1
- 2
- {1, 2}

So, we need all combinations of those elements:

- ∅
- {1}
- {2}
- {{1,2}}
- {1, 2}
- {1, {1,2}}
- {2, {1,2}}
- {1, 2, {1, 2}}



$$\therefore P(\{1, 2, \{1,2\}\}) = \{\emptyset, \{1\}, \{2\}, \{\{1,2\}\}, \{1, 2\}, \{1, \{1,2\}\}, \{2, \{1,2\}\}, \{1, 2, \{1,2\}\}\}$$

This power set has $2^3 = 8$ elements.

We can use set builder notation and logical definition to make very precise proofs

Example: Prove that $\overline{A \cap B} = \overline{A} \cup \overline{B}$

Proof:

1.	$\overline{A \cap B} = \{x \mid x \notin A \cap B\}$	Def'n of complement
2.	$= \{x \mid \neg(x \in (A \cap B))\}$	Def'n of $\notin$
3.	$= \{x \mid \neg(x \in A \wedge x \in B)\}$	Def'n of $\cap$
4.	$= \{x \mid \neg(x \in A) \vee \neg(x \in B)\}$	DeMorgan's law
5.	$= \{x \mid x \notin A \vee x \notin B\}$	Def'n of $\notin$
6.	$= \{x \mid x \in \overline{A} \vee x \in \overline{B}\}$	Def'n of complement
7.	$= \{x \mid x \in \overline{A} \cup \overline{B}\}$	Def'n of $\cup$
8.	$= \overline{A} \cup \overline{B}$	Set builder notation

❑

We can also construct proofs by repeatedly applying known set identities

Example: Prove that $\overline{A \cup (B \cap C)} = (\overline{C} \cup \overline{B}) \cap \overline{A}$

Proof:

1.	$\overline{A \cup (B \cap C)}$	$= \overline{A} \cap \overline{(B \cap C)}$	DeMorgan's law
2.		$= \overline{A} \cap (\overline{B} \cup \overline{C})$	DeMorgan's law
3.		$= (\overline{B} \cup \overline{C}) \cap \overline{A}$	Commutative law
4.		$= (\overline{C} \cup \overline{B}) \cap \overline{A}$	Commutative law

❑

Note how similar this process is to that of proving logical equivalences using known logical equivalences.

In-class exercises

Problem 1: Prove DeMorgan's law for complement over intersection using a membership table.

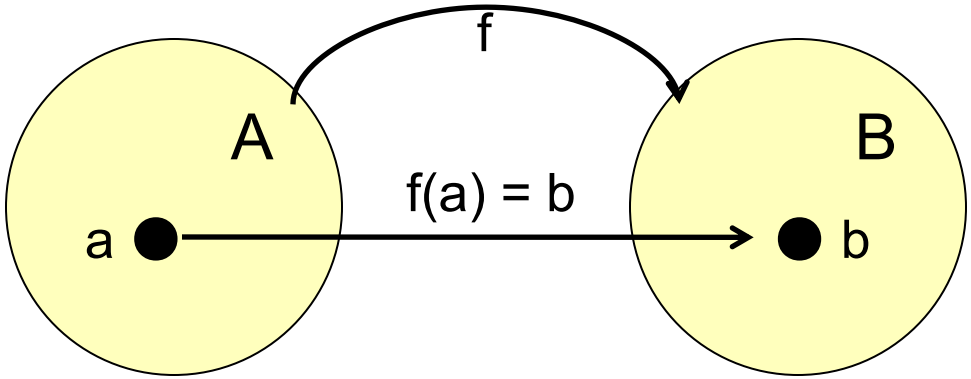
Problem 2: Prove the complementation law using set builder notation.

Sets give us a way to formalize the concept of a function

Definition: Let A and B be nonempty sets. A function, f, is an assignment of exactly one element of set B to each element of set A.

Note: We write $f : A \rightarrow B$ to denote that f is a function from A to B

Note: We say that $f(a) = b$ if the element $a \in A$ is mapped to the unique element $b \in B$ by the function f

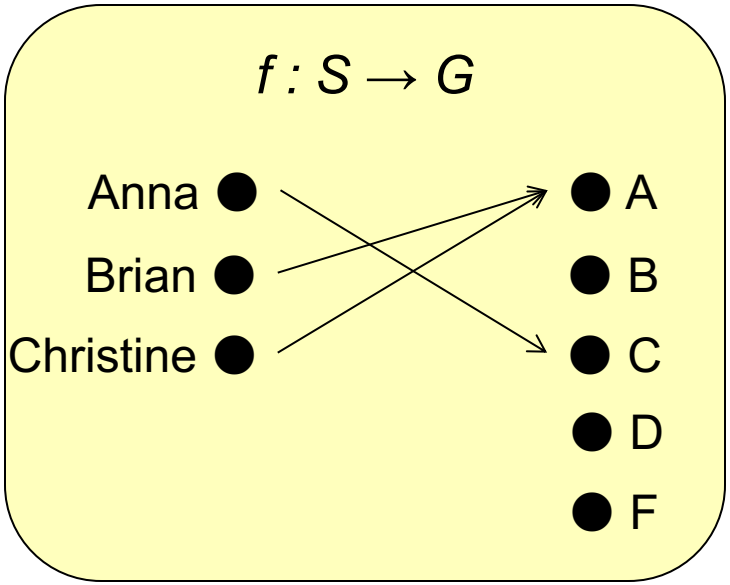


Functions can be defined in a number of ways

1. Explicitly
   - $f : Z \rightarrow Z$
   - $f(x) = x^2 + 2x + 1$

2. Using a programming language
   - int min(int x, int y) = { x < y ? return x : return y; }

3. Using a relation
   - Let S = {Anna, Brian, Christine}
   - Let G = {A, B, C, D, F}

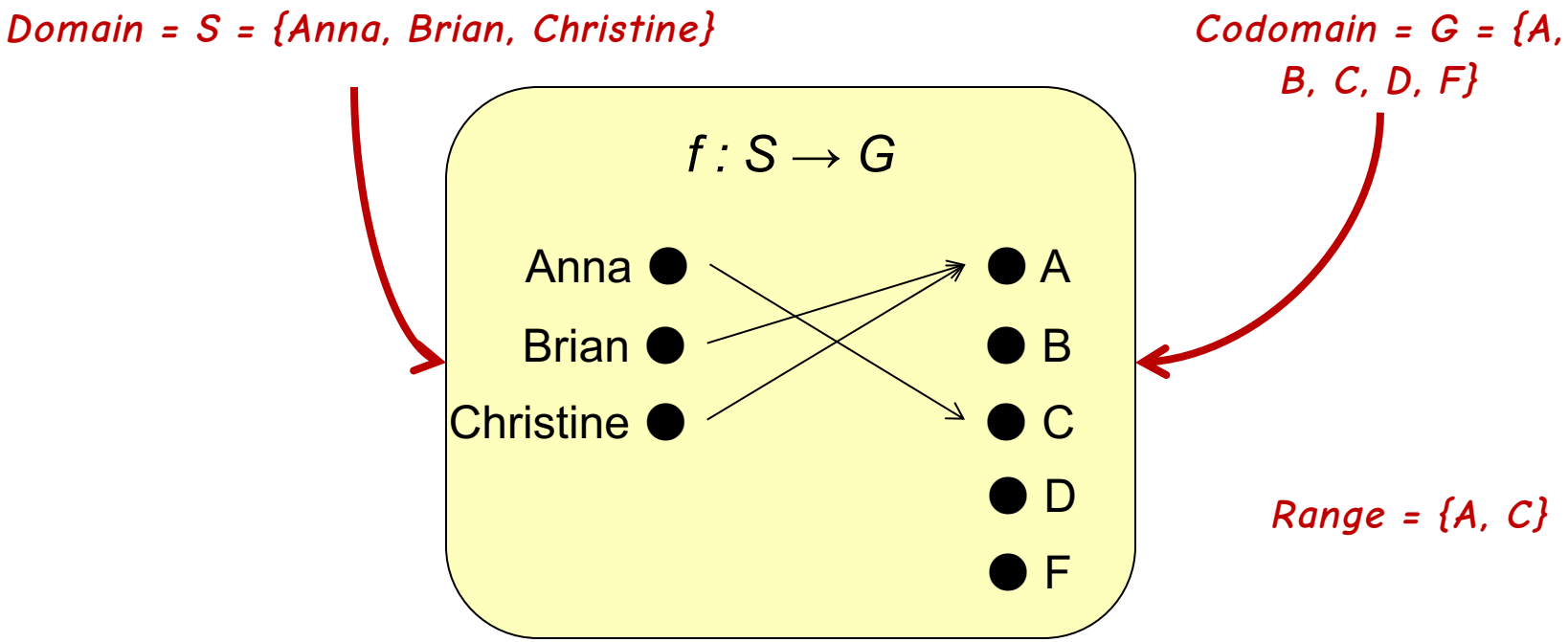


More terminology

The domain of a function is the set that the function maps from, while the codomain is the set that is mapped to

If f(a) = b, b is called the image of a, and a is called the preimage of b

The range of a function f : A → B is the set of all images of elements of A



What are the domain, codomain, and range of the following functions?

1. $f : \mathbf{Z} \rightarrow \mathbf{Z}$, $f(x) = x^3$
   - Domain:
   - Codomain:
   - Range:

2. $g : \mathbf{R} \rightarrow \mathbf{R}$, $g(x) = x - 2$
   - Domain:
   - Codomain:
   - Range:

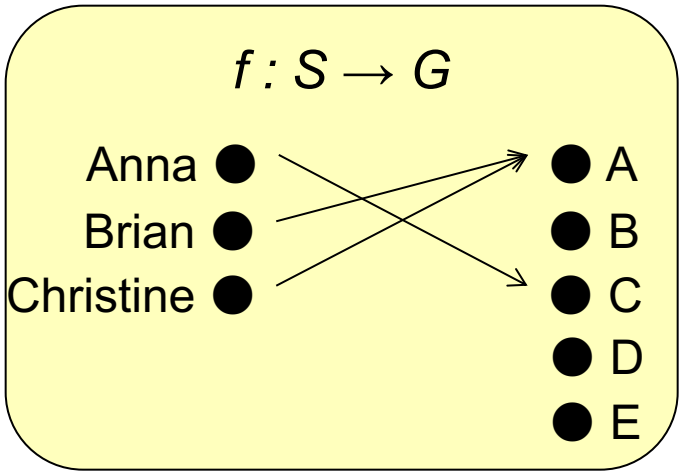
3. `int foo(int x, int y) = { return (x*y)%2; }`
   - Domain:
   - Codomain:
   - Range:

A one-to-one function never assigns the same image to two different elements

Definition: A function $f : A \rightarrow B$ is one-to-one, or injective, iff $\forall x,y \in A\ [(f(x) = f(y)) \rightarrow (x = y)]$

Are the following functions injections?

- $f : \mathbf{R} \rightarrow \mathbf{R}$, $f(x) = x + 1$
- $f : \mathbf{Z} \rightarrow \mathbf{Z}$, $f(x) = x^2$
- $f : \mathbf{R}^+ \rightarrow \mathbf{R}^+$, $f(x) = \sqrt{x}$
- $f : S \rightarrow G$

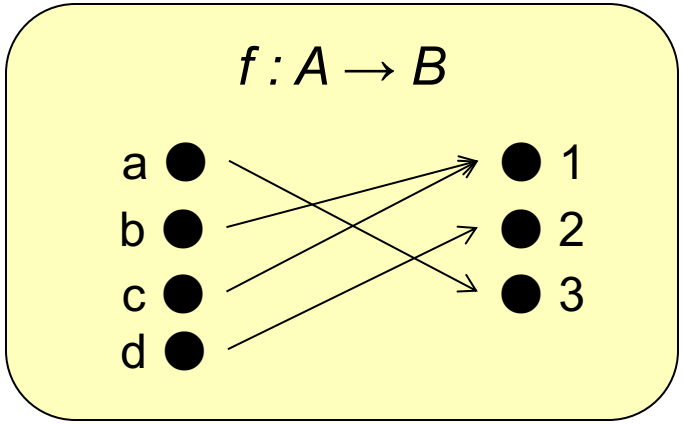


An onto function "uses" every element of its codomain

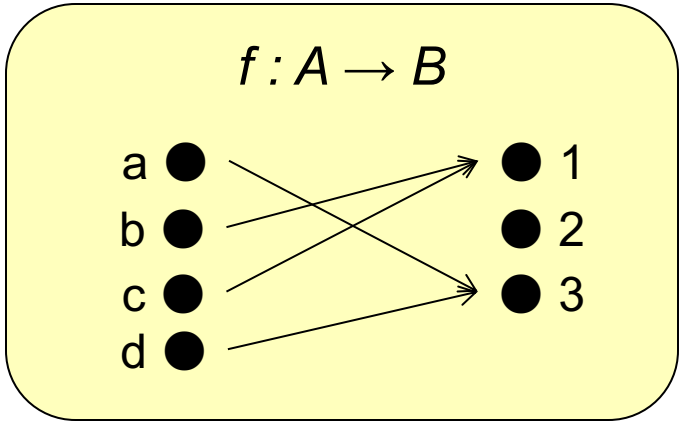
Definition: We call a function $f : A \to B$ onto, or surjective, iff for every element $b \in B$, there is some element $a \in A$ such that $f(a) = b$.

Think about an onto function as "covering" the entirety of its codomain.

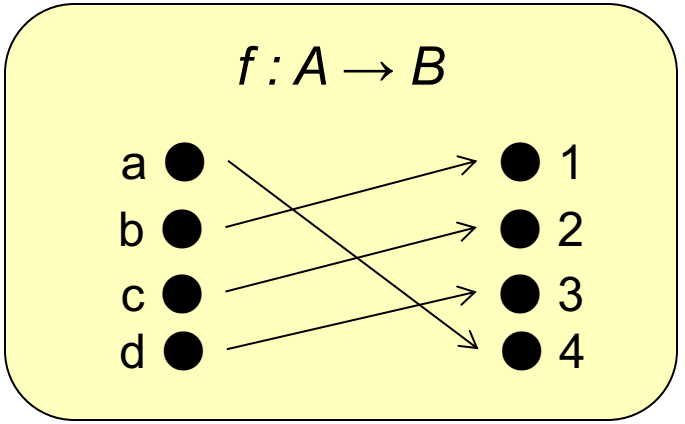
The following function is a surjection:



Are the following functions one-to-one, onto, both, or neither?

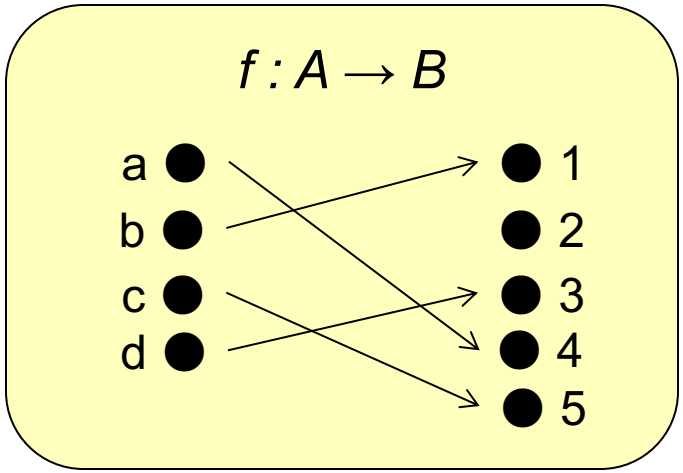


Neither!

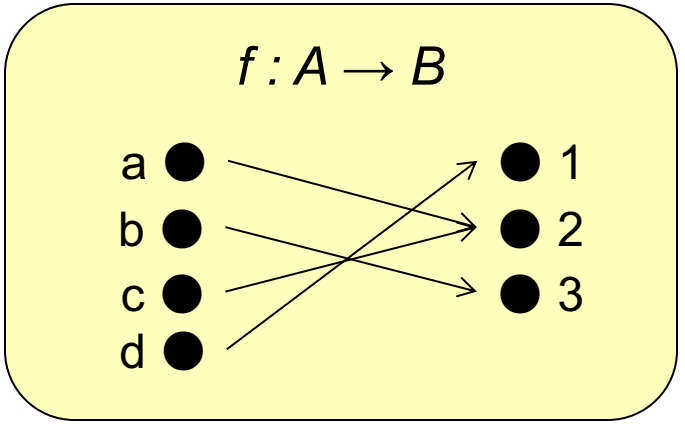


One-to-one and onto

(Aside: Functions that are both one-to-one and onto are called *bijections*)



One-to-one

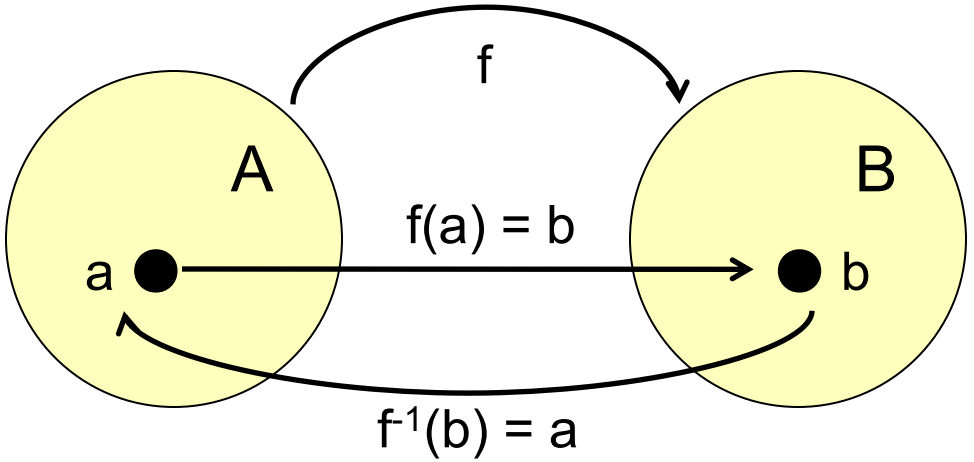


Onto

Bijections have inverses

Definition: If $f : A \to B$ is a bijection, the inverse of f is the function $f^{-1} : B \to A$ that assigns to each $b \in B$ the unique value $a \in A$ such that $f(a) = b$. That is, $f^{-1}(b) = a$ iff $f(a) = b$.

Graphically:



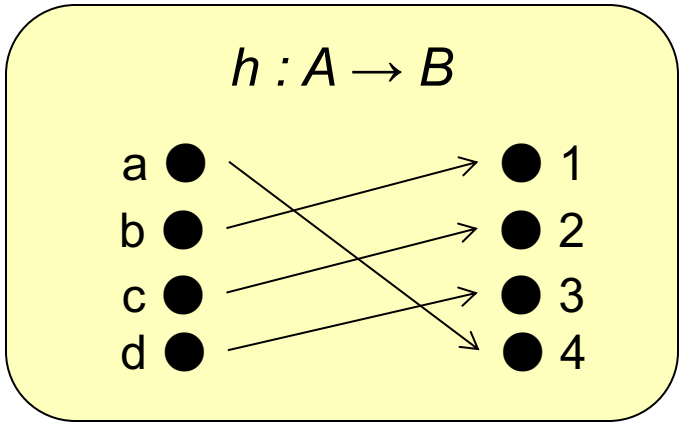
Note: Only a bijection can have an inverse. (Why?)

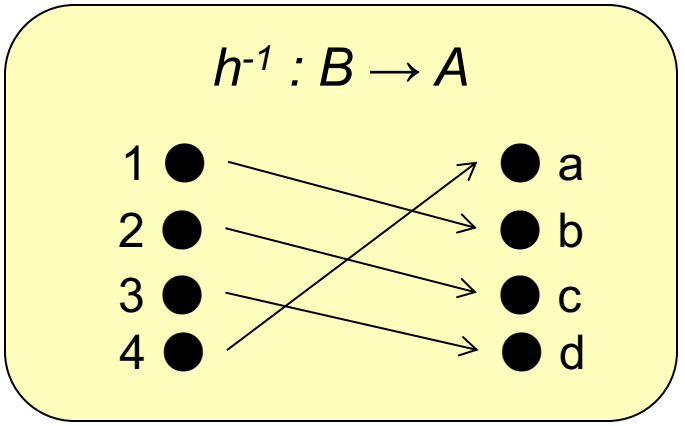
Do the following functions have inverses?

1. $f : \mathbf{R} \rightarrow \mathbf{R}$, $f(x) = x^2$

2. $g : \mathbf{Z} \rightarrow \mathbf{Z}$, $g(x) = x + 1$

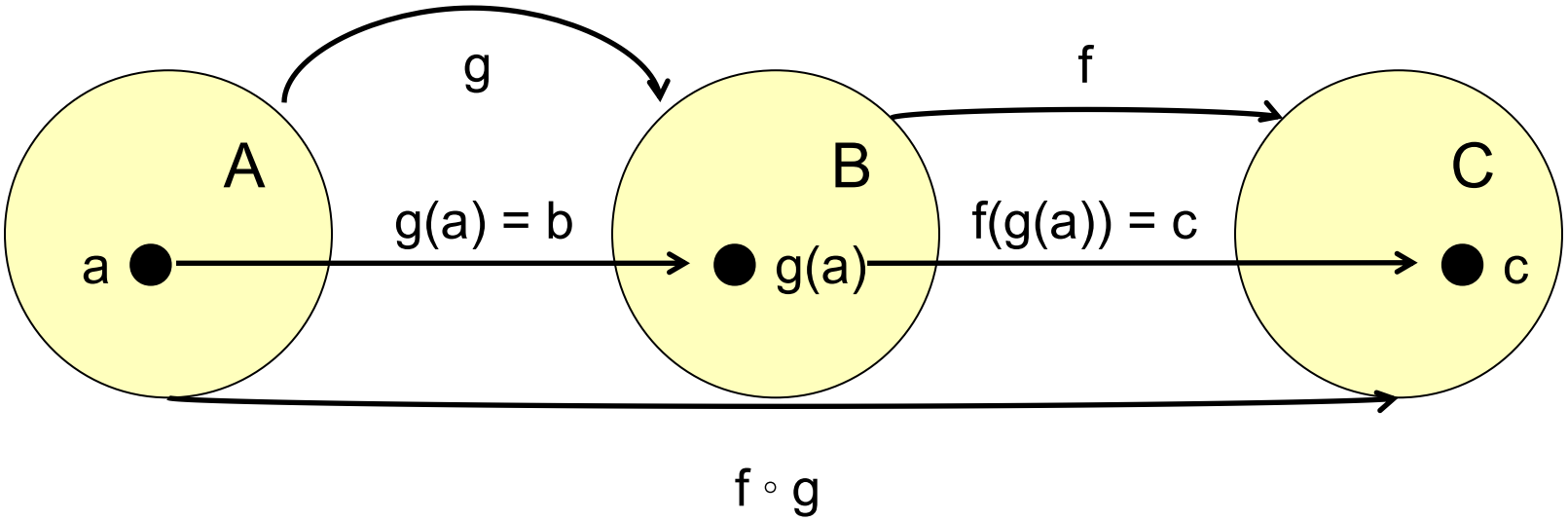
3. $h : A \rightarrow B$





Functions can be composed with one another

Definition: Given two functions $g : A \to B$ and $f : B \to C$, the composition of f and g, denoted $f \circ g$, is defined as $(f \circ g)(x) = f(g(x))$.



Note: For $f \circ g$ to exist, the codomain of g must be a subset of the domain of f.

Can the following functions be composed? If so, what is their composition?

Let $f : A \rightarrow A$ such that $f(a) = b$, $f(b) = c$, $f(c) = a$
$g : B \rightarrow A$ such that $g(1) = b$, $g(4) = a$

1. $(f \circ g)(x)$?
2. $(g \circ f)(x)$?

Let $f : \mathbf{Z} \rightarrow \mathbf{Z}$, $f(x) = 2x + 1$
$g : \mathbf{Z} \rightarrow \mathbf{Z}$, $g(x) = x^2$

1. $(f \circ g)(x)$?
2. $(g \circ f)(x)$?

Note: There is not a guarantee that $(f \circ g)(x) = (g \circ f)(x)$.

Important functions

Definition: The floor function maps a real number x to the largest integer y that is not greater than x. The floor of x is denoted $\lfloor x \rfloor$.

Definition: The ceiling function maps a real number x to the smallest integer y that is not less than x. The ceiling of x is denoted $\lceil x \rceil$.

Examples:

- $\lfloor 1.2 \rfloor = 1$
- $\lfloor 7.0 \rfloor = 7$
- $\lfloor -42.24 \rfloor = -43$

- $\lceil 1.2 \rceil = 2$
- $\lceil 7.0 \rceil = 7$
- $\lceil -42.24 \rceil = -42$

We actually use floor and ceiling quite a bit in computer science…

Example: A byte, which holds 8 bits, is typically the smallest amount of memory that can be allocated on most systems. How many bytes are needed to store 123 bits of data?

Answer: We need $\lceil 123/8 \rceil = \lceil 15.375 \rceil = 16$ bytes

Example: How many 1400 byte packets can be transmitted over a 14.4 kbps modem in one minute?

Answer: A 14.4 kbps modem can transmit 14,400*60 = 864,000 bits per minute. Therefore, we can transmit $\lfloor 864{,}000/(1400*8) \rfloor = \lfloor 77.1428571 \rfloor = 77$ packets.

In-class exercises

Problem 3: Find the domain and range of each of the following functions.

a. The function that determines the number of zeros in some bit string
b. The function that maps an English word to its two rightmost letters
c. The function that assigns to an integer the sum of its individual digits

Problem 4: Compute the following

a. $\lfloor 435.5 \rfloor$
b. $\lceil 89/90 \rceil$
c. $\lceil 5.5 + \lfloor 1.22 \rfloor \rceil$

Final thoughts

- Set identities are useful tools!
- We can prove set identities in a number of (equivalent) ways
- Sets are the basis of functions, which are used throughout computer science and mathematics
- Next time:
  - Summations (Section 2.4)