2110200
DISCRETE STRUCTURE

อ. ดร.อติวงศ์ สุชาโต

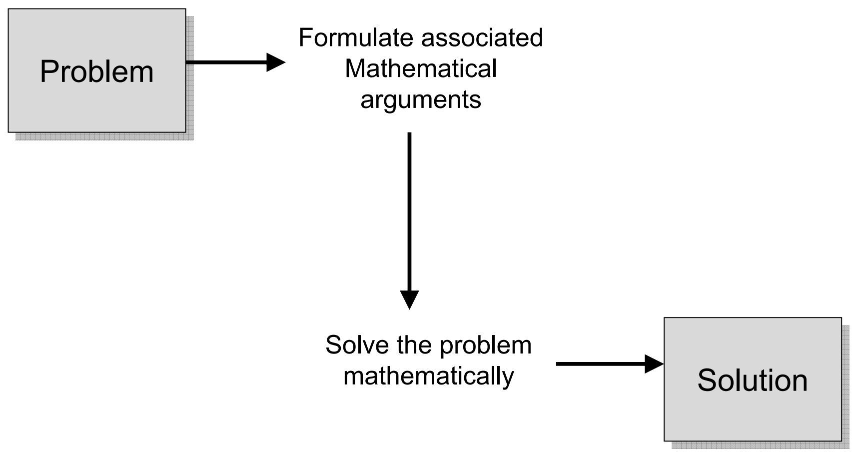
Course Outline

- 4 parts:
- Part1: Logic, Sets, Relations, Functions, and Mathematical Reasoning
- Part2: Graphs and Trees
- Part3: Counting, Recurrence Relations, and Generating Functions
- Part4: Number Theory

Grading

- In-class Quiz 1
- In-class Quiz 2
- In-class Quiz 3
- In-class Quiz 4
- Final Exam

Why ???



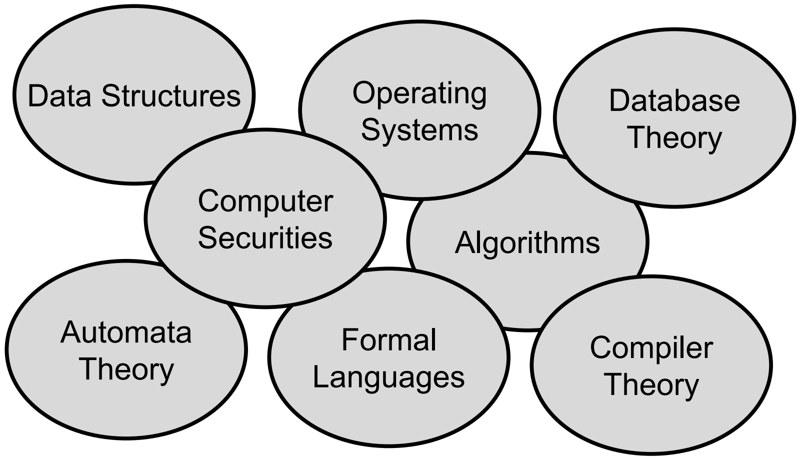
Goals of Discrete Math.

- **Mathematical Reasoning**
  - Read, comprehend, and construct mathematical arguments
- **Combinatorial Analysis**
  - Perform analysis to solve counting problems
- **Discrete Structure**
  - Able to work with discrete structures: sets, graphs, finite-state machines, etc.

Goals of Discrete Math.

- **Algorithmic Thinking**
  - Specify, verify, and analyze an algorithm
- **Applications and Modeling**
  - Apply the obtained problem-solving skills to model and solve problems in computer science and other areas, such as:
    - Business
    - Chemistry
    - Linguistics
    - Geology
    - etc

Gateway to . . .



Foundations of Discrete Math.

- **<u>Logic</u>**
  - Specify the meaning of Mathematical statements
  - Basis of all Mathematical reasoning
- **<u>Sets</u>**
  - Sets are collections of objects, which are used for building many important discrete structures.
- **<u>Functions</u>**
  - Used in the definition of some important structures
  - Represent complexity of an algorithm, and etc.

Readings

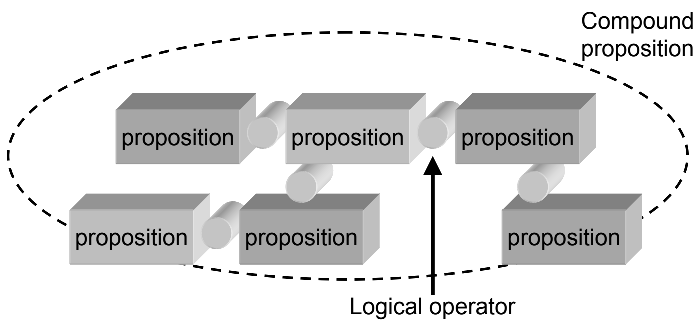
- Rosen: Section 1.1 to 1.4 and Section 1.6 to 1.8

Logic

- Rules of logic gives precise meaning to mathematical statements.

Proposition: Building Blocks of Logic

- Proposition =
  - Declarative sentence
  - Either *TRUE* or *FALSE* (not both)



Logical Operators

- Negation (NOT)
- Conjunction (AND)
- Disjunction (OR)
- Exclusive OR (XOR)
- Implication (IF..THEN)
- Biconditional (IF & ONLY IF)

Negation

- The negation of *p* has opposite truth value to *p*

p	$\neg p$
T	F
F	T

Conjunction

- The conjunction of *p* and *q*, is true when, and only when, both *p* and *q* are true.

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction

- The disjunction of *p* and *q*, is true when at least one of *p* or *q* is true.

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

Exclusive OR

- Exclusive or = OR but NOT both

$$p \oplus q = (p \vee q) \wedge \neg(p \wedge q)$$

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

Implication

- It is false when p is true and q is false, and true otherwise.

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Biconditional

- $p \leftrightarrow q$ is true when p and q have the same truth value.
  - Intuitively, $p \leftrightarrow q$ is $(p \rightarrow q) \wedge (q \rightarrow p)$

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

General Compound Proposition

- Example:

$$(p \wedge q) \vee \neg p$$

p	q	$p \wedge q$	$\neg p$	$(p \wedge q) \vee \neg p$
T	T	T	F	T
T	F	F	F	F
F	T	F	T	T
F	F	F	T	T

Contrapositive

- The *contrapositive* of an implication $p \rightarrow q$ is:

$$\neg q \rightarrow \neg p$$

- has the same truth values as $p \rightarrow q$

Converse and Inverse

- The *converse* of an implication $p \rightarrow q$ is:

$$q \rightarrow p$$

- The *inverse* of an implication $p \rightarrow q$ is:

$$\neg p \rightarrow \neg q$$

- DO NOT have the same truth values as $p \rightarrow q$

Precedence of Logical Operators

Operator	Precedence
$\neg$	1
$\wedge$	2
$\vee$	3
$\rightarrow$	4
$\leftrightarrow$	5

$$p \wedge \neg q \vee r \rightarrow p \leftrightarrow s$$

$$\downarrow$$

$$((p \wedge (\neg q) \vee r) \rightarrow p) \leftrightarrow s$$

Translating from Natural language

- Example (Rosen, p10):

  You cannot ride the rollercoaster if you are under 4 feet tall unless you are older than 16 years old.

q: You can ride the roller coaster
r: You are under 4 feet tall
s: You are older than 16 years old

$$(r \wedge \neg s) \rightarrow \neg q$$

q: You can ride the roller coaster
$\neg$ *r*: You are at least 4 feet tall
s: You are older than 16 years old

$$\neg(\neg r \vee s) \rightarrow \neg q$$

Consistency

- Translating natural language to logical expressions is essential to specifying system spec.
- Specifications are ***"consistent"*** when they do not conflict with one another. i.e.:

> There must be an assignment of truth values to every expression that make all the expression true.

Consistency

- Whenever the system is being upgraded, users cannot access the file system.
- If users can access the file system, they can save new files.
- If users cannot save new files, the system is not being upgraded.

Consistency

- Whenever the system is being upgraded, users cannot access the file system. $p \rightarrow \neg q$
- If users can access the file system, they can save new files. $q \rightarrow r$
- If users cannot save new files, the system is not being upgraded. $\neg r \rightarrow \neg p$

p	q	r	$p \rightarrow \neg q$	$q \rightarrow r$	$\neg r \rightarrow \neg p$
T	F	T	**T**	**T**	**T**

These spec. are consistent.

Tautology, Contradiction, & Contingency

- A compound proposition that is always *true* is called a ***"tautology"***.
- A compound proposition that is always *false* is called a ***"contradiction"***.
- If neither a tautology nor a contradiction, it is called a ***"contingency"***.

Logical Equivalences

The propositions p and q are called "**logical equivalent**" ($p \equiv q$) if $p \leftrightarrow q$ is a tautology

Showing Logically Equivalent propositions

1. Show that the truth values of these propositions are always the same.

→ Construct truth tables.

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Showing Logically Equivalent propositions

- Example (Rosen p22):
  Show that $p \rightarrow q \equiv \neg p \vee q$

p	q	$p \rightarrow q$	$\neg p$	$\neg p \vee q$
T	T	T	F	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Logically Equivalent

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Showing Logically Equivalent propositions

1. Show that the truth values of these propositions are always the same.
2. Use series of established equivalences.

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Logical Equivalences

- Distributive Laws
  $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
  $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
- De Morgan's Laws
  $\neg (p \wedge q) \equiv \neg p \vee \neg q$
  $\neg (p \vee q) \equiv \neg p \wedge \neg q$
- More can be found in Rosen p.24

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Showing Logically Equivalent propositions

- <u>Example</u> (Rosen p25):
  Show that $\neg (p \vee (\neg p \wedge q)) \equiv \neg p \wedge \neg q$

$$\begin{aligned}
\neg(p \vee (\neg p \wedge q)) &\equiv \neg p \wedge \neg(\neg p \wedge q) && \textit{De Morgan's} \\
&\equiv \neg p \wedge (\neg(\neg p) \vee \neg q) && \textit{De Morgan's} \\
&\equiv \neg p \wedge (p \vee \neg q) && \textit{Double negative} \\
&\equiv (\neg p \wedge p) \vee (\neg p \wedge \neg q) && \textit{Distributive} \\
&\equiv F \vee (\neg p \wedge \neg q) \\
&\equiv \neg p \wedge \neg q
\end{aligned}$$

Predicate Logic

Rockport, MA: June 2004

Predicate Logic

- In Propositional Logic, 'the atomic units' are propositions.
- E.g.:
  - *p*: John goes to school., *q*: Mary goes to school.
- In Predicate Logic, we look at each proposition as the combination of ***variables*** and ***predicates*** .
- E.g.:
  - X goes to school, where X can be John or Mary.

Predicate Logic

- The statement "*x* go to school" has two parts:
  Variable "*x*"
  The predicate "go to school"
- This statement can be denoted by *P(x)*, where *P* denotes the predicate "go to school".
- *P(x)* is said to be the value of the propositional function *P* at *x*.
- Once a value has been assigned to the variable *x*, the statement *P(x)* becomes a proposition and has a truth value.
- E.g: P(John) and P(Mary) have truth values.

Creating propositions from a propositional function

1. Assign values to all variables in a propositional function.
2. Use "Quantification"

Universal Quantifier

- $\forall x P(x)$ (read "for all x P(x)") denotes:

> *P(x)* is true for all values *x* in the universal of discourse.

- $\forall x P(x)$ is the same as:

$$P(x_1) \wedge P(x_2) \wedge \ldots \wedge P(x_n)$$

When all elements in the universe of discourse can be listed as $(x_1, x_2, \ldots, x_n)$

Universal Quantifier

- Example (Rosen p.31):
- What is the truth value of $\forall x P(x^2 \geq x)$, when the universe of discourse consists of:
  - all real numbers?
  - all integers?

  Since $x^2 \geq x$ only when $x \leq 0$ or $x \geq 1$, $\forall x P(x^2 \geq x)$ is false if the universe consists of all real numbers. However, it is true when the universe consists of only the integers.

Existential Quantifier

- $\exists x P(x)$ (read "for some x P(x)") denotes:

> There exists an element *x* in the universe of discourse that *P(x)* is true.

- $\exists x P(x)$ is the same as:

$$P(x_1) \vee P(x_2) \vee \ldots \vee P(x_n)$$

When all elements in the universe of discourse can be listed as $(x_1, x_2, \ldots, x_n)$

Existential Quantifier

- Example (Rosen p.32):
- What is the truth value of $\exists x P(x)$ where P(x) is the statement $x^2 > 10$, and the universe of discourse consists of the positive integers not exceeding 4?

  Since the elements in the universe can be listed as {1,2,3,4}, $\exists x P(x)$ is the same as $P(1) \vee P(2) \vee P(3) \vee P(4)$. There for $\exists x P(x)$ is true since *P(4)* is true.

Negations

$$\neg \forall x P(x) \equiv \exists x \neg P(x)$$

$$\neg \exists x P(x) \equiv \forall x \neg P(x)$$

Negation of

"Every 2^{nd} year students loves Discrete math." is

"There is a 2^{nd} year student who does not love Discrete math."

Negation of

"Some student in this class get 'A'." is

"None of the students in this class get 'A'."

Translating from Natural language

- Example (Rosen, p36):
  1. Some student in this class has visited Mexico.
  2. Every student in this class has visited Canada or Mexico.

M(x): X has visited Mexico
C(x): X has visited Canada
The universe of discourse consists of the students in this class.

1. $\exists x M(x)$
2. $\forall x (M(x) \vee C(x))$

Translating from Natural language

The universe of discourse consists of all people
M(x): X has visited Mexico.
C(x): X has visited Canada.
S(x): X is a student in this class.

"Some student in this class has visited Mexico" can be written "There is a person X having the properties that X is a student in this class AND X has visited Mexico."

$$\exists x (S(x) \wedge M(x))$$

CANNOT be written as $\exists x (S(x) \rightarrow M(x))$. Why??
What about the other statement?

Sets

Chiang Mai, February 2005

Sets

- A set is an unordered collection of objects.
- Objects in a set are called "members" or "elements" of that set.
- Two sets are equal ↔ they have the same elements

- Are {1,2,3} and {3,2,1} equal?
- Are {0,1,2} and {0,0,0,1,1,2} equal?

Set Builder Notation

- Stating the properties that all elements must have to be members.

O = {x | x is a prime number less than 100}
R = {x | x is a real number}
U = {x | x is any of the objects under consideration}

Subset

$$A \subseteq B \leftrightarrow \forall x\ (x \in A \rightarrow x \in B)$$

Proper Subset

$$A \subset B \leftrightarrow (A \subseteq B) \wedge (A \neq B)$$

For any set S, "$\varnothing \subseteq S$" and "$S \subseteq S$"

Proof: $\varnothing \subseteq S$ and $S \subseteq S$

- Show that $\forall x(x\in\varnothing \rightarrow x\in S)$
  - Since $x\in\varnothing$ is always false, then $x\in\varnothing \rightarrow x\in S$ is always true no matter what x is.
- Show that $\forall x(x\in S \rightarrow x\in S)$
  - Since $p \rightarrow p$ is a tautology the $x\in S \rightarrow x\in S$ is true no matter what.

Cardinality

- For a set *S*, if there are exactly *n* distinct elements in *S*, where *n* is a nonnegative interger, we say that *S* is a *finite set* and that *n* is the *cardinality* of *S* ($|S|=n$)
- A set is "infinite" if it is not finite.

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Power Set

- Given a set *S*, the power set of *S*, ***P(S)***, is the set of all subsets of *S*
- If *S* has *n* elements, then *P(S)* has 2^n elements.

- <u>Examples</u> (Rosen p.82):

S	P(S)
{0,1,2}	{∅,{0},{1},{2},{0,1},{0,2},{1,2},{0,1,2}}
∅	{∅}
{∅}	{∅,{∅}}

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Ordered n–tuple

- The *ordered n-tuple* $(a_1,a_2,..,a_n)$ is the ordered collection that has a_1 as its first element, a_2 as its second element,…, and an as its n^{th} element.

Two ordered n-tuples are equal ↔ each corresponding pair of their elements is equal

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Cartesian Products

$$A \times B = \{ (a,b) \mid a \in A \wedge b \in B\}$$

$$A_1 \times A_2 \times \ldots \times A_n = \{ (a_1,a_2,\ldots,a_n) \mid a_i \in A_i \text{ for } i=1,2,..,n\}$$

- Examples:
- What is the Cartesian product AxBxC, where A={0,1}, B={j,k}, C={x,y,z}?

AxBxC={(0,j,x),(0,j,y),(0,j,z),(0,k,x),(0,k,y),(0,k,z),
(1,j,x),(1,j,y),(1,j,z),(1,k,x),(1,k,y),(1,k,z)}

Using Set Notation with Quantifiers

- Specify the universe of discourse .
- E.g.:

$\forall x \in \mathbf{R}(x^2 \geq 0)$

means "for every real number $x^2 \geq 0$"

which is true.

Set Operations

- Union ($\cup$)
- Intersection ($\cap$)
- Difference ($-$)
- Complement ($'$)
- Symmetric difference ($\oplus$)

Symmetric Difference

- $A \oplus B$ is the set containing those elements in *either A or B* but *NOT in both A and B*.

Example:
A = {1,3,5}, B = {1,2,3}, $A \oplus B$ = {2,5}

Principle of Inclusion–Exclusion

$|A \cup B| = |A|+|B|-|A \cap B|$

More general (Chapter 6):

$|A_1 \cup A_2 \cup \ldots \cup A_n| =$
$\Sigma|A_i| - \Sigma|A_i \cap A_j| + \Sigma|A_i \cap A_j \cap A_k| - \ldots$
$+(-1)^{n+1}| A_1 \cap A_2 \cap \ldots \cap A_n|$

Set Identities

- Distributive Laws
  - $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
  - $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
- De Morgan's Laws
  - $(A \cup B)' = A' \cap B'$
  - $(A \cap B)' = A' \cup B'$
- More can be found in Rosen p.89

Showing that two sets have the same elements

1. Show that each set is a subset of the other.
2. Use set builder notation and logical equivalences.
3. Build membership tables.
4. Use set identities.

Proving Set Equality

Showing that each is a subset of the other

- Example (Rosen p.89): Prove that $(A \cap B)' = A' \cup B'$

1) Suppose $x \in (A \cap B)'$. So, $x \notin A \cap B$
Then, $\neg((x \in A) \wedge (x \in B))$ is true.
2) De Morgan's $\Rightarrow \neg(x \in A) \vee \neg(x \in B)$ is true.
Then, $x \in A' \vee x \in B'$
3) Definition of Union $\Rightarrow x \in A' \cup B'$
$x \in (A \cap B)' \rightarrow x \in A' \cup B'$
This shows $(A \cap B)' \subseteq A' \cup B'$

Proving Set Equality

Showing that each is a subset of the other

- Example (Rosen p.89): Continued

4) Suppose $x \in A' \cup B'$.
Definition of Union $\Rightarrow x \in A' \vee x \in B'$
$\neg(x \in A) \vee \neg(x \in B)$ is true.

5) Then, $\neg(x \in A \cap B)$ is true.
$\therefore x \notin A \cap B$. So, $x \in (A \cap B)'$.

6) $x \in A' \cup B' \rightarrow x \in (A \cap B)'$
This shows $A' \cup B' \subseteq (A \cap B)'$

Proving Set Equality

Showing that each is a subset of the other

- Example (Rosen p.89): Continued

3) $(A \cap B)' \subseteq A' \cup B'$
and $\rightarrow (A \cap B)' = A' \cup B'$

6) $A' \cup B' \subseteq (A \cap B)'$

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Proving Set Equality

Using set builder notation and logic equivalances

- Example (Rosen p.89): Prove that $(A \cap B)' = A' \cup B'$

$$
\begin{aligned}
(A \cap B)' &= \{ x \mid x \notin A \cap B \} \\
&= \{ x \mid \neg(x \in A \cap B) \} \\
&= \{ x \mid \neg((x \in A) \wedge (x \in B)) \} \\
&= \{ x \mid (x \notin A) \vee (x \notin B) \} \\
&= \{ x \mid (x \in A') \cup (x \in B') \} \\
&= \{ x \mid x \in A' \cup B' \} \\
&= A' \cup B'
\end{aligned}
$$

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Proving Set Equality

Using membership table

- Example Prove that $(A \cap B)' = A' \cup B'$

A	B	A′	B′	$A' \cup B'$	$(A \cap B)$	$(A \cap B)'$
0	0	1	1	1	0	1
0	1	1	0	1	0	1
1	0	0	1	1	0	1
1	1	0	0	0	1	0

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Proving Set Equality

Use set identities

- Example (Rosen p.91):
  Show that $(A \cup (B \cap C))' = (C' \cup B') \cap A'$

$$\begin{aligned}(A \cup (B \cap C))' &= A' \cap (B \cap C)' \\ &= A' \cap (B' \cup C') \\ &= (B' \cup C') \cap A' \\ &= (C' \cup B') \cap A'\end{aligned}$$

Generalized Union and Intersection

$$A_1 \cup A_2 \cup \ldots \cup A_n = \bigcup_{i=1}^{n} A_i$$

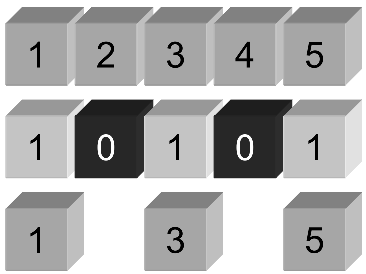
$$A_1 \cap A_2 \cap \ldots \cap A_n = \bigcap_{i=1}^{n} A_i$$

Computer Representation of Sets

Universal set = {1,2,3,4,5}

Representation for odd integers

Odd integer set = {1,3,5}



Computer Representation of Sets

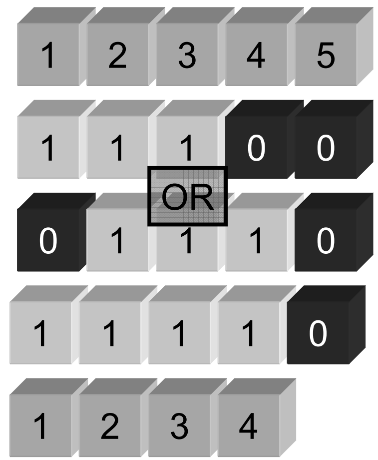
Universal set = {1,2,3,4,5}

Representation for A={1,2,3}

Representation for B={2,3,4}

$A \cup B$

={1,2,3,4}



Computer Representation of Sets

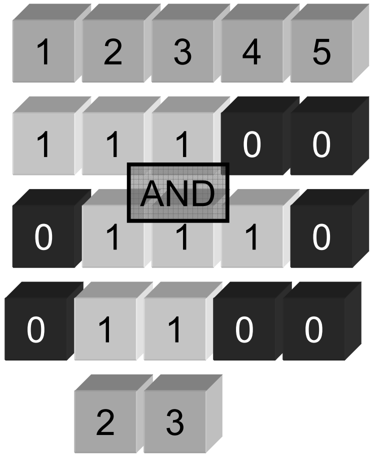
Universal set = {1,2,3,4,5}

Representation for A={1,2,3}

Representation for B={2,3,4}

$A \cap B$

={2,3}



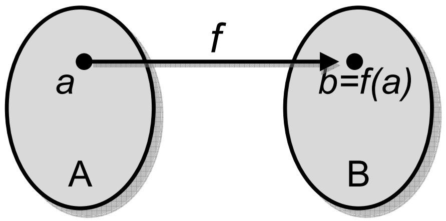
Functions

Washington D.C., April 2004

Functions

Definition:

- A function *f* from *A* to *B* is an assignment.
- assigns exactly one element of *B* to each of *A*



A: Domain
B: Codomain
b is *the image* of *a*.
a is *a pre-image* of *b*.
Range of f is the set of all images.

- Function cannot be "one-to-many".
- $\forall a \in A$, *f(a)* must be assigned to some *b.*

Adding and Multiplying Functions

- Two real-valued functions *with the same domain* can be added and multiplied.

f_1, f_2 are functions from A to $\mathbf{R}$
$\rightarrow f_1+f_2$ and f_1f_2 are also functions from A to $\mathbf{R}$.

$$(f_1+f_2)(x) = f_1(x)+f_2(x)$$
$$(f_1f_2)(x) = f_1(x)f_2(x)$$

Adding and Multiplying Functions

- Example (Rosen p.99):
- f_1, f_2 are functions from ***R*** to *R*. $f_1(x)=x^2$, $f_2(x)=x-x^2$. What are the functions f_1+f_2 and f_1f_2?

$(f1+f2)(x) = f1(x)+f2(x) = x2 + x - x2 = x$

$(f1f2)(x) = f1(x)f2(x) = x2\ (x - x2) = x3 – x4$

One-to-one Functions

A function f is *one-to-one* or *injective* $\leftrightarrow \forall x \forall y\ (f(x)=f(y) \rightarrow x=y)$

Examples (Rosen p.100)

Determine whether these functions are one-to-one.

$f_1(x) = x^2$ from the set of integers to the set of integers

Since $f(1) = f(-1) = 1$, $f_1(x)$ is not one-to-one.

$f_2(x) = x+1$

$x+1 \neq y+1$ when $x \neq y$, then $f_2(x)$ is one-to-one.

Conditions Guaranteeing One-to-one

- Strictly increasing function:

$\forall x\ \forall y\ ((x<y) \rightarrow (f(x)<f(y)))$

- Strictly decreasing function:

$\forall x\ \forall y\ ((x<y) \rightarrow (f(x)>f(y)))$

where the universe of discourse = domain of *f*

Strictly increasing function
or
Strictly decreasing function
$\rightarrow$ one-to-one

Onto Functions

A function f is *onto* or *surjective* $\leftrightarrow \forall y \exists x\ (f(x) = y)$

Examples (Rosen p.101)

Determine whether these functions are onto.

$f_1(x) = x^2$ from the set of integers to the set of integers

No, since there is no integer x that $f_1(x)=-1$

$f_2(x) = x+1$

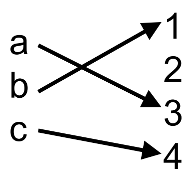
Yes, for every $f_2(x)=y$, there is an integer $x=y-1$

One-to-one Correspondence

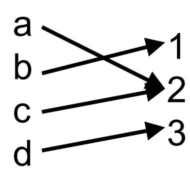
- *One-to-one* AND *Onto*
- Also called *"bijection"*

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

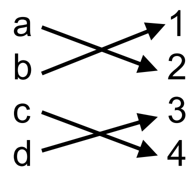
Examples



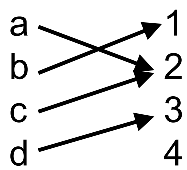
1-to-1, not onto



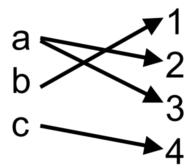
not 1-to-1,onto



1-to-1,onto



neither 1-to-1,nor onto



not a function

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Inverse Functions

- Let *f* be a *one-to-one correspondent* function from *A* to *B*.
- $f^{-1}(b)$ assigns to *b*, belonging to *B*, the unique element *a*, belonging to *A*, such that *f(a)=b*.

$$f^{-1}(b)=a \leftrightarrow f(a)=b$$

A function that is *NOT one-to-one correspondent* is *NOT invertible*.

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Composite Functions

- $(f \bullet g)(a) = f(\ g(a)\ )$
- $f \bullet g$ cannot be defined unless the range of *g* is a subset of the domain of *f*.
- If f is a one-to-one correspondent function from A to B

$$(f^{-1} \bullet f)(a) = a, \quad a \in A$$
$$(f \bullet f^{-1})(b) = b, \quad b \in B$$

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Some Important Functions

- Floor function $\lfloor \; \rfloor$

  $\lfloor x \rfloor$ = the largest integer $\leq x$

- Ceiling function $\lceil \; \rceil$

  $\lceil x \rceil$ = the smallest integer $\geq x$

$\lfloor 1/2 \rfloor$ = $\quad$ $\lfloor -1/2 \rfloor$ = $\quad$ $\lfloor 1 \rfloor$ =

$\lceil 1/2 \rceil$ = $\quad$ $\lceil -1/2 \rceil$ = $\quad$ $\lceil 1 \rceil$ =

Examples

- Example (Rosen p.106):
- Each byte is made up of 8 bits. How many bytes are required to encoded 100 bits of data?

$\lceil 100/8 \rceil = \lceil 12.5 \rceil$ = 13 bytes

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Useful Properties

$\lfloor x \rfloor = n \leftrightarrow n \leq x < n+1$

$\lceil x \rceil = n \leftrightarrow n-1 < x \leq n$

$\lfloor x \rfloor = n \leftrightarrow x-1 < n \leq x$

$\lceil x \rceil = n \leftrightarrow x \leq n < x+1$

$\lfloor -x \rfloor = -\lceil x \rceil$

$\lceil -x \rceil = -\lfloor x \rfloor$

$\lfloor x+n \rfloor = \lfloor x \rfloor + \mathrm{n}$

$\lceil x+n \rceil = \lceil x \rceil + n$

$x-1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x+1$

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Factorial Function

- $f(n) = n!$ is the product of the first n positive integers, so that

$$f(n) = 1 \cdot 2 \cdot \ldots \cdot (n-1) \cdot n$$
$$\text{and } f(0) = 0! = 1$$

Atiwong Suchato
Faculty of Engineering, Chulalongkorn University

Logic: Key Terms

- Proposition
- Truth value
- Negation
- Logical Operator
- Compound proposition
- Truth table
- Disjunction
- Conjunction
- Exclusive or
- Implication
- Inverse
- Converse
- Contrapositive
- Biconditional
- Bit operations
- Tautology
- Contradiction
- Contingency
- Consistency
- Logical equivalence
- Predicate
- Propositional function
- Universe of discourse
- Existential quantifier
- Universal quantifier

Sets: Key Terms

- Set
- Element
- Member
- Empty/Null set
- Universal set
- Venn diagram
- Set equality
- Subset
- Proper subset
- Finite set
- Infinite set
- Cardinality
- Power set
- Union
- Intersection
- Difference
- Complement
- Symmetric difference
- Membership table

Functions: Key Terms

- Function
- Domain
- Codomain
- Image
- Pre-image
- Range
- Onto / Surjection
- One-to-one / Injection
- One-to-one correspondence / bijection
- Inverse
- Composition
- Floor function
- Ceiling function
- Factorial