Data Structures and Algorithms

Set, relations, and functions theory

Lecture outline

- Set
- Relations
- Functions
- Logarithm, summations, recursion, proof, estimation

Sets

- A **set** is a collection of distinct objects.
- Example (let A denote a set):
  - $A = \{apple, pear, grape\}$
  - $A = \{1,2,3,4,5\}$
  - $A = \{1, b, c, d, e, f\}$
  - $A = \{(1,2), (3,4), (9,10)\}$
  - $A = \{< 1,2,3 >, < 3,4,5 >, < 6,7,8 >\}$
  - $A =$a collection of anything that is meaningful.

Members and Equality of Sets

- The objects that make up a set are called members or elements of the set.
- Two sets are equal iff they have the same members.
  - That is, a set is completely determined by its members.
  - Order does not matter in a set.
- **Cardinality**: a measure of the number of elements in a set.

Set notations

- The notation of {...} describes a set. Each member or element is separated by a comma.
  - E.g., $S = \{apple, pear, grape\}$
    - S is a set
    - The member of S are: apple, pear, grape.
- Set representations
  - Enumeration: $S = \{2, 4, 6, 8, 10\}$
  - Description: $S = \{x | x \ is \ an \ even \ integer \ number \ and \ 0 \leq x \leq 10\}$

The membership symbol ∈ and the empty set ∅

- The fact that x is a member of a set S can be expressed as $x \in S$
  - Reads: x is in S, or x is a member of S, or x belongs to S.
- An example, $S = \{1,2,3\}$, $1 \in S$, $2 \in S$, $3 \in S$
- The negation of ∈ is written as ∉ (is not in).
- The empty set is a set without any element
  - Denoted by {} or ∅
  - For any object x, $x \notin \emptyset$

Subsets

- A is a **subset** ($\subseteq$) of B, or B is a superset of A iff every member of A is a member of B.
  - $A \subseteq B$ iff forall x if $x \in a$, then x in B
- An example:
  - $(-2,\ \ 0,\ \ 6) \subseteq \{-3,\ -2,\ -1, 0, 1, 3, 6\}$
- **Negation**: A is not a subset of B or B is not a superset of A iff there is a member of A that is not a member of B
  - $A \not\subseteq B$ iff there exist x, $x \in A$, $x \notin B$

Proper subsets

- A is a proper subset ($\subset$) of B, or B is a proper superset of A iff A is a subset of B and A is not equal to B.
  - $A \subset B$ iff $A \subseteq B$ and $A \neq B$

- Examples:
  - $\{1, 2, 3\} \subset \{1, 2, 3, 4, 5\}$
  - $Z_+ \subset Z \subset Q \subset R$
  - If $S \neq \emptyset$ then $\emptyset \subset S$

Power sets

- The set of all subsets of a set is called the power set of the set
- The power set of S is denoted by P(S).
- Example:
  - $P(\emptyset)=\{\emptyset\}$
  - $P(\{1,2\})=\{\emptyset,\{1\},\{2\},\{1,2\}\}$
  - What is $P(\{1,2,3\})$?
- How many elements does the power set of S have? Assume S has n elements.

$\in$ and $\subseteq$ are different

- Examples:
  - $1 \in \{1\}$ is true
  - $1 \subseteq \{1\}$ is false
  - $1 \in \{1, 2, 3\}$ is true

- Which of the following statement is true?
  - $S \subseteq P(S)$
  - $S \in P(S)$

Mutual inclusion and set equality

- Set A and B have the same members iff they mutually include
  - $A \subseteq B$ and $B \subseteq A$
- That is, $A = B$ iff $A \subseteq B$ and $B \subseteq A$
- Mutual inclusion is very useful for proving the equality of sets
- To prove an equality, we prove two subset relationships.

An example: equality of sets

- Denote Z as the set of (all) integers
- Let A = {x $\in$ Z | x = 2m for m $\in$ Z }
- Let B = {x $\in$ Z | x = 2n-2 for n $\in$ Z }
- To show A $\subseteq$ B, note that

  2m=2(m+1) -2 = 2n-2
- To show that B $\subseteq$ A, note that

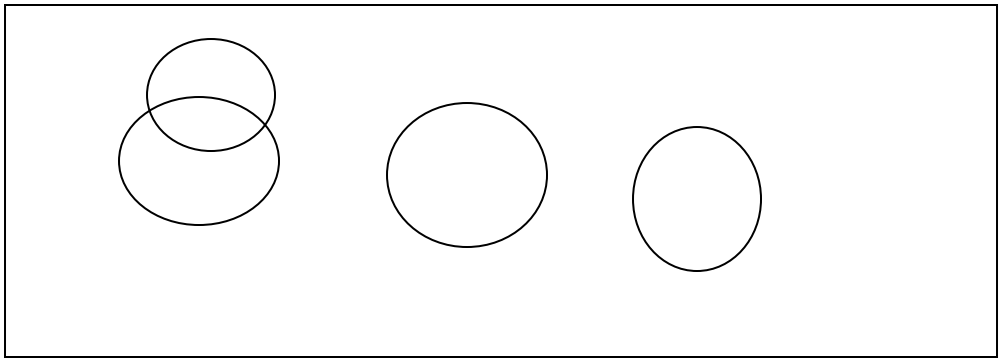
  2n-2=2(n-1)=2m
- That is, A = B. (A, B both denote the set of all even integers.)

Universal sets

- Depending on the context of discussion
  - Define a set of U such that all sets of interest are subsets of U.
  - The set U is known as a universal set.
- Examples:
  - When dealing with integers, U may be Z.
  - When dealing with plane geometry, U may be the set of points in the plane.

Venn diagram

- Venn diagram is used to visualize relationships of some sets.
- Each subset (of U, the rectangle) is represented by a circle inside the rectangle.



Set operations

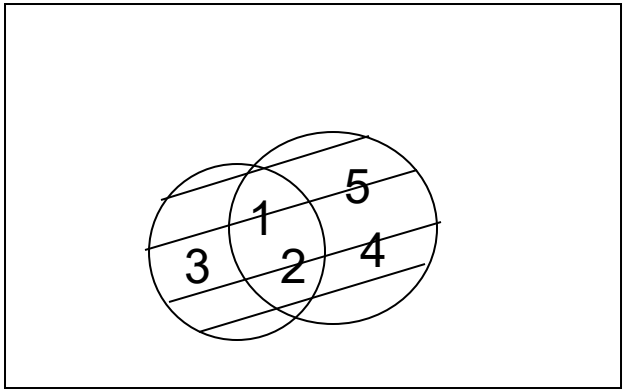
- Let A, B be subsets of some universal set U. The following set operations create new sets from A and B.
- Union:
  - $A \cup B = \{x \in U \mid x \in A \text{ or } x \in B\}$
- Intersection:
  - $A \cap B = \{x \in U \mid x \in A \text{ and } x \in B\}$
- Difference:
  - $A - B = A \setminus B = \{x \in U \mid x \in A \text{ and } x \notin B\}$
- Complement
  - $A' = U - A = \{x \in U \mid x \notin A\}$

Set union

- An example

  $\{1, 2, 3\} \cup \{1, 2, 4, 5\} = \{1, 2, 3, 4, 5\}$

  The venn diagram

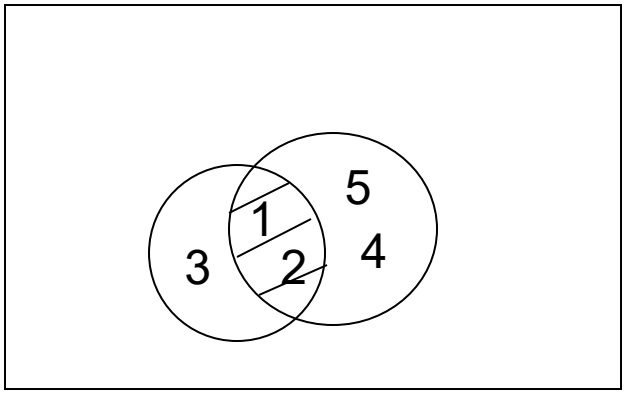


Set intersection

- An example

  $\{1, 2, 3\} \cap \{1, 2, 4, 5\} = \{1, 2\}$

  The venn diagram

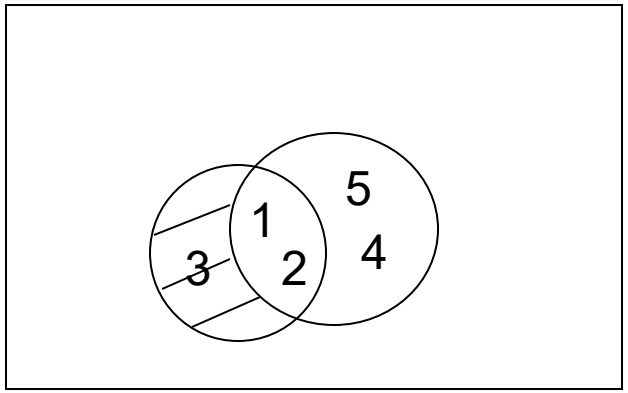


Set difference

- An example

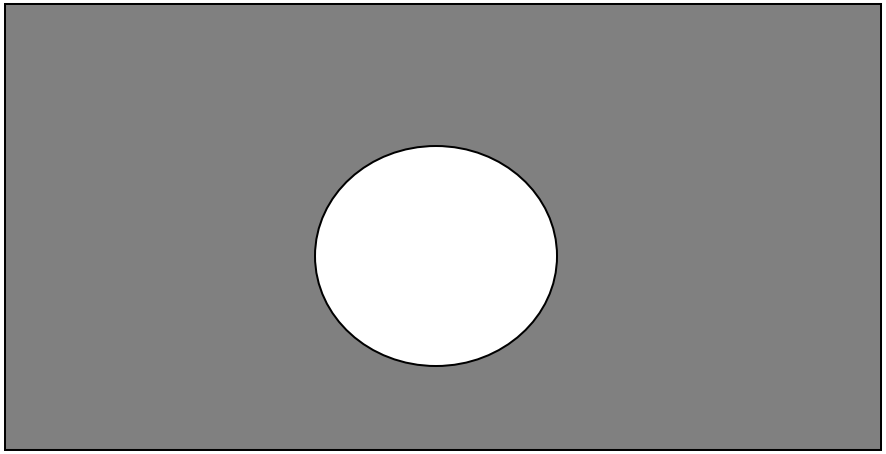
  {1, 2, 3} - {1, 2, 4, 5} = {3}

  The venn diagram



Set complement

- The venn diagram



Basic set identities (equalities)

- Commutative laws

  $A \cup B = B \cup A$

  $A \cap B = B \cap A$

- Associative laws

  $(A \cup B) \cup C = A \cup (B \cup C)$

  $(A \cap B) \cap C = A \cap (B \cap C)$

- Distributive laws

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

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

Basic set identities (cont'd …)

- Identity laws

  $\varnothing \cup A = A \cup \varnothing = A$

  $A \cap U = U \cap A = A$

- Double complement law

  $(A')' = A$

- Idempotent laws

  $A \cup A = A$

  $A \cap A = A$

- De Morgan's laws

  $(A \cup B)' = A' \cap B'$

  $(A \cap B)' = A' \cup B'$

Basic set identities (cont'd …)

- Absorption laws
  - $A \cup (A \cap B) = A$
  - $A \cap (A \cup B) = A$
- Complement law
  - $(U)' = \varnothing$
  - $\varnothing' = U$
- Set difference law
  - $A - B = A \cap B'$
- Universal bound law
  - $A \cup U = U$
  - $A \cap \varnothing = \varnothing$

Proof methods

- There are many ways to prove set identities
  - Applying existing identities
  - Using mutual inclusion (**MI**)
- Prove $(A \cap B) \cap C = A \cap (B \cap C)$ using MI
  - First show: $(A \cap B) \cap C \subseteq A \cap (B \cap C)$
  - Let $x \in (A \cap B)$ and $x \in C$
    - $(x \in A$ and $x \in B)$ and $x \in C$
    - $x \in A$ and $x \in (B \cap C)$
    - $x \in A \cap (B \cap C)$
  - Then show that $A \cap (B \cap C) \subseteq (A \cap B) \cap C$

More proof examples

- Let B = {x | x is a multiple of 4}

  A = {x | x is a multiple of 8}

  Then we have A $\subseteq$ B
- Proof: let x $\in$ A. We must show that x is a multiple of 4. We can write x = 8m for some integer m. We have
  - x = 8m = 2*4m = 4 k, where k = 2m,
  - so k is a integer.
  - Thus, x is a multiple of 4, and therefor x $\in$ B.

More proof examples

- Prove $\{x \mid x \in Z \text{ and } x \geq 0 \text{ and } x^2 < 15\}$
  $= \{x \mid x \in Z \text{ and } x \geq 0 \text{ and } 2x < 7\}$
- Proof:
  - Let $A = \{x \mid x \in Z \text{ and } x \geq 0 \text{ and } x^2 < 15\}$
    $B = \{x \mid x \in Z \text{ and } x \geq 0 \text{ and } 2x < 7\}$
  - Let $x \in A$. x can only be 0, 1, 2, 3
    2x for 0, 1, 2, 3 are all less then 7.
    Thus, $A \subseteq B$.
  - Likewise, we can also show that $B \subseteq A$

Algebraic proof examples

Prove:

$[A \cup (B \cap C)] \cap ([A' \cup (B \cap C)] \cap (B \cap C)') = \varnothing$

Proof:

$[A \cup (B \cap C)] \cap ([A' \cup (B \cap C)] \cap (B \cap C)')$

$= ([A \cup (B \cap C)] \cap [A' \cup (B \cap C)]) \cap (B \cap C)'$ (associative)

$= ([(B \cap C) \cup A] \cap [(B \cap C) \cup A']) \cap (B \cap C)'$ (commutative)

$= [(B \cap C) \cup (A \cap A')] \cap (B \cap C)'$ (distributive)

$= [(B \cap C) \cup \varnothing] \cap (B \cap C)'$ (complement)

$= (B \cap C) \cap (B \cap C)'$ (Identity)

$= \varnothing$ (identity)

Algebraic proof examples

- Prove:

  $(A \cup B) - C = (A - C) \cup (B - C)$

- Proof:

  $(A \cup B) - C = (A \cup B) \cap C'$ (difference)

  $= C' \cap (A \cup B)$ (commutative)

  $= (C' \cap A) \cup (C' \cap B)$ (distributive)

  $= (A \cap C') \cup (B \cap C')$ (commutative)

  $= (A - C) \cup (B - C)$ (difference)

Disproving an alleged Set property

- Is the following true?

  $(A - B) \cup (B - C) = A - C$

- Solution: Draw a Venn diagram and construct some sets to confirm the answer
- **Counterexample**: A = {1, 2, 4, 5}, B = {2, 3, 5, 6}, and C = {4, 5, 6, 7}

  $A - B = \{1, 4\}$, $B - C = \{2, 3\}$, $A - C = \{1, 2\}$

  $(A - B) \cup (B - C) = \{1, 2, 3, 4\}$

Pigeonhole principle

- If more than *k* pigeons fly into *k* pigeonholes, then at least one hole will have more than one pigeon.
- **Pigeonhole principle**: if more than *k* items are placed into *k* bins, then at least one bin contains more than one item.
- Simple, and obvious!!
- To apply it, may not be easy sometimes. Need to be clever in identifying pigeons and pigeonholes.

Example

- How many people must be in a room to guarantee that two people have last names that begin with the same initial?
- How many times must a single die be rolled in order to guarantee getting the same value twice?

Another example

- Prove that if four numbers are chosen from the set {1, 2, 3, 4, 5, 6}, at least one pair must add up to 7.

- Proof: There are 3 pairs of numbers from the set that add up to 7, i.e.,

(1, 6), (2, 5), (3, 4)

Apply pigeonhole principle: bins are the pairs, and the numbers are the items.

Summary

- Sets are extremely important for Computer Science.
- A set is simply an unordered list of objects.
- Set operations: union, intersection, difference.
- To prove set equalities
  - Applying existing identities
  - Using mutual inclusion
- Pigeonhole principle

Relations

Ordered n-tuples

- An ordered n-tuple is an ordered sequence of n objects, denoted as

  $(x_1, x_2, \ldots, x_n)$

  - 1st coordinate (or component) is x_1
  - ...
  - n-th coordinate (or component) is x_n

- An ordered pair: An ordered 2-tuple
  - (x, y)
- An ordered triple: An ordered 3-tuple
  - (x, y, z)

Equality of tuples vs sets

- Two tuples are equal iff they are equal coodinate-wise
  - $(x_1, x_2, \ldots, x_n) = (y_1, y_2, \ldots, y_n)$ iff

    $x_1 = y_1, x_2 = y_2, \ldots, x_n = y_n$

- $(2, 1) \neq (1, 2)$, but $\{2, 1\} = \{1, 2\}$
- $(1, 2, 1) \neq (2, 1)$, but $\{1, 2, 1\} = \{2, 1\}$
- (1, 2-2, a) = (1, 0, a)
- $(1, 2, 3) \neq (1, 2, 4)$ and $\{1, 2, 3\} \neq \{1, 2, 4\}$

Cartesian products

- Let A_1, A_2, …A_n be sets
- The cartesian products of A_1, A_2, …A_n is
  - $A_1 \times A_2 \times \ldots \times A_n$
    $= \{ (x_1, x_2, \ldots, x_n) \mid x_1 \in A_1$ and $x_2 \in A_2$ and … and $x_n \in A_n)$
- Examples: A = {x, y}, B = {1, 2, 3}, C = {a, b}
- AxB={(x, 1), (x, 2), (x, 3), (y, 1), (y, 2), (y, 3)}
- AxBxC = {(x, 1, a), (x, 1, b), …, (y, 3, a), (y, 3, b)}
- Ax(BxC) = {(x, (1, a)), (x, (1, b)), …, (y, (3, a)), (y, (3, b))}

Relations

- A **relation** is a set of ordered pairs
  - Let **x R y** mean x is R-related to y
  - Let A be a set containing all possible x
  - Let B be a set containing all possible y

  Relation R can be treated as a set of ordered pairs

  $R = \{(x, y) \in AxB \mid x\ R\ y\}$
- Example: We have the relation "is-capital-of" between cities and countries:

  Is-capital-of = {(Beijing, CHN), (WashingtonDC, US), …}

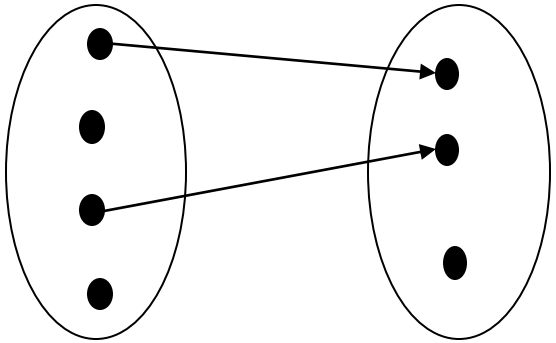
Relations are sets

- $R \subseteq AxB$ as a relation from A to B
- R is a relation from A to B iff $R \subseteq AxB$
  - Furthermore, x R y iff $(x, y) \in R$.
- If the relation R only involves two sets, we say it is a **binary relation**.
- We can also have an n-ary relation, which involves n sets.

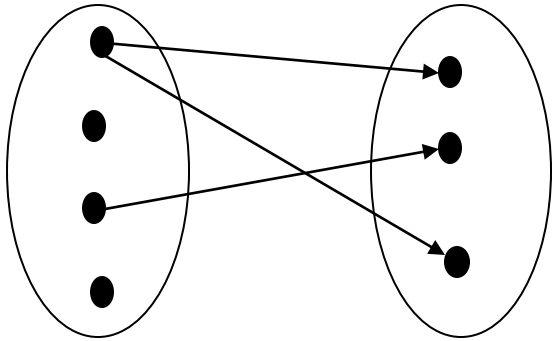
Various kinds of binary relations

- One-to-one relation: each first component and each second component appear only once in the relation.
- One-to-many relation: if some first component s_1 appear more than once.
- Many-to-one relation: if some second component s_2 is paired with more than one first component.
- Many-to-many relation: if at least one s_1 is paired with more than one second component and at least one s_2 is paired with more than one first component.

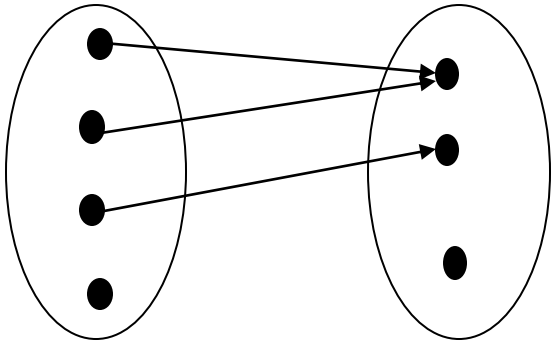
Visualizing the relations



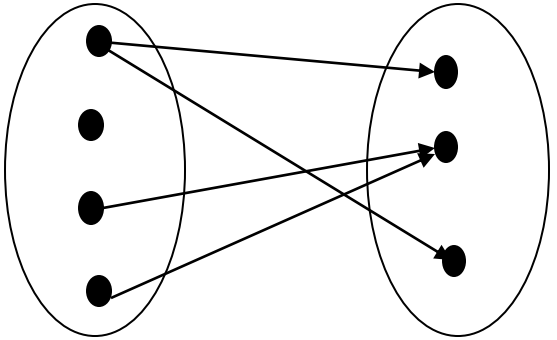
One-to-one



One-to-many



Many-to-one



Many-to-many

Binary relation on a set

- Given a set A, a binary relation R on A is a subset of AxA ($R \subseteq AxA$).

- An example:
  - A = {1, 2}. Then AxA={(1,1), (1,2), (2,1), (2,2)}. Let R on A be given by x R y $\leftrightarrow$ x+y is odd.
  - then, (1, 2) $\in$ R, and (2, 1) $\in$ R

Properties of Relations: Reflexive

- Let R be a binary relation on a set A.
  - R is reflexive: iff for all $x \in A$, $(x, x) \in R$.
- Reflexive means that every member is related to itself.
- Example: Let A = {2, 4, a, b}
  - R = {(2, 2), (4, 4), (a, a), (b, b)}
  - S = {(2, b), (2, 2), (4, 4), (a, a), (2, a), (b, b)}
- R, S are reflexive relations on A.
- Another example: the relation $\leq$ is reflexive on the set Z_+.

Properties of Relations: Symmetric

- A relation R on a set A is symmetric iff for all x, y $\in$ A, if (x, y) $\in$ R then (y, x) $\in$ R .
- Example: A = {1, 2, b}
  - R = {(1, 1), (b, b)}
  - S = {(1, 2)}
  - T = {(2, b), (b, 2), (1, 1)}
- R, T are symmetric relations on A.
- S is not a symmetric relation on A.
- The relation $\leq$ is reflexive on the set Z_+, but not symmetric. E.g., $3 \leq 4$ is in, but not $4 \leq 3$

Properties of Relations: Anti-symmetric

- A relation R on a set A is anti-symmetric iff for all x, y $\in$ A. if (x, y) $\in$ R and (y, x) $\in$ R then x = y.
- Example: A = {1, 2, b}
  - R = {(1, 1), (b, b)}
  - S = {(1, 2)}
  - T = {(2, b), (b, 2), (1, 1)}
- R, S are anti-symmetric relations on A.
- T is not an anti-symmetric relation on A.
- The relation $\leq$ is reflexive on the set Z_+, but not symmetric. It is anti-symmetric.

Properties of Relations: Transitive

- A relation R on a set A is transitive iff for all x, y, z $\in$ A, if (x, y) $\in$ R and (y, z) $\in$ R, then (x, z) $\in$ R.
- Example: A = {1, 2, b}
  - R = {(1, 1), (b, b)}
  - S = {(1, 2), (2, b), (1, b)}
  - T = {(2, b), (b, 2), (1, 1)}
- R, S are transitive relations on A.
- T is not a transitive relation on A.
- The relation $\leq$ is reflexive on the set Z_+, but not symmetric. It is also anti-symmetric, and transitive (why?).

Transitive closure

- Let R be a relation on A
- The smallest transitive relation on A that includes R is called the transitive closure of R.
- Example: A = {1, 2, b}
  - R = {(1, 1), (b, b)}
  - S = {(1, 2), (2, b), (1, b)}
  - T = {(2, b), (b, 2), (1, 1)}
- The transitive closures of R and S are themselves
- The transitive closure of T is T $\cup$ {(2, 2), (b, b)}

Equivalence relations

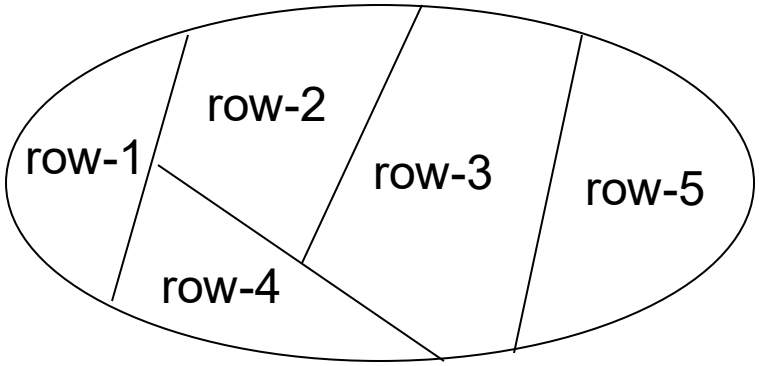
- A relation on a set A is an equivalence relation if it is
  - Reflexive.
  - Symmetric
  - Transitive.
- Examples of equivalence relations
  - On any set S, x R y $\leftrightarrow$ x = y
  - On integers ≥ 0, x R y $\leftrightarrow$ x+y is even
  - On the set of lines in the plane, x R y $\leftrightarrow$ x is parallel to y.
  - On {0, 1}, x R y $\leftrightarrow$ $x = y^2$
  - On {1, 2, 3}, R = {(1, 1), (2, 2), (3, 3), (1, 2), (2, 1)}.

Congruence relations are equivalence relations

- We say x is congruent modulo m to y
  - That is, x C y iff m divides x-y, or x-y is an integral multiple of m.
  - We also write $x \equiv y \pmod{m}$ iff x is congruent to y modulo m.
- Congruence modulo m is an equivalent relation on the set Z.
  - Reflexive: m divides x-x = 0
  - Symmetry: if m divides x-y, then m divides y-x
  - Transitive: if m divides x-y and y-z,
    then m divides (x-y)+(y-z) = x-z

An important feature

- Let us look at the equivalence relation:
  - S = {x | x is a student in our class}
  - x R y $\leftrightarrow$ “x sits in the same row as y”
- We group all students that are related to one another. We can see this figure:



- We have partitioned S into subsets in such a way that everyone in the class belongs to one and only one subset.

Partition of a set

- A partition of a set S is a collection of nonempty disjoint subsets (S_1, S_2, .., S_n) of S whose union equals S.
  - $S_1 \cup S_2 \cup \ldots \cup S_n = S$
  - If $i \neq j$ then $S_i \cap S_j = \varnothing$ ($S_i \cap S_j$ are disjoint)
- Examples, let A = {1, 2, 3, 4}
  - {{1}, {2}, {3}, {4}} a partition of A
  - {{1, 2}, {3, 4}} a partition of A
  - {{1, 2, 3}, {4}} a partition of A
  - {{}, {1, 2, 3}, {4}} not a partition of A
  - {{1, 2}, {3, 4}, {1, 4}} not a partition of A

Equivalent classes

- Let R be an equivalence relation on a set A.
  - Let $x \in A$
- The equivalent class of x with respect to R is:
  - $R[x] = \{y \in A \mid (x, y) \in R\}$
  - If R is understood, we write [x] instead of R[x].
- Intuitively, [x] is the set of all elements of A to which x is related.

Theorems on equivalent relations and partitions

Theorem 1: An equivalence relation R on a set A determines a partition of A.

- i.e., the distinctive equivalence classes of R form a partition of A.

Theorem 2: a partition of a set A determines an equivalence relation on A.

- i.e., there is an equivalence relation R on A such that the set of equivalence classes with respect to R is the partition.

An equivalent relations induces a partition

- Let A = {0, 1, 2, 3, 4, 5}
- Let R be the congruence modulo 3 relation on A
- The set of equivalence classes is:
  - {[0], [1], [2], [3], [4], [5]} =
    {{0, 3}, {1, 4}, {2, 5}, {3, 0}, {4, 1}, {5, 2}} =
    {{0, 3}, {1, 4}, {2, 5}}
- Clearly, {{0, 3}, {1, 4}, {2,5}} is a partition of A.

An partition induces an equivalent relation

- Let A = {0, 1, 2, 3, 4, 5}
- Let a partition P = {{0, 5}, {1, 2, 3}, {4}}
- Let R =

  $\{\{0, 5\} \times \{0, 5\} \cup \{1, 2, 3\} \times \{1, 2, 3\} \cup \{4\} \times \{4\}\}$

  $= \{(0, 0), (0, 5), (5, 0), (5, 5), (1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3), (4, 4)\}$
- It is easy to verify that R is an equivalent relation.

Partial order

- A binary relation R on a set S is a partial order on S iff R is
  - Reflexive
  - Anti-symmetric
  - Transitive
- We usually use $\leq$ to indicate a partial order.
- If R is a partial order on S, then the ordered pair (S, R) is called a **partially ordered set** (also known as **poset**).
- We denote an arbitrary partially ordered set by $(S, \leq)$.

Examples

- On a set of integers, x R y $\leftrightarrow$ x $\leq$ y is a partial order ($\leq$ is a partial order).
- for integers, a, b, c.
  - a $\leq$ a (reflexive)
  - a $\leq$ b, and b $\leq$ a implies a = b (anti-symmetric)
  - a $\leq$ b and b $\leq$ c implies a $\leq$ c (transitive)
- Other partial order examples:
  - On the power set P of a set, A R B $\leftrightarrow$ A $\subseteq$ B
  - On Z_+, x R y $\leftrightarrow$ x divides y.
  - On {0, 1}, x R y $\leftrightarrow$ $x = y^2$

Some terminology of partially ordered sets

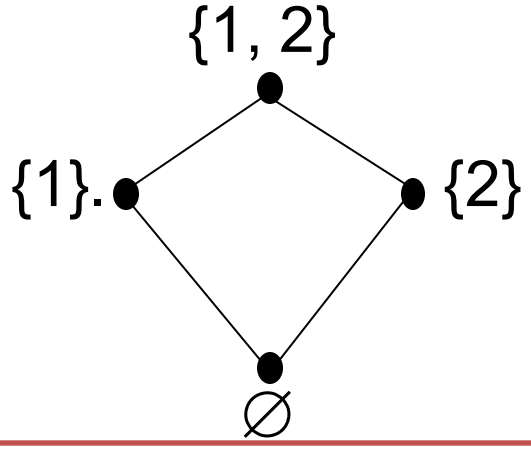
- Let $(S, \leq)$ be a partially ordered set
- If $x \leq y$, then either $x = y$ or $x \neq y$.
- If $x \leq y$, but $x \neq y$, we write $x < y$ and say that x is a **predecessor** of y, or y is a **successor** of x.
- A given y may have many predecessors, but if $x < y$ and there is no z with $x < z < y$, then x is an immediate predecessor of y.

Visualizing partial order: Hasse diagram

- Let S be a finite set.
- Each of the element of S is represented as a dot (called a **node**, or **vertex**).
- If x is an immediate predecessor of y, then the node for y is placed above node x, and the two nodes are connected by a straight-line segment.
- The Hasse diagram of a partially ordered set conveys all the information about the partial order.
- We can reconstruct the partial order just by looking at the diagram

An example Hasse diagram

- $\subseteq$ on the power set P({1, 2}):
  - Poset: (P({1, 2}), $\subseteq$)
- P({1, 2}) = {$\varnothing$, {1}, {2}, {1, 2}}
- $\subseteq$ consists of the following ordered pairs
  ($\varnothing$, $\varnothing$), ({1}, {1}), ({2}, {2}), ({1, 2}, {1, 2}),
  ($\varnothing$, {1}), ($\varnothing$, {2}), ($\varnothing$, {1, 2}), ({1}, {1, 2}),
  ({2}, {1, 2})



Total orders

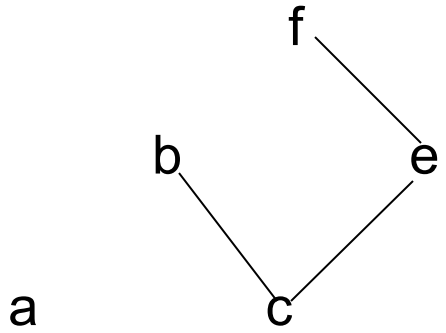
- A partial order on a set is a **total order** (also called **linear order**) iff any two members of the set are related.
- The relation $\leq$ on the set of integers is a total order.
- The Hasse diagram for a total order is on the right

Least element and minimal element

- Let $(S, \leq)$ be a poset. If there is a $y \in S$ with $y \leq x$ for all $x \in S$, then y is a **least element** of the poset. If it exists, is unique.
- An element $y \in S$ is **minimal** if there is no $x \in S$ with $x < y$.
- In the Hasse diagram, a least element is below all orders.
- A minimal element has no element below it.
- Likewise we can define **greatest element** and **maximal element**

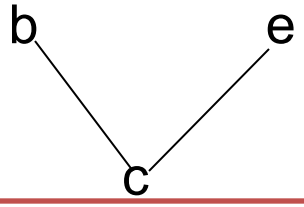
Examples: Hasse diagram

- Consider the poset:

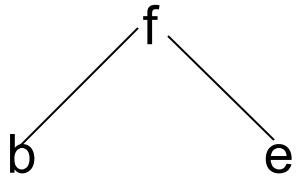


- The maximal elements are a, b, f
- The minimal elements are a, c.

A least element but no greatest element



A greatest element but no least element



Summary

- A binary relation on a set S is a subset of SxS.
- Binary relations can have properties of reflexivity, symmetry, anti-symmetry, and transitivity.
- Equivalence relations. A equivalence relation on a set S defines a partition of S.
- Partial orders. A partial ordered set can be represented graphically.

Functions

High school functions

- Functions are usually given by formulas
  - $f(x) = \sin(x)$
  - $f(x) = e^x$
  - $f(x) = x^3$
  - $f(x) = \log x$
- A function is a computation rule that changes one value to another value
- Effectively, a function associates, or relates, one value to another value.

"general" functions

- We can think of a function as relating one object to another (need not be numbers).
- A relation f from A to B is a **function** from A to B iff
  - for every $x \in A$, there exists a unique $y \in B$ such that x f y, or equivalently $(x, y) \in f$
- Functions are also known as transformations, maps, and mappings.

Notational convention

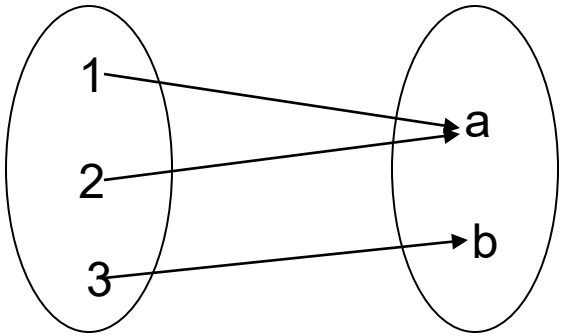
- Sometimes functions are given by stating the rule of transformation, for example,
  - $f(x) = x + 1$
- This should be taken to mean

  $f = \{(x, f(x)) \in AxB \mid x \in A\}$

  where A and B are some understood sets.

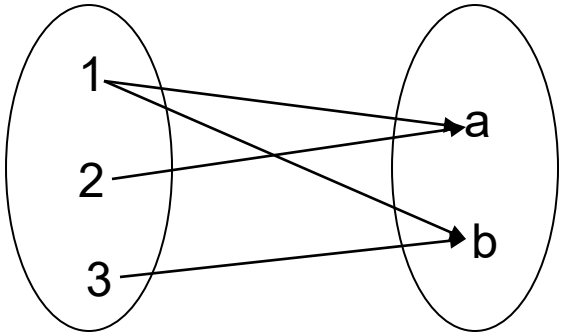
Examples

- Let A = {1, 2, 3} and
  B = {a, b}



- R = {(1, a), (2, a), (3, b)} is **a function** from A to B

- R = {(1, a), (1, b), (2, a), (3, b)} is **not a function** from A to B



Notations and concepts

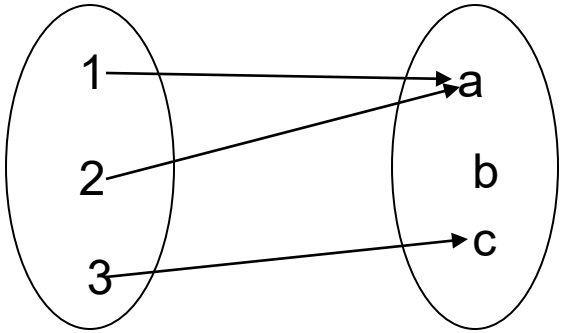
- Let A and B be sets, f is a function from A to B. We denote the function by:

  f: A $\rightarrow$ B

- *A* is the **domain**, and *B* is the **codomain** of the function.
- If (*a, b*) $\in$ *f*, then *b* is denoted by *f*(*a*); *b* is the **image** of *a* under *f*, *a* is a **preimage** of *b* under *f*.
- The range of *f* is the set of images of *f*.
  - The range of *f* is the set *f(A).*

An example

- Let the function f be



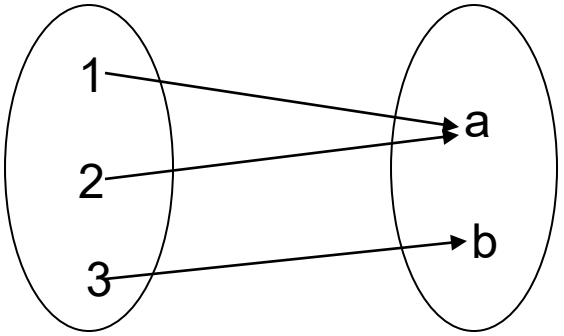
- Domain is {1, 2, 3}
- Codomain is {a, b, c}
- Range is {a, c}

Equality of functions

- Let f: A $\rightarrow$ B and g: C $\rightarrow$ D.
- We denote function f = function g
  - iff set f = set g
- Note that this force A = C, but not B = D
  - Some require B = D as well.

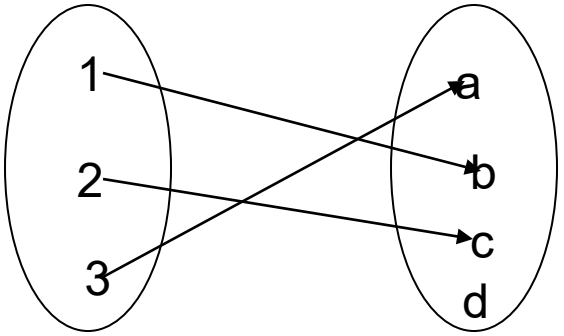
Properties of functions: **onto**

- Let f: A $\rightarrow$ B
  - The function f is an **onto** or **surjective** function iff the range of f equals to the codomain of f.
  - Or for any y $\in$ B, there exists some x $\in$ A, such that f(x) = y.
- The function on the right is onto.
- f: Z $\rightarrow$ Z with $f(x) = x^2$ is not onto



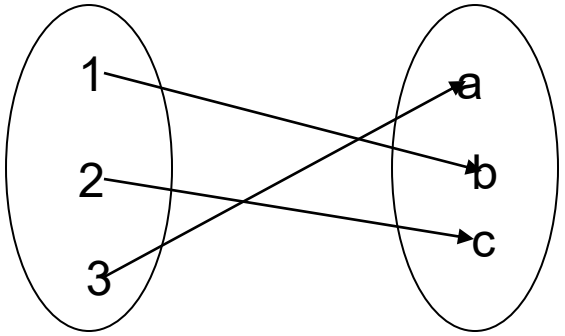
One-to-one functions

- A function f: A $\rightarrow$ B is **one-to-one**, or **injective** if no member of B is the image under f of two distinct elements of A.
- Let A = {1, 2, 3}
- Let B = {a, b, c, d}
- Let f = {(1, b), (2, c), (3, a)}
- The function f is one-to-one
- f: Z $\rightarrow$ Z with $f(x) = x^2$ is not one-to-one because f(2) = f(-2) = 4.



Bijections (one-to-one correspondences)

- A function f: A $\rightarrow$ B is **bijective** if f is both one-to-one and onto.
- Let A = {1, 2, 3}
- Let B = {a, b, c}
- Let f = {(1, b), (2, c), (3, a)}
- The function f is one-to-one
- f: Z $\rightarrow$ Z with f(x) = x^2 is not bijective because it is not one-to-one.



Composition of functions

- Let f: A $\rightarrow$ B and g: B $\rightarrow$ C. Then the composition function , g $\circ$ f, is a function from A to C defined by (g $\circ$ f)(a) =g(f(a)).
- Note that the function f is applied first and then g.
- Let f: R $\rightarrow$ R be defined by f(x) = x^2.
- Let g: R $\rightarrow$ R be defined by g(x) = $\lfloor x \rfloor$.

(g $\circ$ f)(2.3) = g(f(2.3)) =g($(2.3)^2$) = g(5.29)
= $\lfloor 5.29 \rfloor$ = 5.

Inverse functions

- **Identity function**: the function that maps each element of a set A to itself, denoted by i_A. We have i_A: $A \rightarrow A$.
- Let f: $A \rightarrow B$. If there exists a function g: $B \rightarrow A$ such that $g \circ f = i_a$ and $f \circ g = i_b$, then g is called **the inverse function** of f, denoted by f^{-1}
- **Theorem**: Let f: $A \rightarrow B$. f is a bijection iff f^{-1} exists.
- Example:
  - f: $R \rightarrow R$ given by $f(x) = 3x+4$. $f^{-1} = (x - 4)/3$
  - $(f \circ f^{-1})(x) = 3(x-4)/3 + 4 = x$ identity function

Summary

- We have introduced many concepts,
  - Function
  - Domain, codomain
  - Image, preimpage
  - Range
  - Onto (surjective)
  - One-to-one (injective)
  - Bijection (one-to-one correspondence)
  - Function composition
  - Identity function
  - Inverse function

Logarithm

- A logarithm of base b for value y is the power to which b is raised to get y
  - Definition: $\log_b y = x, b > 0$
  - The mimimum number of bits for encoding a value y

Logarithms have the following properties, for any positive values of m, n, and r, and any positive integers a and b.

1. $\log(nm) = \log n + \log m.$
2. $\log(n/m) = \log n - \log m.$
3. $\log(n^r) = r \log n.$
4. $\log_a n = \log_b n / \log_b a.$

Summations

- Summations are simply the sum of costs for a function with a range of parameter values
  - notated as:

$$\sum_{i=1}^{n} f(i)$$

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}.$$

$$\sum_{i=1}^{n} \frac{1}{2^i} = 1 - \frac{1}{2^n},$$

$$\sum_{i=0}^{n} 2^i = 2^{n+1} - 1.$$

$$\sum_{i=1}^{n} \frac{i}{2^i} = 2 - \frac{n+2}{2^n}.$$

Recursion

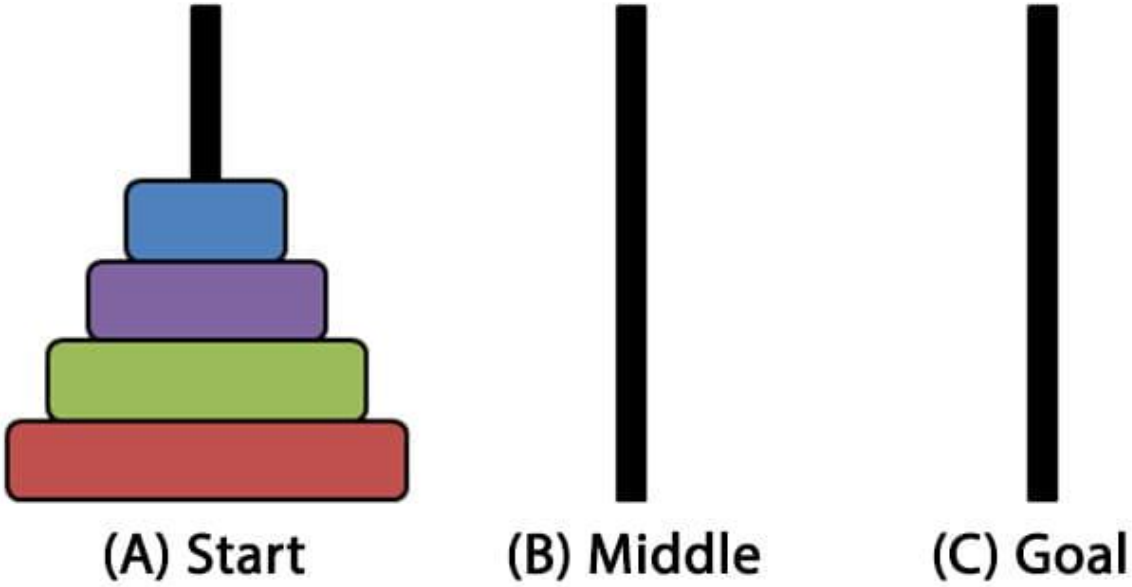
- An algorithm is recursive if it calls itself to do part of its work.
- Example:
  - Compute n!

$$fact(n) = \begin{cases} 1, & n = 0 \\ n * fact(n-1), & n > 0 \end{cases}$$

```
long fact(int n) {          // Compute n! recursively
  // To fit n! into a long variable, we require n <= 12
  Assert((n >= 0) && (n <= 12), "Input out of range");
  if (n <= 1)  return 1; // Base case: return base solution
  return n * fact(n-1);  // Recursive call for n > 1
}
```

Recursion (cont'd ...)

- One more example: Towers of Hanoi puzzle



  - **How can you solve this problem?** (think it)

Recursion (cont'd ...)

- A recursive function contains two parts
  - **Base case**, which can bs sovled easily,
    e.g., fact(n)=1 if n=0.
  - **Recursion case**, including a single or multiple calls for iteself with smaller problem sizes,
    e.g., fact(n)=n*fact(n-1) if n>0

Recursion (cont'd ...)

- fact(n) = n * fact(n-1)
- Recursive fcuntion may be diffcult to understand, the key is as follows:
  - Do not think how function fact(n-1) execute
  - Just assume that fact(n-1) returen the correct results
  - The ideas of *abstract* and *divide and conquer* is very useful in not only algorithm design but also solving many problems in our lives.

Mathematical proof

- Solving any problem has two distinct parts:
  - The investigation and the argument.
- Three templates for mathematical proof
  - Direct proof
    - Logic deduction
  - Proof by contradiction
  - Proof by mathematical induction

Proof by contradiction

- Consider to show that $P \rightarrow Q$, if we can prove (not Q)$\rightarrow$(not P), then $P \rightarrow Q$ holds
  - E.g., prove that there is no largest integer.

Proof:

Step 1: Assume that there is a largest integer, call it *B*.

Step 2: consider that C=B+1. C is an integer and C>B.

B then is not the largest integer. A contradiction happens.

Therefore, the assumption that there is a largest integer is incorrect.

Proof by induction

- Prove S(n) = 1 + 2 + …+n = n(n+1)/2, for n ≥1
  - Proof:
    1. Check the base case. S(1) = 1 = 1(1+1)/2 √
    2. Assume the equation holds for n-1, i.e.,
       S(n-1)=1+2+…+(n-1)=(n-1)(n-1+1)/2=(n-1)n/2
    3. Consider the case for n.

       S(n) = 1+2+…+(n-1)+n

       = S(n-1) + n

       = (n-1)n/2 + n, by the assumption

       = n(n+1)/2 √

Estimation Techniques

Known as “back of the envelope” or “back of the napkin” calculation

1. Determine the major parameters that effect the problem.
2. Derive an equation that relates the parameters to the problem.
3. Select values for the parameters, and apply the equation to yield and estimated solution.

Estimation Example

How many library bookcases does it take to store books totaling one million pages?

Estimate:

- Pages/inch
- Feet/shelf
- Shelves/bookcase

Overall summary

- We talk about mathematical notation, background, and techniques that
  - used throughout the course
  - provided primarily for review and reference.

- You might return to the relevant sections when you encounter unfamiliar notaton or mathematical techniques in later course.

Exercises

- Exercise 1
  - I1.6: 1.3, 1.12, 1.14
- Exercise 2
  - I2.9: 2.3, 2.5, 2.11, 2.17, 2.30, 2.33