Discrete Mathematics		
	Chapter 8	
Instructor: Bow-Yaw Wang		Spring 2008

Relations

1. Relations and Their Properties

Definition 1. *Let A and B be sets. A **binary relation from** A **to** B is a subset of $A \times B$.*

Let R be a binary relation. We sometimes write aRb for $(a, b) \in R$. Examples of binary relations are $<$, $\in$, etc.

Functions belong to a special class of relations. Let $f : A \to B$. Define $F = \{(a, b) : b = f(a)\}$. F is a binary relation from A to B such that aFb and aFb' implies $b = b'$.

Definition 2. *Let A be a set. A **relation on** A is a relation from A to A.*

For instance, $<, =$ are relation on $\mathbb{R}$.

Definition 3. *A relation R on A is called **reflexive** if $(a, a) \in R$ for all $a \in A$.*

For instance, $\leq$ is reflexive.

Definition 4. *A relation R on A is called **symmetric** if $(a, b) \in R$ implies $(b, a) \in R$ for all $a, b \in A$.*
*A relation R on A is called **antisymmetric** if $(a, b) \in R$ and $(b, a) \in R$ implies $a = b$.*

For instance, $\neq$ is symmetric and $\geq$ is antisymmetric.

Definition 5. *A relation R on A is called **transitive** if $(a, b) \in R$ and $(b, c) \in R$ implies $(a, c) \in R$ for all $a, b, c \in R$.*

For instance, $<$ is transitive.
Remark. Do you still remember the "proof" that symmetry and transitivity imply reflexivity?

Definition 6. *Let R be a relation from A to B and S a relation from B to C. The **composite** of R and S, $S \circ R$, is defined by*

$$S \circ R = \{(a, c) : \exists b \in B.(a, b) \in R \wedge (b, c) \in S\}.$$

Remark. When R and S happen to be functions, $S \circ R$ is equivalent to the function composition.

Definition 7. *Let R be a relation on A. Define*

$$R^1 = R \text{ and } R^{n+1} = R^n \circ R.$$

For convenience, we define $R^0 = I$ where I is the identity relation $\{(a, a) : a \in A\}$.

Theorem 1. *The relation R on A is transitive if and only if $R^n \subseteq R$ for $n \in \mathbb{Z}^+$.*

Proof. ($\Leftarrow$) Let $a, b, c \in A$ with aRb and bRc. Then aR^2c by definition. Hence aRc for $R^2 \subseteq R$.
($\Rightarrow$) We prove by induction on n.
BASIS STEP: $n = 1$, $R \subseteq R$ is trivial.
INDUCTIVE STEP: Assume $R^k \subseteq R$. We want to show $R^{k+1} \subseteq R$. Consider any a, c such that $aR^{k+1}c$. There is a b such that aRb and bR^kc because $R^{k+1} = R^k \circ R$. By inductive hypothesis, bR^kc implies bRc. Hence aRc follows from the transitivity of R. □

2. n-ary Relations and Their Applications

Definition 8. *Let $A_0, A_1, \ldots, A_{n-1}$ be sets. An n-**ary relation** on these sets is a subset of $A_0 \times A_1 \times \cdots \times A_{n-1}$. $A_0, A_1, \ldots, A_{n-1}$ are called the **domain** of the relation and n is its **degree**.*

Definition 9. *Let R be an n-ary relation over $A_0, A_1, \ldots, A_{n-1}$ and $C : A_0 \times A_1 \times \cdots \times A_{n-1} \to \{\textbf{false}, \textbf{true}\}$. The **selection operator** $s_C : \wp(A_0 \times A_1 \times \cdots \times A_{n-1}) \to \wp(A_0 \times A_1 \times \cdots \times A_{n-1})$ is defined by*

$$s_C(R) = \{(a_0, \ldots, a_{n-1}) \in R : C(a_0, \ldots, a_{n-1}) = \textbf{true}\}.$$

Definition 10. *The **projection** $P_{i_0 i_1 \cdots i_{m-1}}$ maps the n-tuple $(a_0, a_1, \ldots, a_{n-1})$ to the m-tuple $(a_{i_0}, a_{i_1}, \ldots, a_{i_{m-1}})$, where $m \leq n$ and $0 \leq i_k < n$ for all k.*

Definition 11. *Let R be a relation of degree m and S a relation of degree n. The **join** $J_p(R,S)$, where $p \le m$ and $p \le n$, is a relation of degree $m+n-p$ such that*

$$(a_0, a_1, \ldots, a_{m-p-1}, c_0, c_1, \ldots, c_{p-1}, b_p, b_{p+1}, \ldots, b_{n-1}) \in J_p(R,S)$$

if and only if

$$(a_0, a_1, \ldots, a_{m-p-1}, c_0, c_1, \ldots, c_{p-1}) \in R$$

and

$$(c_0, c_1, \ldots, c_{p-1}, b_p, b_{p+1}, \ldots, b_{n-1}) \in S.$$

We can think databases as n-ary relations. The database query language SQL (Structured Query Language) has the operations we defined here.

Example 1. *Try to interpret the following SQL query:*

```
SELECT Grade
FROM Transcripts
WHERE Department='Information Management'
```

*Solution.*The command `SELECT` corresponds to projection. The clause `WHERE` specifies the condition of the selection operator. Finally, `FROM` denotes the n-ary relation we're interested in. □

Although database queries in SQL seem to be simple from mathematical point of view, they require intensive computation to implement. Consider the example and ask yourself: how many students in the university are there? Note that we have only one transcript database for all these years. It is by no mean "simple" to collect information from the database.

3. Representing Relations

We can represent relations by matrices or graphs.

3.1. Matrix Representation.

Example 2. *Consider any binary relation R on $\{a_0, a_1, \ldots, a_{n-1}\}$. Define $\mathbf{M}_R = [m_{ij}]_{n\times n}$ where*

$$m_{ij} = \begin{cases} 1 & \text{if } (a_i, a_j) \in R \\ 0 & \text{if } (a_i, a_j) \notin R \end{cases}$$

If R is symmetric, $\mathbf{M}_R$ is symmetric. If R is reflexive and $\mathbf{M}_R = [m_{ij}]_{n\times n}$, $m_{ii} = 1$ for $0 \le i < n$.

We introduce the following matrix operations:

Definition 12. *Let $\mathbf{A} = [a_{ij}]_{m\times n}$, $\mathbf{B} = [b_{ij}]_{m\times n}$ and $a_{ij}, b_{ij} \in \{0,1\}$ for $0 \le i < m$, $0 \le j < n$. Define $\mathbf{A} \wedge \mathbf{B} = [c_{ij}]_{m\times n}$ where*

$$c_{ij} = \begin{cases} 1 & \text{if } a_{ij} = 1 \wedge b_{ij} = 1 \\ 0 & \text{otherwise} \end{cases}$$

Definition 13. *Let $\mathbf{A} = [a_{ij}]_{m\times n}$, $\mathbf{B} = [b_{ij}]_{m\times n}$ and $a_{ij}, b_{ij} \in \{0,1\}$ for $0 \le i < m$, $0 \le j < n$. Define $\mathbf{A} \vee \mathbf{B} = [c_{ij}]_{m\times n}$ where*

$$c_{ij} = \begin{cases} 1 & \text{if } a_{ij} = 1 \vee b_{ij} = 1 \\ 0 & \text{otherwise} \end{cases}$$

Definition 14. *Let $\mathbf{A} = [a_{ik}]_{m\times l}$, $\mathbf{B} = [b_{kj}]_{l\times n}$ and $a_{ik}, b_{kj} \in \{0,1\}$ for $0 \le i < m$, $0 \le k < l$, $0 \le j < n$. Define $\mathbf{A} \odot \mathbf{B} = [c_{ij}]_{m\times n}$ where*

$$c_{ij} = \begin{cases} 1 & \text{if } \bigvee_{k=0}^{l-1} a_{ik} = 1 \wedge b_{kj} = 1 \\ 0 & \text{otherwise} \end{cases}$$

When $m = l$, define $\mathbf{A}^{[0]} = \mathbf{I}_n$ and $\mathbf{A}^{[r+1]} = \mathbf{A} \odot \mathbf{A}^{[r]}$.

Let R_0 and R_1 be relations on $\{0, \ldots, n-1\}$. It is straightforward to see $\mathbf{M}_{R_0 \cup R_1} = \mathbf{M}_{R_0} \vee \mathbf{M}_{R_1}$, $\mathbf{M}_{R_0 \cap R_1} = \mathbf{M}_{R_0} \wedge \mathbf{M}_{R_1}$ and $\mathbf{M}_{R_1 \circ R_0} = \mathbf{M}_{R_0} \odot \mathbf{M}_{R_1}$. Note that the order of R_0 and R_1 is reversed in the composite.

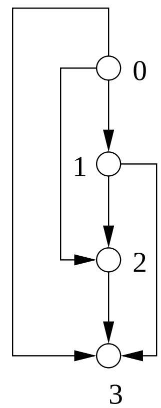

FIGURE 1. Graph Representation for $<$

3.2. **Graph Representation.**

Definition 15. *A **directed graph** (or **digraph**), $G = (V, E)$, consists of the set V of **vertices** and $E \subseteq V \times V$ the set of **edges**. For the edge (a, b), the vertex a is its **initial vertex** and b its **terminal vertex**. The edge (a, a) is called a **loop**.*

Example 3. *Draw a digraph to represent $<$ on $\{0, 1, 2, 3\}$.*

*Solution.*Figure 1 shows the solution. □

4. CLOSURES OF RELATIONS

Definition 16. *Let R be a relation on A. The smallest transitive relation that contains R is called the **transitive closure** of R. Similarly, the smallest reflexive relation that contains R is called the **reflexive closure** of R. And the smallest symmetric relation that contains R is called the **symmetric closure** of R.*

The relation $\Delta_A = \{(a, a) : a \in A\}$ is the **diagonal relation** on A. Note that when $|A| = n$, $\mathbf{M}_{\Delta_A} = \mathbf{I}_n$.

Example 4. *What is the reflexive closure of $<$ on $\mathbb{Z}$?*

*Solution.*Consider $< \cup \Delta_A = \leq$. It is easy to see $\leq$ is the reflexive closure of $<$. □
Remark. What is the symmetric closure of $<$?

Definition 17. *Let $G = (V, E)$ be a digraph. A **path** from v_0 to v_n is a sequence of edges $(v_0, v_1), (v_1, v_2), \ldots, (v_{n-1}, v_n)$ in G. We usually write $v_0, v_1, \ldots, v_n$ to denote the path and say the **length** of path is n. For any $v \in V$, we view the empty sequence as a path (of length 0). If $n > 0$ and $v_0 = v_n$, we say the path is a **circuit** or **cycle**.*

Just like a relation can be represented by a digraph. A digraph corresponds to a relation E on V.

Theorem 2. *Let R be a relation on A. Consider the digraph (A, R). There is a path of length $n > 0$ from a to b if and only if $(a, b) \in R^n$.*

Proof. We prove by induction.
BASIS STEP: $n = 1$. Obvious.
INDUCTIVE STEP: Assume there is a path of length k from c to b if and only if $(c, b) \in R^k$. Consider any path of length $k + 1$ from a to b. The path consists a path of length 1 from a to c and a path of length k from c to b. Hence $(a, c) \in R$ and $(c, b) \in R^k$. We have $(a, b) \in R^{k+1}$.
On the other hand, if $(a, b) \in R^{k+1}$, there exists a c such that $(a, c) \in R$ and $(c, b) \in R^k$. The result follows from the inductive hypothesis as well. □

Definition 18. *Let R be a relation on A. The **connectivity relation** R^+ consists of (a, b) such that there is a path from a to b in the digraph (A, R).*

By Theorem 2, we have

$$R^+ = \bigcup_{n=1}^{\infty} R^n.$$

```
(1) procedure transitive-closure(M_R : zero-one n × n matrix )
(2) A := M_R
(3) B := A
(4) for i := 2 to n do
(5)     A := A ⊙ M_R
(6)     B := B ∨ A
(7) od
(8) { B is the zero-one matrix for R+ }
(9) end
```

FIGURE 2. An Algorithm for Computing Transitive Closure

Theorem 3. *The transitive closure of R equals to R^+.*

Proof. Note that $R \subseteq R^+$ by definition. Furthermore, $(a, b), (b, c) \in R^+$ implies $(a, b) \in R^i$ and $(b, c) \in R^j$ for some i, j. Hence $(a, c) \in R^{i+j} \subseteq R^+$. R^+ is transitive.

It remains to show that any transitive relation containing R must contain R^+. Let S be any transitive relation containing R. Since S is transitive, $S^n \subseteq S$ by Theorem 1 for $n \in \mathbb{Z}^+$. Therefore

$$R^+ = \bigcup_{n=1}^{\infty} R^n \subseteq \bigcup_{n=1}^{\infty} S^n \subseteq S.$$

□

Lemma 1. *Let A be a set with $|A| = n$ and R a relation on A. If there is a path of length > 0 from a to b in the digraph (A, R), then there is a path of length $\leq n$. When $a \neq b$, if there is a path of length > 0 in (A, R), then there is a non-empty path of length $< n$.*

Proof. Consider the shortest path $v_0 = a, v_1, v_2, \ldots, v_m = b$ from a to b of length m. If $v_0 = v_m$ and $m > n$, there must be some i, j with $i < j$ such that $v_i = v_j$ by the pigeonhole principle. Then $v_0, v_1, \ldots, v_i, v_{j+1}, v_{j+2}, \ldots, v_m$ is a shorter path from a to b. A contradiction.

Now suppose $a = v_0 \neq v_m = b$ and $m \geq n$. There must be i, j with $i < j$ such that $v_i = v_j$. We can also construct a shorter path and lead to a contradiction. □

Theorem 4. *Let $\mathbf{M}_R = [m_{ij}]_{n \times n}$ be the matrix of the relation R on a set with n elements. Then the matrix $\mathbf{M}_{R^+}$ of R^+ is*

$$\mathbf{M}_{R^+} = \mathbf{M}_R \vee \mathbf{M}_R^{[2]} \vee \mathbf{M}_R^{[3]} \vee \cdots \vee \mathbf{M}_R^{[n]}.$$

Proof. $(a, b) \in R^+$, then there is a path from a to b by definition. By Lemma 1, it suffices to consider paths of length at most n. By Theorem 2,

$$R^+ = R \cup R^2 \cup \cdots \cup R^n.$$

The result follows from the matrix representation of relations. □

Figure 2 shows an algorithm for computing transitive closure of R. Let's analyze its time complexity. First observe that computing the entry c_{ij} in $\mathbf{A} \odot \mathbf{B}$

$$c_{ij} = \bigvee_{k=0}^{n-1} a_{ik} \wedge b_{kj}$$

requires $O(n)$ steps. Since there are n^2 entries, $\mathbf{A} \odot \mathbf{M}_R$ takes $O(n^3)$ steps. Moreover, $\mathbf{B} \vee \mathbf{A}$ takes $O(n^2)$ steps. Thus an iteration of the loop (line (5) and (6)) takes $O(n^3) + O(n^2) = O(n^3)$ steps. There are n iterations, we have the time complexity $O(n^4)$ for Line (4) to (7). Line (2) and (3) take $O(n^2)$ steps respectively. The time complexity of the algorithm is $O(n^4)$.

We can actually do better than $O(n^4)$. Figure 3 shows the algorithm developed by Warshall. If we let $w_{ij}^{[k]}$ denote the value of w_{ij} at the end of the k-th iteration of outmost loop, then $w_{ij}^{[k]}$ is 1 if and only if there is a path from i to j via vertices $\{v_0, v_1, \cdots, v_k\}$. Line (6) can be rewritten as

$$w_{ij}^{[k]} = w_{ij}^{[k-1]} \vee (w_{ik}^{[k-1]} \wedge w_{kj}^{[k-1]}).$$

We can rephrase it as the equivalence of the following two statements:

```
(1)  procedure Warshall (M_R : n × n zero-one matrix)
(2)  W := M_R
(3)  for k := 0 to n − 1 do
(4)      for i := 0 to n − 1 do
(5)          for j := 0 to n − 1 do
(6)              w_ij := w_ij ∨ (w_ik ∧ w_kj)
(7)          od
(8)      od
(9)  od
(10) { W = [w_ij] is M_R }
(11) end
```

FIGURE 3. Warshall's Algorithm

- There is a path from v_i to v_j via $\{v_0, v_1, \ldots, v_k\}$;
- There is a path from v_i to v_j via $\{v_0, v_1, \ldots, v_{k-1}\}$ or a path from v_i to v_k and v_k to v_j via $\{v_0, v_1, \ldots, v_{k-1}\}$.

Clearly, these two statements are equivalent. Hence line (6) basically tells us whether there is a path from v_i to v_j via $\{v_0, v_1, \ldots, v_k\}$. Since $w_{ij}^{[n-1]}$ denotes that there is a path from v_i to v_j via all vertices, the result follows.

Since line (6) takes $O(1)$ steps and it repeats n^3 times, we conclude the Warshall algorithm has time complexity $O(n^3)$.

5. Equivalence Relations

Definition 19. *A relation R on A is called an* **equivalence relation** *if it is reflexive, symmetric, and transitive.*

Example 5. *Let $n \in \mathbb{Z}^+$. Show $\{(a, b) : a \equiv b(n)\}$ is an equivalence relation.*

Proof. For any $a, b, c \in \mathbb{N}$, we have

- $a \equiv a(n)$;
- $a \equiv b(n)$. Then $n|a - b$. Thus $n|b - a$. So $b \equiv a(n)$;
- $a \equiv b(n)$ and $b \equiv c(n)$. Then $nk = a - b$ and $nk' = b - c$. So $n(k + k') = (a - b) + (b - c) = a - c$, $n|a - c$. We have $a \equiv c(n)$.

□

Definition 20. *Let R be an equivalence relation on A. Let $a \in A$. Define the* **equivalence class of** *a, $[a]_R$, to be*

$$[a]_R = \{b : (a, b) \in R\}.$$

Sometimes, we may write $[a]$ if R is clear from context.

Example 6. *Find all equivalence classes in Example 5.*

Solution.

$$\begin{aligned}
[0] &= \{\ldots, -2n, -n, 0, n, 2n, \ldots\} \\
[1] &= \{\ldots, -2n + 1, -n + 1, 1, n + 1, 2n + 1, \ldots\} \\
&\cdots \\
[n - 1] &= \{\ldots, -2n + (n - 1), -n + (n - 1), (n - 1), n + (n - 1), 2n + (n - 1), \ldots\}
\end{aligned}$$

□

The equivalence classes of the relation congruence modulo n are called **congruence classes modulo** n.

Let A be a set. A **partition** of A is a collection of disjoint nonempty subsets of A. The equivalence classes of R on A form a partition of A. More precisely,

Theorem 5. *Let R be an equivalence relation on A. The following statements are equivalence:*

(i) *aRb;*
(ii) *$[a] = [b]$;*
(iii) *$[a] \cap [b] \neq \emptyset$.*

Proof. ((i)⇒(ii)) Assume aRb. Consider any $c \in [a]$. We have aRc by definition. Then bRc by symmetry and transitivity. So $c \in [b]$. $[a] \subseteq [b]$. $[b] \subseteq [a]$ follows by symmetric arguments. Hence $[a] = [b]$.

((ii)⇒(iii)) Trivial.

((iii)⇒(i)) Let $c \in [a] \cap [b]$. Then aRc and bRc. aRb follows by symmetry and transitivity. □

Since $\{[0], [1], [2], \ldots, [n-1]\}$ forms a partition of $\mathbb{Z}$, any integer must belong to one and only one of the congruence classes.

Conversely, we can obtain an equivalence relation from a partition.

Theorem 6. *Let A be a set and $\{A_i : i \in I\}$ a partition of A. Define*

$$R = \{(a, b) : \exists i. a \in A_i \wedge b \in A_i\}.$$

Then R is an equivalence relation with $A_i, i \in I$ its equivalence classes.

Proof. We have

- aRa. Since $\{A_i\}$ is a partition, $a \in A_i$ for some i.
- aRb implies bRa. Since the definition of R is symmetric, the result follows.
- aRb and bRc implies aRc. By definition, there are i, j such that $a \in A_i \wedge b \in A_i$ and $b \in A_j \wedge c \in A_j$. But $\{A_i\}$ is a partition, $i = j$. We have $a \in A_i \wedge c \in A_i$, aRc.

The equivalences classes of R follows by definition. □

6. Partial Orderings

Definition 21. *Let R be a relation on A. R is called* ***partial ordering*** *or* ***partial order*** *if it is reflexive, antisymmetric, and transitive. The set A is called a* ***partially ordered set****, or* ***poset****, and is denoted by (A, R).*

For instance, $(\mathbb{Z}, \geq)$ is a poset. And $(\wp(S), \subseteq)$ is a poset for any set S. Note that not all pairs of elements can be ordered; neither $\{0\} \subseteq \{1\}$ nor $\{1\} \subseteq \{0\}$.

Definition 22. *Let $(S, \preceq)$ be a poset and $a, b \in S$. a and b are called* ***comparable*** *if either $a \preceq b$ or $b \preceq a$. Otherwise, they are called* ***incomparable****.*

Notation. We will write $a \prec b$ for $a \preceq b$ but $a \neq b$.

When all pairs of elements are comparable, we call the relation a **total ordering**.

Definition 23. *If $(S, \preceq)$ is a poset and every two elements of S are comparable, S is called a* ***totally ordered*** *or* ***linearly ordered set****, and $\preceq$ is called a* ***total order*** *or a* ***linear order****. A totally ordered set is also called a* ***chain****.*

For instance, $(\mathbb{Z}, \leq)$ is a chain.

Definition 24. *Let $(S, \preceq)$ be a poset. $(S, \preceq)$ is a* ***well-ordered set*** *if $\preceq$ is a total ordering such that every nonempty subset of S has a least element (according to $\preceq$).*

For instance, $(\mathbb{Z}^+, \leq)$ is a well-ordered set but $(\mathbb{Z}, \leq)$ is not.

Example 7. *Consider the relation $\preceq \subseteq \mathbb{Z}^+ \times \mathbb{Z}^+$ such that $(a, b) \preceq (a', b')$ if $a < a'$, or $a = a' \wedge b \leq b'$. Then $(\mathbb{Z}^+ \times \mathbb{Z}^+, \preceq)$ is a well-ordered set.*

*Solution.*Consider any subset A of $\mathbb{Z}^+ \times \mathbb{Z}^+$. Define A_0 to be the projection of A on the x-coordinate. $A_0 \subseteq \mathbb{Z}^+$. Then A_0 has a least element, say a_0. Define A_1 to be the selection of A on the condition that the x-coordinate equals to a_0. Let A_2 be the projection of A_1 on the y-coordinate. Then $A_2 \subseteq \mathbb{Z}^+$. Hence it has a least element a_2. Then (a_0, a_2) is a least element of A. □

Theorem 7. *(The Principle of Well-Ordered Induction) Let $(S, \preceq)$ be a well-ordered set. Then $P(x)$ is true for all $x \in S$ if*

BASIS STEP: $P(x_0)$ is true for the least element of S, and

INDUCTIVE STEP: For every $y \in S$ if $P(x)$ is true for all $x \prec y$, then $P(y)$ is true.

Proof. Consider the set $A = \{y : \neg P(y)\}$. $A \subseteq S$, A has a least element y_0 for S is well-ordered. y_0 cannot be the least element of S because of basis step. Consider the set $B = \{x : x \prec y\}$. B is not empty and $\forall x \in B. P(x)$ by the choice of y_0. Then $P(y_0)$ holds by the inductive step. A contradiction. □

The following example represents a partially ordered relation by a undirected graph. The graph is called a **Hasse diagram**.

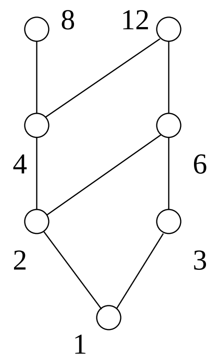

FIGURE 4. A Hasse Diagram

Example 8. *Draw the Hasse diagram for $a|b$ where $a, b \in \{1, 2, 3, 4, 6, 8, 12\}$.*

*Solution.*Figure 4 shows the diagram.

Definition 25. *Let $(S, \preceq)$ be a poset and $a \in S$. Then a is* ***maximal*** *in $(S, \preceq)$ if there is no $b \in S$ such that $a \prec b$. Similarly, a is* ***minimal*** *in $(S, \preceq)$ if there is no $b \in S$ such that $b \prec a$.*

Example 9. *What are the maximal and minimal elements of the poset $(\{2, 3, 4, 5, 10, 12, 15, 20, 24\}, |)$?*

*Solution.*The maximal elements are $15, 20, 24$. And the minimal elements are $2, 3, 5$. □

Definition 26. *Let $(S, \preceq)$ be a poset and $a \in S$. Then a is the* ***greatest element*** *of $(S, \preceq)$ if $b \preceq a$ for all $b \in S$. Similarly, a is the* ***least element*** *of $(S, \preceq)$ if $a \preceq b$ for all $b \in S$.*

Example 10. *What are the greatest and least elements of the poset $(\wp(S), \subseteq)$?*

Solution.S and $\emptyset$ are the greatest and least elements of $(\wp(S), \subseteq)$ respectively. □

Remark. There are several maximal and minimal elements in a poset. But there is at most one greatest and one least element in a poset. (why?)

Definition 27. *Let $(S, \preceq)$ be a poset and $A \subseteq S$. An element $u \in S$ is called an* ***upper bound*** *of A if $a \preceq u$ for all $a \in A$. Similarly, an element $l \in S$ is called a* ***lower bound*** *of A if $l \preceq a$ for all $a \in A$.*

Definition 28. *Let $(S, \preceq)$ be a poset and $A \subseteq S$. An element x is called the* ***least upper bound*** *if x is an upper bound and $x \preceq u$ for all upper bound u of A. Similarly, an element y is called the* ***greatest lower bound*** *if y is a lower bound and $l \preceq y$ for all lower bound l of A.*

Definition 29. *A partially ordered set where every pair of elements has both a least upper bound and a greatest lower bound is called a* ***lattice****.*

Example 11. *Is the poset $(\mathbb{Z}^+, |)$ a lattice?*

*Solution.*Let $a, b \in \mathbb{Z}^+$. The $\text{lcm}(a, b)$ and $\gcd(a, b)$ are the least upper bound and greatest lower bound of $\{a, b\}$ respectively. $(\mathbb{Z}^+, |)$ is a lattice. □

Example 12. *Let S be a set. Is the poset $(\wp(S), \subseteq)$ a lattice?*

*Solution.*Let $A, B \subseteq S$. Then $A \cup B$ and $A \cap B$ are the least upper bound and greatest lower bound of $\{A, B\}$ respectively. $(\wp(S), \subseteq)$ is a lattice. □

An important application of partially ordered sets is the denotational semantics of programming languages. In denotational semantics, the meaning of a statement is defined as a monotonic function over a partially ordered set. Program constructs are interpreted by operations on such monotonic functions.

6.1. **Topological Sorting.** Given a partial ordering $\preceq$, we are interested in finding a total ordering **compatible** with $\preceq$, that is, $a \leq b$ if $a \preceq b$ for any a, b. We will need the following lemma.

Lemma 2. *Every finite nonempty poset $(S, \preceq)$ has at least one minimal element.*

procedure *topological-sort* $((S, \preceq)$: finite poset)
$k := 1$
while $S \neq \emptyset$ **do**
    $a_k :=$ a minimal element of S
    $S := S - \{a_k\}$
    $k := k + 1$
od
$\{\ a_1, a_2, \ldots, a_n$ is a compatible total ordering of $\preceq\ \}$

FIGURE 5. Topological Sorting

Proof. Choose any element a_0 from S. If a_0 is not minimal, there is an element $a_1 \in S$ with $a_1 \preceq a_0$. More generally, if a_i is not minimal, there is an $a_{i+1} \in S$ such that $a_{i+1} \preceq a_i$. Since there are only finitely many elements in S, this process must terminate with a minimal element a_n. □

We can now present the algorithm for finding a total ordering compatible with the partial ordering $\preceq$ in any poset $(S, \preceq)$.