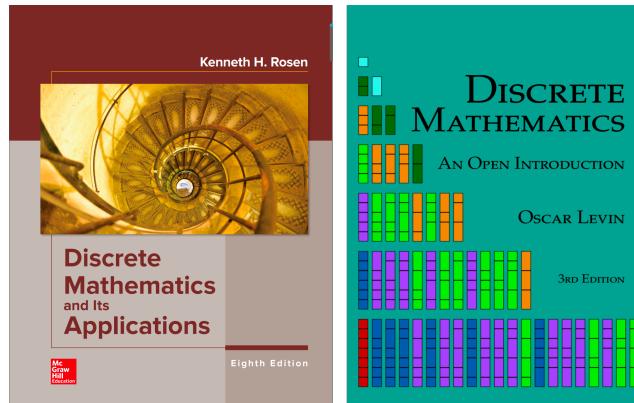




Vietnam National University of HCMC
International University
School of Computer Science and Engineering



Session 10 Advanced Counting

Assoc. Prof. Dr. Nguyen Van Sinh

nvsinh@hcmiu.edu.vn

Contains

Why advanced counting?

Many counting problem cannot be solved easily by using the methods discussed in sessions 8 and 9 such as:

- How many bit strings of length n do not contain two consecutive zeros?
- How many ways are there to assign 7 jobs to 3 employees so that each employee is assigned at least one job?

In order to solve this problem, we can base on the “Recurrence Relation”.

In this section, we are working on two points:

- Recurrence relations.
- Solving linear recurrence relations

A. Recurrence Relation

Definition: A recurrence relation for the sequence $\{a_n\}$ is an equation that expresses a_n in terms of one or more of the previous terms of the sequence, namely, a_0, a_1, \dots, a_{n-1} for all integers n with $n \geq n_0$, where n_0 is a nonnegative integer. A sequence is called a solution of a recurrence relation if its terms satisfy the recurrence relation.

Example:

Let $\{a_n\}$ be a sequence that satisfies the recurrence relation $a_n = a_{n-1} - a_{n-2}$ for $n = 2, 3, 4, \dots$, and suppose that $a_0 = 3$ and $a_1 = 5$. What are a_2 and a_3 ?

Solution:

We see from the recurrence relation that $a_2 = a_1 - a_0 = 5 - 3 = 2$, and $a_3 = a_2 - a_1 = 2 - 5 = -3$. Similarly: $a_4 = a_3 - a_2 = -3 - 2 = -5$.

A. Recurrence Relation

Recursively defined sequences are also known as **recurrence relations**. The actual sequence is a **solution** of the recurrence relations.

Ex: Consider the recurrence relation: $a_{n+1} = 2a_n$ ($n > 0$)
[Given $a_1=1$]

The **solution** is: 1, 2, 4, 8, 16, i.e. $a_n = 2^{n-1}$

So, $a_{30} = 2^{29}$

Example:

Compound Interest:

A person deposits \$10,000 in a savings account that yields 11% interest annually. How much will be there in the account After 30 years?

Solution: the definition is recursive.

Let P_n = account balance after n years.

$$\text{Then } P_n = P_{n-1} + 0.11 P_{n-1} = (1.11)P_{n-1}$$

Initial condition is $P_0 = 10,000\$$

$$P_1 = (1.11)P_0 ; \quad P_2 = (1.11)P_1 = (1.11)^2 P_0$$

$$P_3 = (1.11)P_2 = (1.11)^3 P_0$$

...

$$P_n = (1.11)P_{n-1} = (1.11)^n P_0$$

$$P_{30} = (1.11)^{30} 10,000 = 228922.96$$

Example:

Compute population:

Suppose the population of VN now (2022) are 96 millions. The rate of increasing annually is 0.2%. Compute the population of VN in year 2025

Solution:

Let P_n is population n years after 2022.

We have recurrence relation: $P_n = P_{n-1} + 0.002 P_{n-1}$

Or $P_n = 1.002P_{n-1}$

...

$$P_{\text{final}} = (1.002)^3 96 = ? \text{millions}$$

Example:

Fibonacci sequence:

$$a_n = a_{n-1} + a_{n-2} \quad (n > 2) \quad [\text{Given } a_1 = 1, a_2 = 1]$$

$$a_3 = a_2 + a_1 = 1 + 1 = 2$$

$$a_4 = a_3 + a_2 = 2 + 1 = 3$$

$$a_5 = a_4 + a_3 = 3 + 2 = 5$$

$$a_6 = a_5 + a_4 = 5 + 3 = 8$$

...

$$a_n = a_{n-1} + a_{n-2}$$

Example:

Find the **number of bit strings of length n** that *do not have two consecutive 0s*.

Solution:

Let $b = b_1 b_2 \dots b_n$ be a bit string. Let a_n is a number of bit strings b . Consider two following cases:

- a) $b_n = 1$: number of bit strings b equal number of bit strings $b_1 b_2 \dots b_{n-1}$ that have no 00 and equal a_{n-1} .
- b) $b_n = 0$, then $b_{n-1} = 1$: number of bit strings b equal number of bit strings $b_1 b_2 \dots b_{n-2}$ that have no 00 and equal a_{n-2} .

Example:

Following the sum rule, we have the recurrence relation: $a_n = a_{n-1} + a_{n-2}$ with two first values: $a_1 = 2$ and $a_2 = 3$

Example:

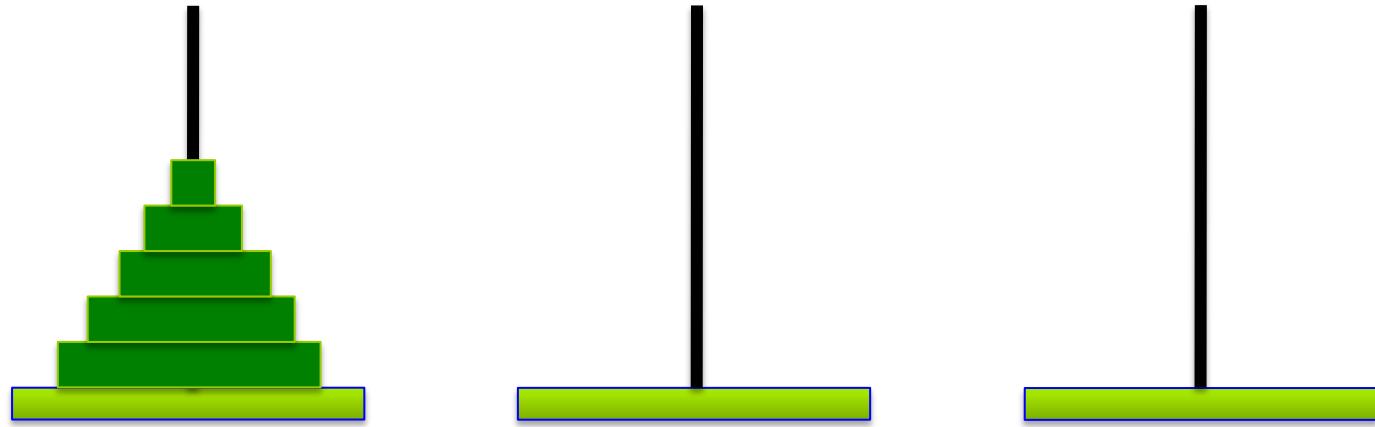
For $n = 1$, the strings are **0** and **1** (a_1)

For $n = 2$, the strings are **01**, **10**, **11** (a_2)

For $n = 3$, the strings are **011**, **111**, **101**, **010**, **110** (a_3)

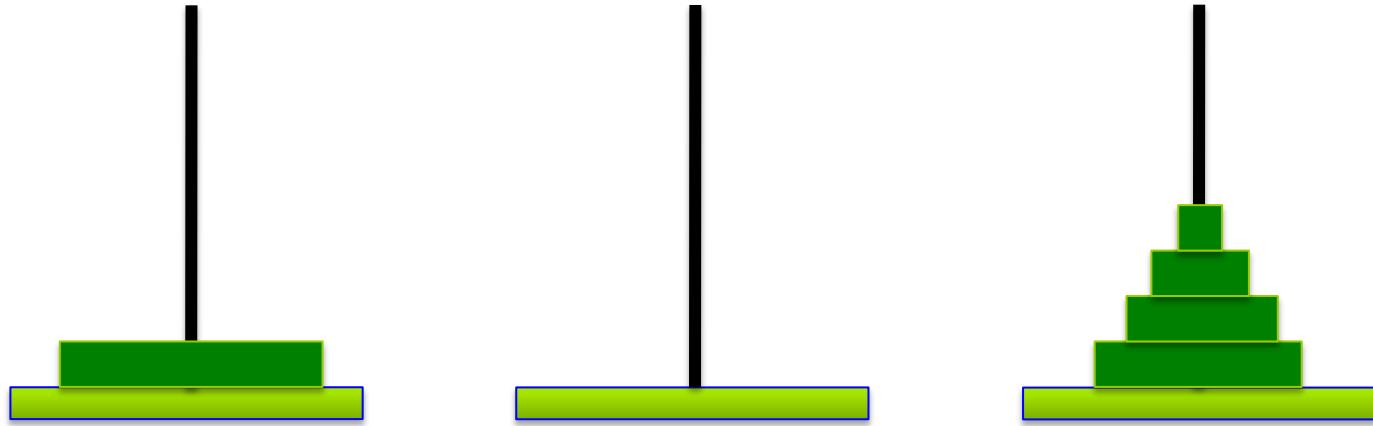
Could you compute for $n = 4?$, $n = 5?$, ...?

Tower of Hanoi



Transfer these disks from one peg to another. However, **at no time, a disk should be placed on another disk of smaller size**. Start with **64 disks**. When you have finished transferring them one peg to another, the world will end.

Tower of Hanoi



Let, H_n = number of moves to transfer n disks. Then

$$H_n = 2H_{n-1} + 1$$

This is a recurrence relation with degree 1, so we need the first value is $H_1 = 1$. However, it is not a final solution.

Tower of Hanoi

We can use an iterative approach to solve this recurrence relation. Note that:

$$\begin{aligned}H_n &= 2H_{n-1} + 1 \\&= 2(2H_{n-2} + 1) + 1 = 2^2H_{n-2} + 2 + 1 \\&= 2^2(2H_{n-3} + 1) + 2 + 1 = 2^3H_{n-3} + 2^2 + 2 + 1 \\&\vdots \\&= 2^{n-1}H_1 + 2^{n-2} + 2^{n-3} + \dots + 2 + 1 \\&= 2^{n-1} + 2^{n-2} + \dots + 2 + 1 \\&= 2^n - 1.\end{aligned}$$

Therefore, with 64 disks, the number of transferring is: $H_{64} = 2^{64} - 1 = 18,446,744,073,709,551,615$

see: *HanoiT.cpp*

Tower of Hanoi

```
void move(int n, int Col1, int Col3, int Col2)
{
    if(n>0)
    {
        move((n-1),Col1,Col2,Col3);
        cout<<"Move a disk from Col1: "<<fromTower <<
              "to Col3: "<<toTower<<"\n";
        move((n-1),Col2,Col3,Col1);
    }
}
```

see: *HanoiT.cpp*

B. Solving Linear Recurrence Relations

A *linear* recurrence relation is of the form

$$a_n = c_1 \cdot a_{n-1} + c_2 \cdot a_{n-2} + c_3 \cdot a_{n-3} + \dots + c_k \cdot a_{n-k}$$

(where c_1, c_2, \dots, c_n are constants)

Its solution is of the form $a_n = r^n$ (where r is a constant) if and only if r is a solution of

$$r^n = c_1 \cdot r^{n-1} + c_2 \cdot r^{n-2} + c_3 \cdot r^{n-3} + \dots + c_k \cdot r^{n-k}$$

This equation is known as the *characteristic equation*.

B. Solving Linear Recurrence Relations

Solving recurrence relation is finding an equation for general a_n so that no need to compute by k previous elements.

There are two methods:

- Substitution method
- Using the characteristic equation

Substitution method

Example: Let C_n is the number of transferring of n disk in Hanoi Tower. We have recurrence relation:

$$\begin{aligned}C_n &= 2C_{n-1} + 1 \text{ and } C_1 = 1 \\&= 2^2C_{n-2} + 2 + 1 \\&= 2^{n-1}C_1 + 2^{n-2} + \dots + 2^2 + 2 + 1 \\&= 1 + 2 + \dots + 2^{n-1} = 2^n - 1\end{aligned}$$

Substitution method

In the **Compound Interest**: suppose Interest annually is “I”; initial money is “M”, after n years we have recurrence relation:

$M_n = M_{n-1} + IM_{n-1} = (1+ I)M_{n-1}$. This is a recurrence relation, degree 1 with $M_0 = M$.

$$\begin{aligned}M_n &= (1+ I)M_{n-1} \\&= (1+ I)^2M_{n-2}\end{aligned}$$

...

$$=(1+ I)^nM_0$$

$$\text{So } M_n = (1+ I)^nM.$$

Characteristic Equation

In order to solve linear homogenous recurrence relation, degree 2 with constant coefficient:

$a_n = c_1 a_{n-1} + c_2 a_{n-2}, c_2 \neq 0$ (1) with first value $a_0 = I_0, a_1 = I_1$
we have characteristic equation: $x^2 = c_1 x + c_2$ (2)

Theorem 1:

If α_1 and α_2 are two distinguish results of (2), then exist only constants b and d so that $a_n = b\alpha_1^n + d\alpha_2^n$

Theorem 2:

If α is a double result of (2), then exist only constants b and so that $a_n = b\alpha^n + dn\alpha^n$

Example 1

Solve: $a_n = a_{n-1} + 2 a_{n-2}$ (Given that $a_0 = 2$ and $a_1 = 7$)

Its solution is of the form $a_n = r^n$

The **characteristic equation** is: $r^2 - r - 2 = 0$. It has two results $r = 2$, and $r = -1$

The sequence $\{a_n\}$ is a solution to this recurrence relation iff

$$a_n = \alpha_1 2^n + \alpha_2 (-1)^n$$

$$a_0 = 2 = \alpha_1 + \alpha_2$$

$$a_1 = 7 = \alpha_1 \cdot 2 + \alpha_2 \cdot (-1) \quad \text{This leads to } \alpha_1 = 3; \alpha_2 = -1$$

So, the solution is $a_n = 3 \cdot 2^n - (-1)^n$

Example 2: Fibonacci sequence

Solve: $f_n = f_{n-1} + f_{n-2}$ (Given that $f_0 = 0$ and $f_1 = 1$)

Its solution is of the form $f_n = r^n$

The **characteristic equation** is: $r^2 - r - 1 = 0$. It has two roots

$$r = \frac{1}{2}(1 + \sqrt{5}) \text{ and } \frac{1}{2}(1 - \sqrt{5})$$

The sequence $\{a_n\}$ is a solution to this recurrence relation iff

$$f_n = \alpha_1 \left(\frac{1}{2}(1 + \sqrt{5})\right)^n + \alpha_2 \left(\frac{1}{2}(1 - \sqrt{5})\right)^n$$

(Now, compute α_1 and α_2 from the initial conditions): $\alpha_1 = 1/\sqrt{5}$ and $\alpha_2 = -1/\sqrt{5}$

The final solution is $f_n = 1/\sqrt{5} \cdot \left(\frac{1}{2}(1 + \sqrt{5})\right)^n - 1/\sqrt{5} \cdot \left(\frac{1}{2}(1 - \sqrt{5})\right)^n$

Relations (chapter 9, page 599)

- Definition
- Representing Relations
- Properties of Relations
- Operations on relations

What is a relation?

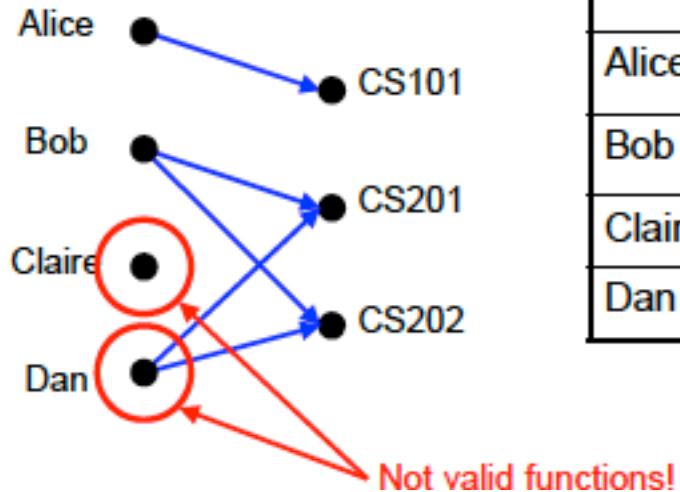
- Let A and B be sets. A binary relation R is a subset of $A \times B$
- Example
 - Let A be the students in a the CS major
 - $A = \{\text{Alice, Bob, Claire, Dan}\}$
 - Let B be the courses the department offers
 - $B = \{\text{CS101, CS201, CS202}\}$
 - We specify relation $R = A \times B$ as the set that lists all students $a \in A$ enrolled in class $b \in B$
 - $R = \{ (\text{Alice, CS101}), (\text{Bob, CS201}), (\text{Bob, CS202}), (\text{Dan, CS201}), (\text{Dan, CS202}) \}$
- If $|A| = m$, $|B| = n$, how many different relations?
- Answer: 2^{mn}

What is a relation?

- Another relation example:
 - ❖ Let A be the set of cities in the U.S.A.
 - ❖ Let B be the set of states in the U.S.A.
 - ❖ We define the relation R to mean x is a city in state y
 - ❖ thus, the following are in our relation:
 - (Middletown, New Jersey)
 - (Middletown, New York)
 - (Cupertino, California)
 - etc.,
- Most relations we know deal with ordered pairs of integers

Representing Relations

We can represent relations graphically:

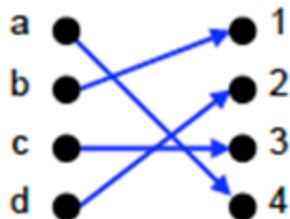


We can represent relations in a table:

	CS101	CS201	CS202
Alice	X		
Bob		X	X
Claire			
Dan		X	X

Relations vs. Functions

- Not all relations are functions
- But consider the following function:



- All functions are relations!
- Any function f from A to B is relation R_f satisfying
 - For every x in A , there exists y in B , $(x, y) \in R_f$.
 - If both $(x, y), (x, z)$ are in R_f , then $y = z$.

When to use which?

A **function** yields a **single result** for any element in its domain.

Example: **age** (of a person), **square** (of an integer) etc.

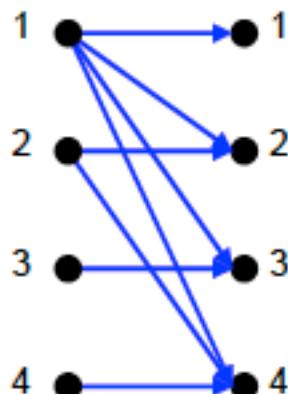
A **relation** allows **multiple mappings** between the domain and the co-domain.

Example: students enrolled in multiple courses.

Relation within a set

A relation on the set A is a relation from A to A

- In other words, the domain and co-domain are the same set
- We will generally be studying relations of this type
 - Let A be the set $\{ 1, 2, 3, 4 \}$
 - Which ordered pairs are in the relation $R = \{ (a,b) \mid a \text{ divides } b \}$
 - $R = \{ (1,1), (1,2), (1,3), (1,4), (2,2), (2,4), (3,3), (4,4) \}$



R	1	2	3	4
1	X	X	X	X

Properties of Relations

We study six properties of relations:

- Reflexive
- Irreflexive
- Symmetric
- Asymmetric
- Antisymmetric
- Transitive

What are these?

Reflexivity

A relation is reflexive if every element is related to itself

- Or, $(a,a) \in R$

Example.

$=$ is reflexive, since $a = a$

\leq is reflexive, since $a \leq a$

$<$ is **not reflexive** is $a < a$ is false.

A relation is irreflexive if every element is *not* related to itself

- Or, $(a,a) \notin R$
- Irreflexivity is the opposite of reflexivity

Symmetry

- A relation is symmetric if, for every $(a,b) \in R$, then $(b,a) \in R$
- Examples of symmetric relations:
 - $=$, sibling(x, y), friend(x, y)
- Examples of relations that are not symmetric:
 - $<$, $>$, \leq , \geq
- A relation is asymmetric if, for every $(a,b) \in R$, then $(b,a) \notin R$
 - Asymmetry is the opposite of symmetry
- Examples of asymmetric relations:
 - $<$, $>$, parent(x, y)

Anti-symmetry

- A relation is antisymmetric if $a=b$ whenever both $(a,b) \in R$ and $(b,a) \in R$.
 - Antisymmetry is *not* the opposite of symmetry
- Examples of antisymmetric relations:
 - $=, \leq, \geq$
- Examples of relations that are not antisymmetric:
 - friend()

More on symmetric relations

A relation can be neither symmetric nor asymmetric

- $R = \{ (a,b) \mid a=|b| \}$
- This is not symmetric
 - $(4, -4)$ is in R but $(-4, 4)$ is not.
- This is not asymmetric
 - $(4, 4)$ is in R .
- This is antisymmetric
 - If both (a, b) and (b, a) in R , then $a = b$.

Transitivity

- A relation is transitive if, for every $(a,b) \in R$ and $(b,c) \in R$, then $(a,c) \in R$
- If $a < b$ and $b < c$, then $a < c$
 - Thus, $<$ is transitive
- If $a = b$ and $b = c$, then $a = c$
 - Thus, $=$ is transitive

Examples of transitive relations

- Consider $\text{Ancestor}(x, y)$: x is an ancestor of y
 - Let Alice be Bob's parent, and Bob be Claire's parent
 - Thus, Alice is an ancestor of Bob, and Bob is an ancestor of Claire
 - Thus, Alice is an ancestor of Claire
 - Thus, Ancestor is a transitive relation

- Consider $\text{Parent}(x, y)$
 - Let Alice be Bob's parent, and Bob be Claire's parent
 - Thus, Alice is a parent of Bob, and Bob is a parent of Claire
 - However, Alice is *not* a parent of Claire
 - Thus, Parent() is *not* a transitive relation

Summary of properties

	=	<	>	≤	≥
Reflexive	X			X	X
Irreflexive		X	X		
Symmetric	X				
Asymmetric		X	X		
Antisymmetric	X			X	X
Transitive	X	X	X	X	X

Operations on relations

Let $A = \{1, 2, 3\}$ and $B = \{1, 2, 3, 4\}$. Define two relations

$$R_1 = \{(1,1), (1,2), (1,3)\}$$

$$R_2 = \{(1,1), (1,3), (1,4)\}$$

Then,

$$R_1 \cup R_2 = \{(1,1), (1,2), (1,3), (1,4)\}$$

$$R_1 \cap R_2 = \{(1,1), (1,3)\}$$

$$R_1 - R_2 = \{(1,2)\}$$

More operations on relations: Composition

Let **S** be a relation from the **set A** to the **set B**, and **R** be a relation from the **set B** to the **set C**. Then, the composition of S and R, denoted by $S \circ R$ is

$$\{(a, c) \mid a \in A, b \in B, c \in C \text{ such that } (a, b) \in S \text{ and } (b, c) \in R\}$$

EXAMPLE. Let $A = \{1, 2, 3\}$, $B = \{1, 2, 3, 4\}$, $C = \{0, 1, 2\}$

$$S = \{(1, 1), (1, 4), (2, 3), (3, 1), (3, 4)\}$$

$$R = \{(1, 0), (2, 0), (3, 1), (3, 2), (4, 1)\}$$

$$\text{Then } S \circ R = \{(1, 0), (1, 1), (2, 1), (2, 2), (3, 0), (3, 1), \dots\}$$

(see Example 10, page 155)

More operations on relations: Composition

$$R^n = R^{n-1} \circ R = R \circ R \circ R \circ R \dots (n \text{ times})$$

EXAMPLE. Let $R = \{(1,1), (2,1), (3,2), (4,3)\}$. Then

$$R^2 = R \circ R = \{(1,1), (2,1), (3,1), (4,2)\}$$

$$R^3 = R^2 \circ R = \{(1,1), (2,1), (3,1), (4,1)\}$$

$$R^4 = R^3 \circ R = \{(1,1), (2,1), (3,1), (4,1)\}$$

Notice that in this case for all $n > 3$, $R^n = R^3$

n-ary relations

Has important applications in computer databases.

DEFINITION. Let $A_1, A_2, A_3, \dots, A_n$ be n sets. An **n-ary relation** is a subset of $A_1 \times A_2 \times A_3 \times \dots \times A_n$

EXAMPLE. R is a relation on $N \times N \times N$ consisting of triples (a, b, c) where $a < b < c$. Thus $(1, 2, 3) \in R$ but $(3, 6, 2) \notin R$

Relational Data Model

Student Record

Name	ID	Major	GPA
Alice	211 324	Physics	3.67
Bob	123 456	ECE	3.67
Carol	351 624	ECE	3.75
David	000 888	Computer Science	3.25

The above table can be viewed as a 4-ary relation consisting of the 4-tuples

(Alice, 211324, Physics, 3.67)

(Bob, 123456, ECE, 3.67)

(Carol, 351624, ECE, 3.75)

(David, 000888, Computer Science, 3.25)

Relational Data Model

Name	ID	Major	GPA
Alice	211 324	Physics	3.67
Bob	123 456	ECE	3.67
Carol	351 624	ECE	3.75
David	000 888	Computer Science	3.25

A **domain** is called a *primary key* when no two n-tuples in the relation have the same value from this domain.
(These are marked red).

Operations on n-ary relations

SELECTION

Let R be an n -ary relation, and C be a condition that the elements in R must satisfy. Then the **selection operator** S_C maps the n -ary relation R to the n -ary relations from R that satisfy the condition C . Essentially it helps filter out tuples that satisfy the desired properties. For example, you may filter out the tuples for all students in ECE, or all students whose GPA exceeds 3.5.

Operations on n-ary relations

PROJECTION

The projection $P_{i,j,k,\dots,m}$ maps each n-tuple $(a_1, a_2, a_3, \dots, a_n)$ to the tuple $(a_i, a_j, a_k, \dots, a_m)$.

Essentially it helps you **delete some of the components** of each n-tuple. Thus, in the table shown earlier, the projection $P_{1,4}$ will retain only that part of the table that contains the student names and their GPAs.

Use of the operations on n-ary relations

SQL queries

SQL queries carry out the operations described earlier:

```
SELECT GPA
```

```
FROM Student Records
```

```
WHERE Department = Computer Science
```

Representing Relations Using Matrices

A relation between finite sets can be represented using a 0-1 matrix. Let $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3\}$. A relation R from A to B can be represented by a matrix M_R , where $m_{ij} = 1$ if $(a_i, b_j) \in R$, otherwise $m_{ij} = 0$

	b1	b2	b3
a1	0	0	0
a2	1	0	0
a3	1	1	0

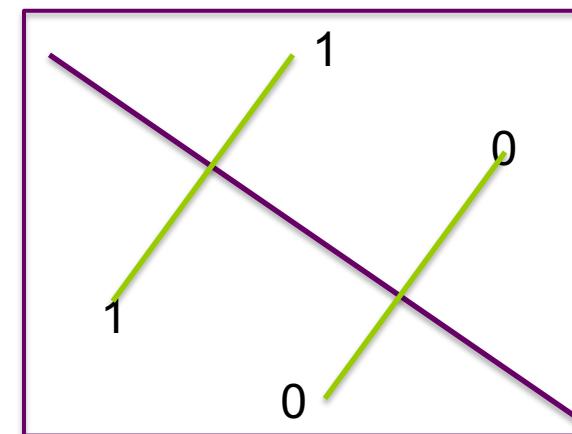
The above denotes a relation R from $A = \{1,2,3\}$ to $B = \{1,2,4\}$, where for each element (a, b) of R , $a > b$

Representing Relations Using Matrices

A reflexive relation on a given set A is recognized by a 1 along the diagonal

1	0	0
1	1	0
1	1	1

A reflexive relation



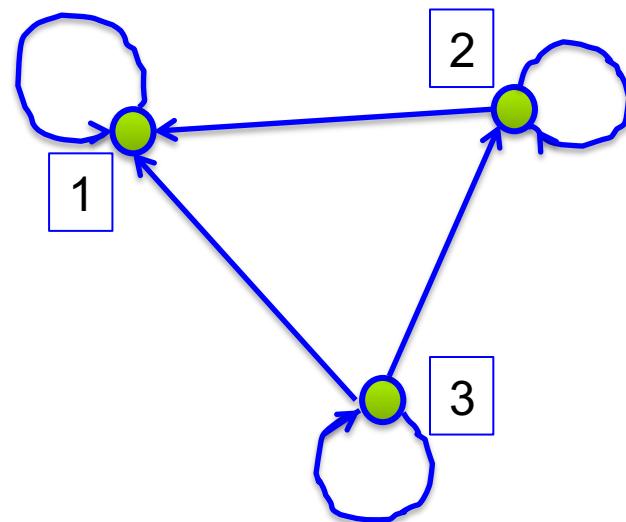
A symmetric relation

Representing Relations Using Digraph

A **relation** on a given set **A** can also be represented by a directed graph

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

Let $A = \{1, 2, 3\}$



A **directed graph** representation of the relation shown on the left

Equivalence Relations

An **equivalence relation** on a **set S** is a relation that is **reflexive, symmetric** and **transitive**.

Examples are:

(1) Congruence relation $R = \{(a,b) \mid a = b \text{ (mod } m)\}$

(2) $R = \{(a, b) \mid L(a) = L(b)\}$ in a set of strings of English characters}, $L(a)$ denotes the length of English character string “a”

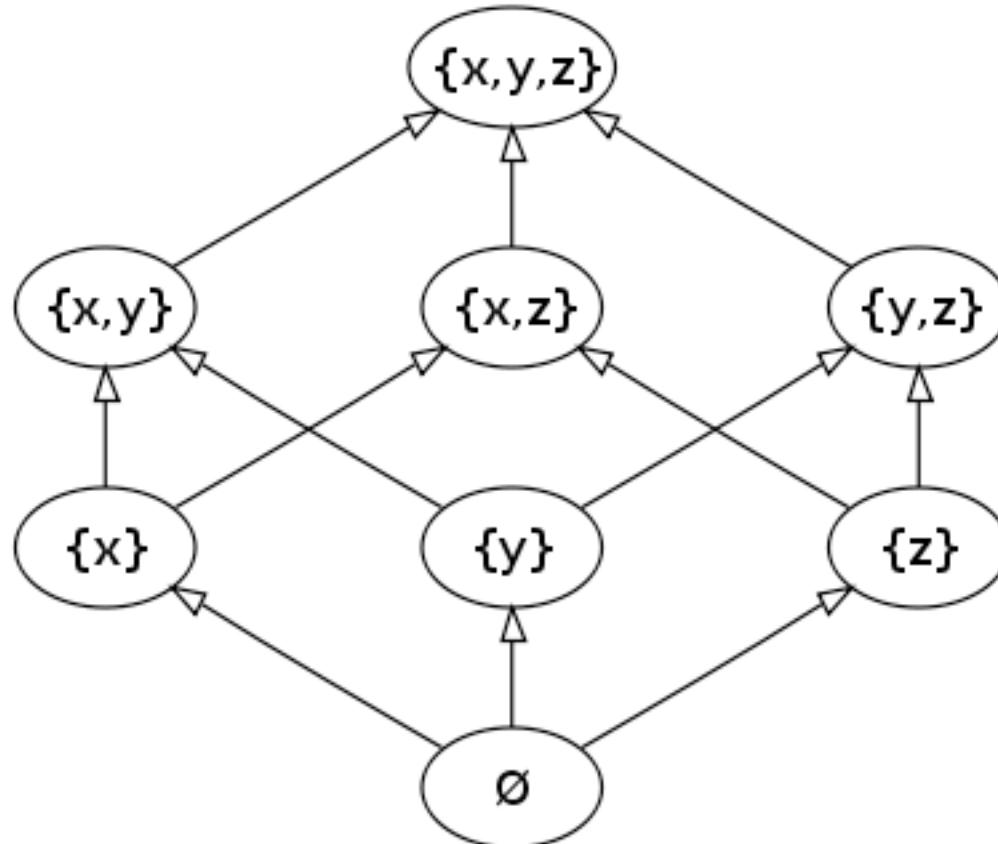
Partial Orders

A relation R on a set S is a **partial order** if it is **reflexive**, **anti-symmetric** and **transitive**. The set is called a **partially ordered set**, or a **poset**.

Examples are

- (1) the \geq relation,
- (2) “ x divides y ” on the set of positive integers
- (3) The relation \subseteq on the power set of a set S

Partial Orders



The relation \subseteq on the power set of a set S forms a partially ordered set

Source: http://en.wikipedia.org/wiki/Partially_ordered_set