San José State University
Department of Computer Science

Ahmad Yazdankhah
ahmad.yazdankhah@sjsu.edu
www.cs.sjsu.edu/~yazdankhah

Mathematical Preliminaries

(Part 2)

Lecture 03

Day 03/31

CS 154

Formal Languages and Computability

Fall 2019

Agenda of Day 03

- Waiting List Enrollment ...
- Announcement
- Summary of Lecture 02
- Lecture 03: Teaching ...
  - Covering one slide from the past
  - Mathematical Preliminaries (Part 2)

Announcement

- Our first quiz will be next Thursday!
- Some of the questions are true/false and multiple choice.
- So, please have Scantron 882 E.
- If you forget, no problem at all! I'll sell it at:

$~~20~~ Each

Now Just $19.99!

Summary of Lecture 02: We learned ...

Sets

- A set is ...
  - ... a collection of objects.
- A list is ...
  - ... a collection of ordered objects.
- A set is known when its boundary is clearly defined.
- Three methods to represent sets ...
  - Roster method
  - Venn diagram
  - Set builder
- Universal set of a set is ...
  - ... the set containing all possible elements under consideration.

- The power set of the set S is ...
  - ... the set of all subsets of S.
  - It is denoted by 2^S.
  - $|2^S| = 2^{|S|}$
- A set is called finite if ...
  - ... its size is a natural number.
- A set is called infinite if ...
  - ... we cannot express its size by a natural number.

Any question?

Empty Set Representation by Set Builder

- How to represent empty set by set builder?
- We know: $A - B = \{x : x \in A \text{ AND } x \notin B\}$
- Substitute A for B: $A - A = \{x : x \in A \text{ AND } x \notin A\}$
- $\therefore \quad \phi = \{x : \text{False}\}$

- So, to represent empty set, just put any false statement in the description part of the set builder.

- For example, the following sets represent empty sets:
- $\{x : x$ is the 8^{th} day of week$\}$
- $\{x : x \notin U\}$

Mathematical Preliminaries

Recap from Math 42

Cartesian Products

Motivation

- Recall that in sets, order of elements does NOT matter.
- But in practice, we do need ordered collections.
- As we said before, in computer science we use "Lists" for ordered collections.
- The question is how we can mathematically model lists?

Introduction

- Mathematicians defined a new mathematical structure called "n-tuple".
- It is denoted by $(a_1, a_2, \ldots, a_n)$ where a_i's are objects.
  - A special case of n-tuple is 2-tuple aka ordered-pair (a_1, a_2).
- We use a mathematical operation called Cartesian product to create n-tuples.
- This operation is named after the great French philosopher, mathematician, and physicist René Descartes (1596-1650).

Cartesian Products

Definition

- Let A and B be two sets.
- The Cartesian product of A and B is the set of all ordered-pairs (a , b), where $a \in A$ and $b \in B$.
- Cartesian product of A and B is denoted by A x B.

Set Builder Definition

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

- Note that (a , b) is the pattern of the elements.

Cartesian Products: Examples

Example 25

- Let $A = \{0, 1\}$, $B = \{3, 6, 9\}$; $A \times B = ?$

Solution

- $\{0, 1\} \times \{3, 6, 9\} = \{(0, 3), (0, 6), (0, 9), (1, 3), (1, 6), (1, 9)\}$

Example 26

- Let $Q = \{q_0, q_1\}$, $\Sigma = \{a , b\}$; $Q \times (\Sigma \cup \{\lambda\}) = ?$
  - "λ" is pronounced "lambda".

Solution

Cartesian Products: Properties

- What is the result of the following Cartesian product?

  $A = \{1, 2\}$, $B = \phi$; $A \times B = ?$

  $A \times B = \phi$

  – In fact, the result of Cartesian product would be ϕ if one of the sets is ϕ.
- How can you prove it?

- Does Cartesian product have commutative property?
- Is this a true statement: $A \times B = B \times A$
- In general, No! $A \times B \neq B \times A$

- But in the following special cases, they are equal:

  $A \times B = B \times A$ iff $(A = B) \vee (A = \phi) \vee (B = \phi)$

Cartesian Products Extension

- We can extend the idea to n sets to produce n-tuple:

$$S_1 \times S_2 \times \ldots \times S_n = \{(x_1, x_2, \ldots, x_n) : x_1 \in S_1, \ldots, x_n \in S_n\}$$

Homework

- Let $Q = \{q_0, q_1\}$, $\Sigma = \{a, b\}$, $\Gamma = \{x, y\}$
- $Q \times (\Sigma \cup \{\lambda\}) \times \Gamma = ?$

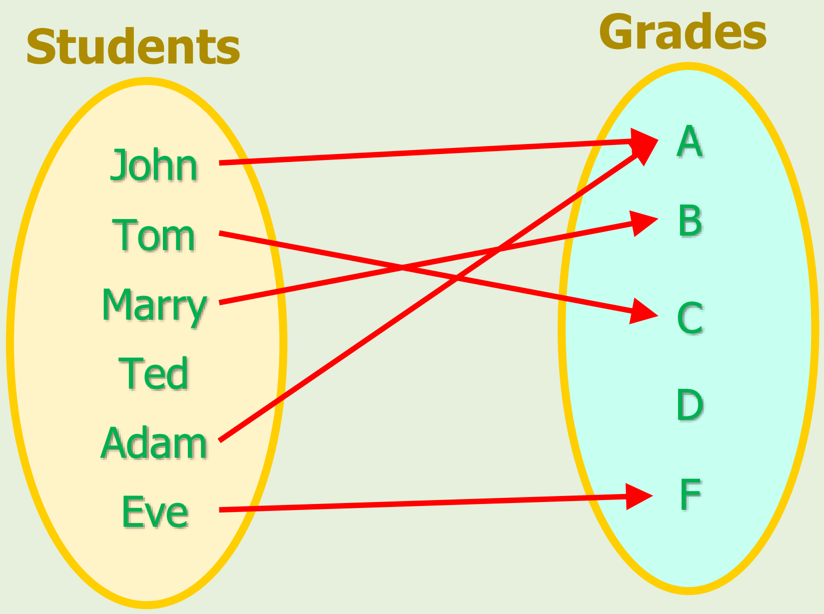
Mathematical Preliminaries

Recap from Math 42

Functions

Introduction

- In many situations in real life,
  there is a relationship between two sets.
- For example, we assign a letter grade to each student of a class.

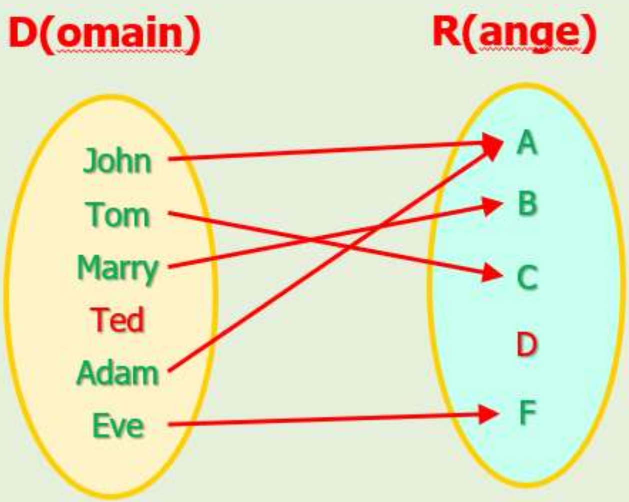


- This relationship is an example of the concept of function.

Functions

Definition

- Let D and R be two sets.
- A function g from D to R is a rule that assigns (or maps) to the elements of D a "unique element" of R.



- The set D is called the "domain" of g.
- The set R is called the "range" of g.

- The function g from D to R is denoted by: $g : D \to R$

Functions: Naming Convention

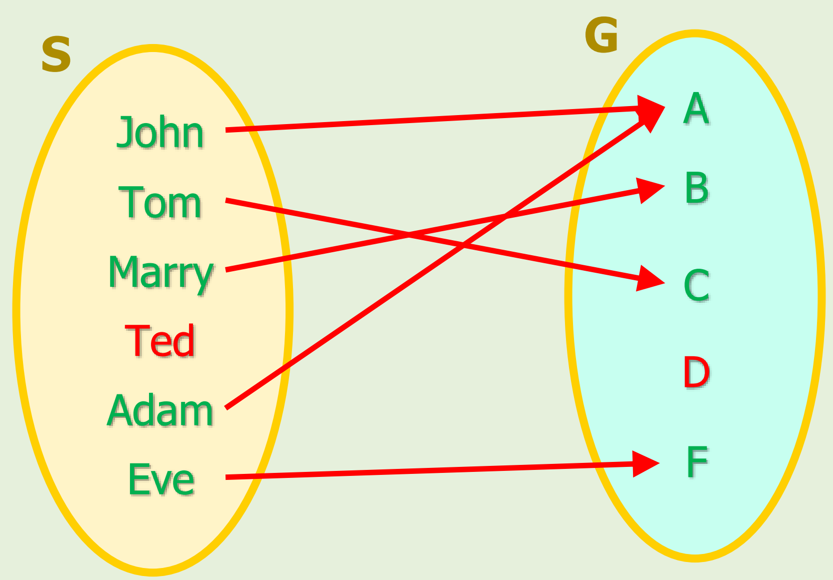
- We usually name a function by lower-case letters (English or Greek) such as f, g, h, δ (pronounce "delta"), etc.

Example 27

S = {John, Tom, Marry, Ted, Adam, Eve}, G = {A, B, C, D, F}

- We can name this function "h".

$h : S \rightarrow G$



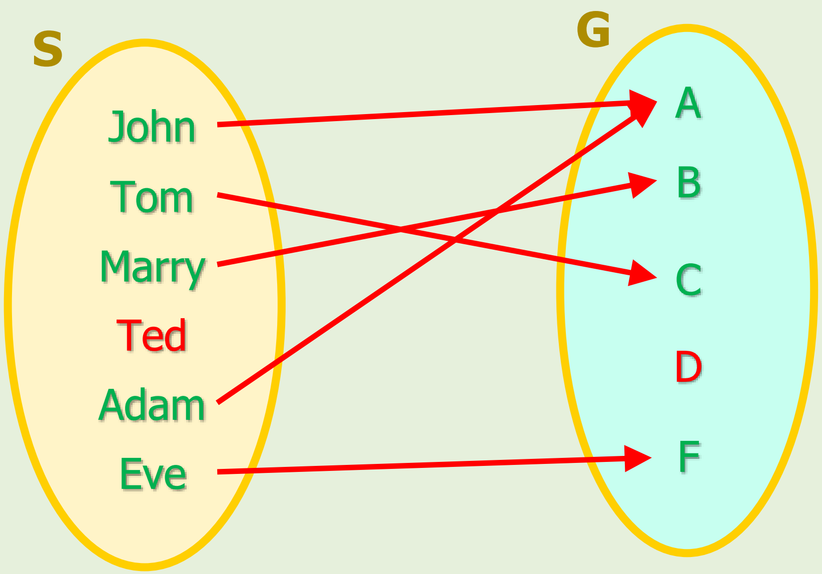
Functions: Algebraic Notation

Example 27 (cont'd)

- What is the rule of this function in algebraic notation?

  S = {John, Tom, Marry, Ted, Adam, Eve}, G = {A, B, C, D, F}

  $h : S \to G$

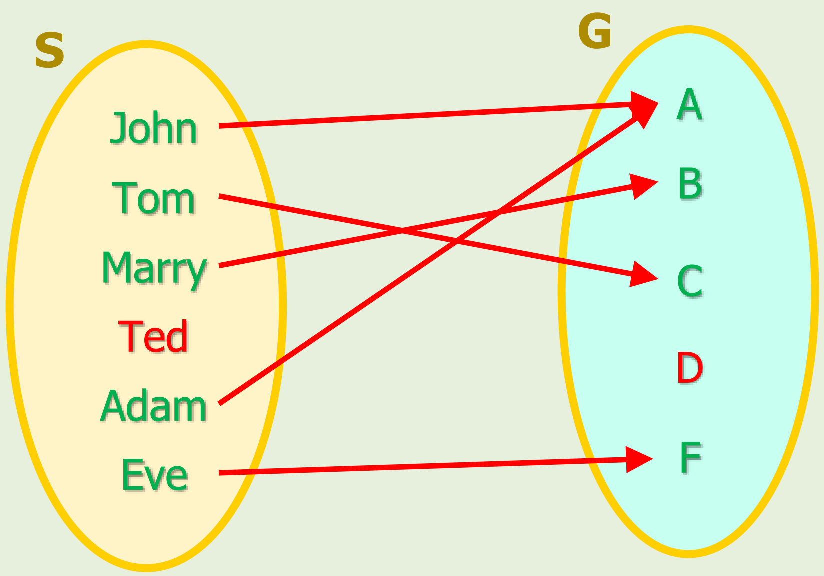


- We can translate the Venn Diagram into algebraic notation like this:

$$\begin{cases} h(\text{John}) = A \\ h(\text{Tom}) = C \\ h(\text{Marry}) = B \\ h(\text{Adam}) = A \\ h(\text{Eve}) = F \end{cases}$$

Functions: Notes

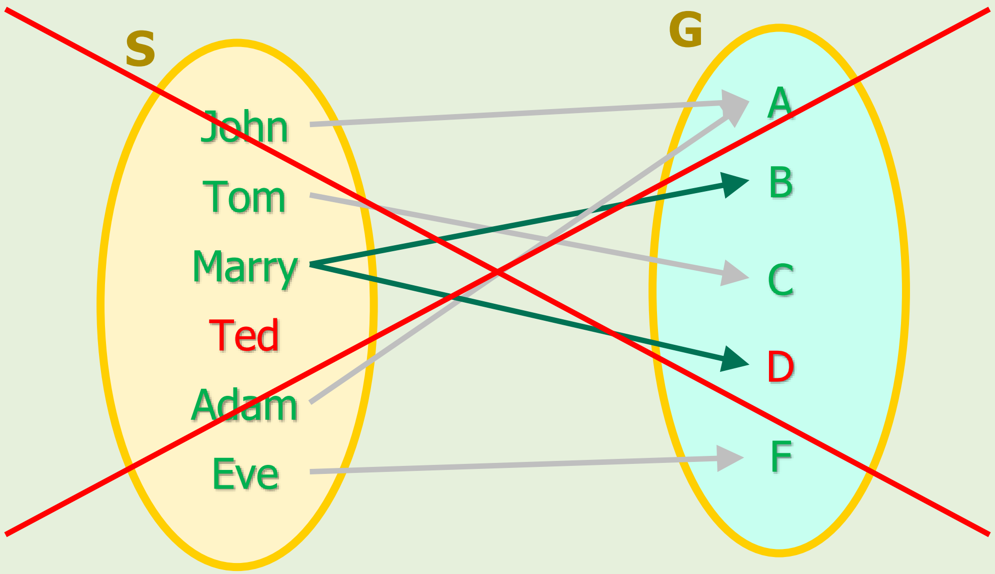
- h(Ted) = ?
- h(Ted) = Undefined



- So, it is possible to have some elements in the domain that is NOT mapped to any value of the range. (e.g. Ted in the domain)
- Also, it is possible to have some elements in the range that is NOT assigned by any value of the domain. (e.g. D in the range)

Functions: Notes

- Is it possible for Marry to have two grades at the same time?



- Absolutely, NO.

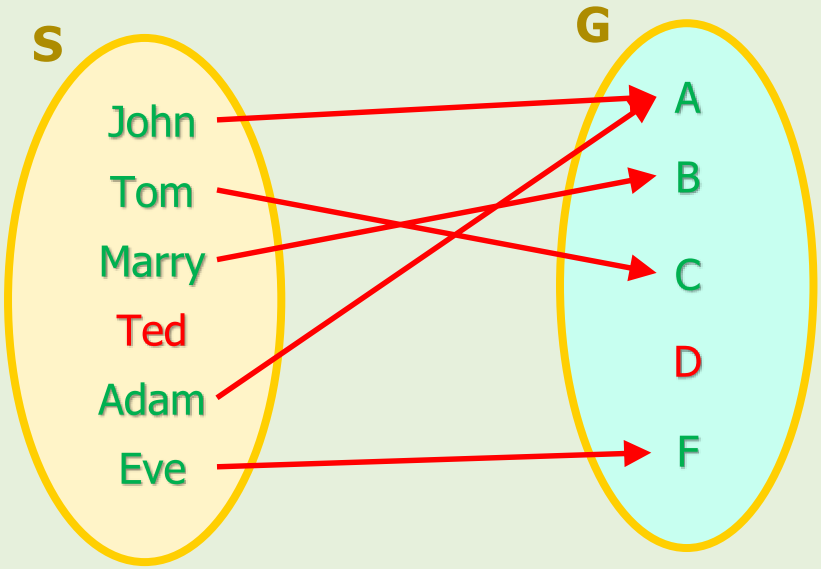
  In this universe,
  it cannot happen.

- That's why, in the definition of function, we said elements of the domain are uniquely mapped to an element of the range.
- In other words, if there is a mapping, it should be unique.

Functions as Sets

Example 27 (cont'd)

$$\begin{cases} h(\text{John}) = A \\ h(\text{Tom}) = C \\ h(\text{Marry}) = B \\ h(\text{Adam}) = A \\ h(\text{Eve}) = F \end{cases}$$



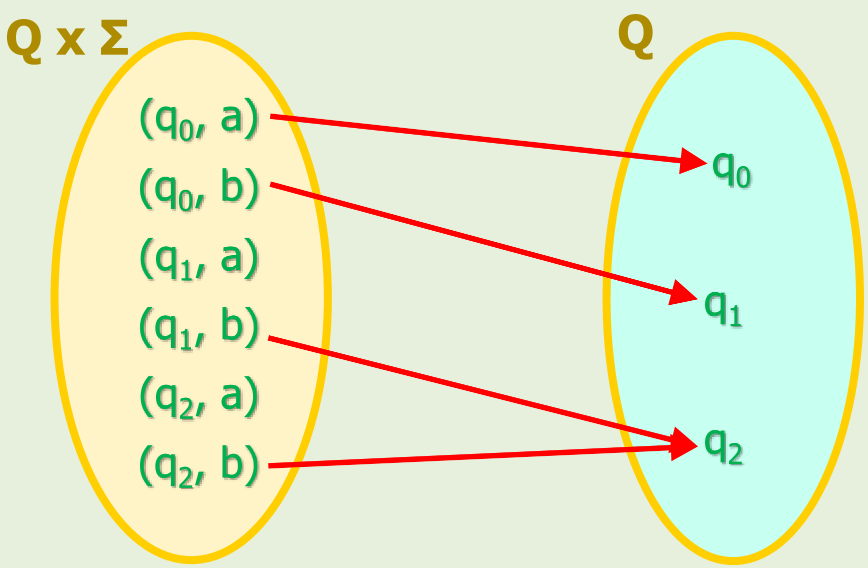
- The above relations can be represented by a set of ordered-pairs:
- H = {(John, A), (Tom, C), (Marry, B), (Adam, A), (Eve, F)}
- And the universal set for H is S x G that contains all possible relations between S's members and G's members.
- U = S x G = {(John, A), (John, B), ..., (John, F), (Tom, A), ... , (Tom, F), ..., (Ted, A), ..., (Ted, F), ..., (Eve, F)}

Functions: Example

Example 28: Mixing Cartesian Product and Function

- Let $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{a , b\}$, $\delta : Q \times \Sigma \rightarrow Q$
- What is the domain and range of δ?
- Domain: $Q \times \Sigma = \{q_0, q_1, q_2\} \times \{a , b\} =$
  $\{(q_0, a), (q_0, b), (q_1, a), (q_1, b), (q_2, a), (q_2, b)\}$
- Range: $\{q_0, q_1, q_2\}$
- The rule of δ is shown in the following Venn diagram.
- Write the rule of δ by using algebraic notation.

$$\begin{cases} \delta(q_0, a) = q_0 \\ \delta(q_0, b) = q_1 \\ \delta(q_1, b) = q_2 \\ \delta(q_2, b) = q_2 \end{cases}$$



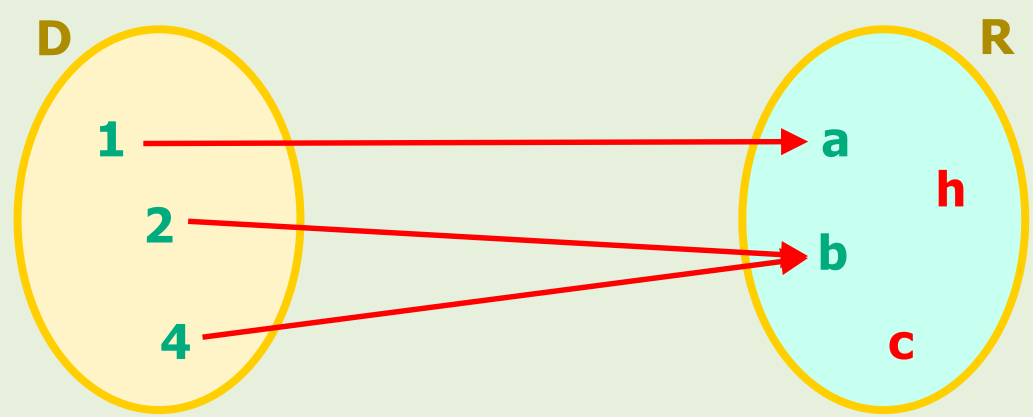
Total Function

Definition

- A function is called total if all of its domain elements are defined.

Example 29

- The following function is total because all domain elements are defined.



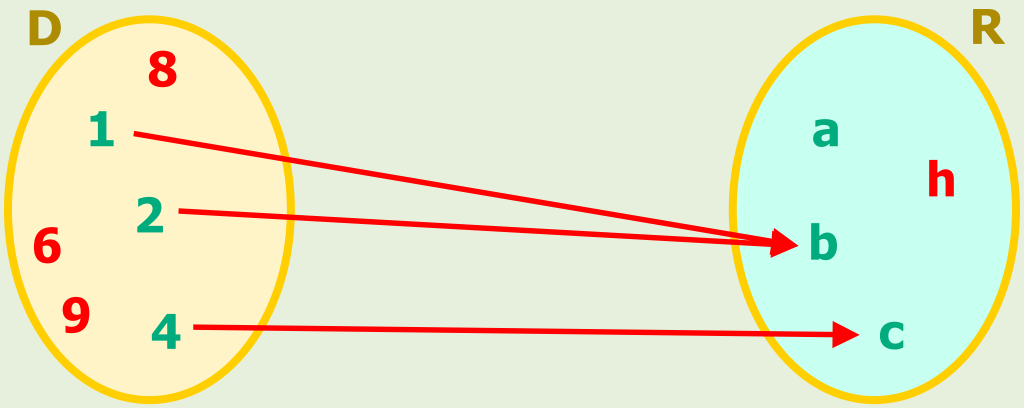
Partial Function

Definition

- A function is called partial if at least one of its domain's elements is undefined.

Example 30

- The following function is partial because f(8) = Undefined
- Note that one example is enough to make a function partial.



Functions

Homework

Functions: Homework #1

- Let $Q = \{q_0, q_1\}$, $\Sigma = \{a\}$, $\Gamma = \{x\}$.
- The function δ is defined as:

$$\delta : Q \times \{\Sigma \cup \{\lambda\}\} \times \Gamma \rightarrow Q$$

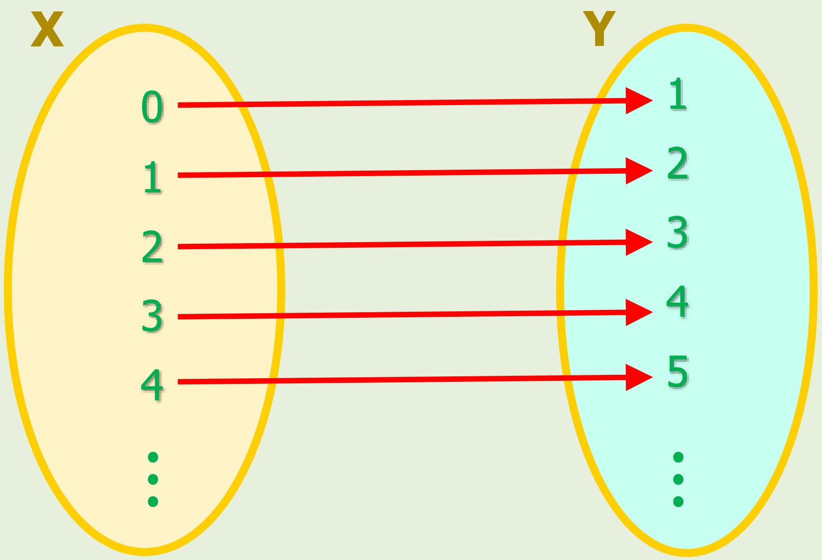
- What is the domain and range of δ?
- Represent the universal set of δ as a set of quadruples (4-tuple).

Functions: Homework #2

- Write the algebraic notation of the following function:

Solution

$$\begin{cases} f(0) = ? \\ f(1) = ? \\ f(2) = ? \\ f(3) = ? \\ \quad \cdots \end{cases}$$



- Sometimes, we can find a pattern between the elements of domain and range.
- What is the pattern of the above example?
- $f(x) = ?$

Functions: Homework #3

- Write a simple Java method to implement the function of HW #2.
- Hint: the argument of the method is the domain and whatever it's returning is range.

Solution

Functions: Homework #4

- What is the range and domain of the following Java code?
- `public int g(int x, int z) { return 2*x + z; }`

Solution

Mathematical Preliminaries

Recap from Math 42

Graphs

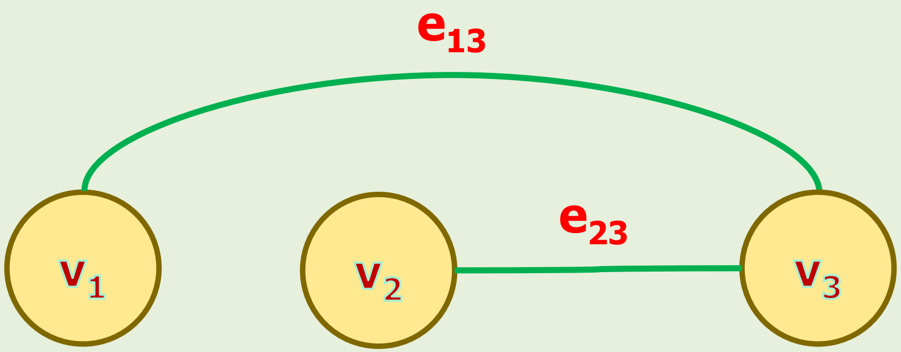
Graphs Definition

Definition

- A graph is a mathematical construct consisting of two sets:
  - A non-empty and finite set of vertices (aka nodes, states) $V = \{v_1, v_2, \ldots, v_n\}$
  - A finite set of edges $E = \{e_1, e_2, \ldots, e_m\}$
  - Each edge connects two vertices.

Example 31

- $V = \{v_1, v_2, v_3\}$
- $E = \{e_{13}, e_{23}\}$



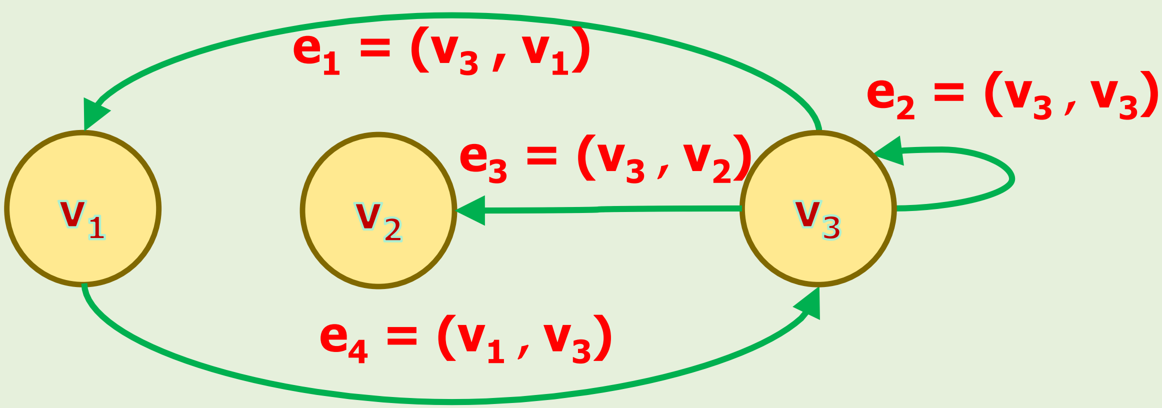
Directed Graphs

- If the direction of the edges matters,
  then we call the graph directed graph (aka digraph).
- The edges are shown by ordered-pair (start-vertex , end-vertex).
  - In this course, we only use directed graphs.

Example 32

- Draw a digraph with the following specifications:

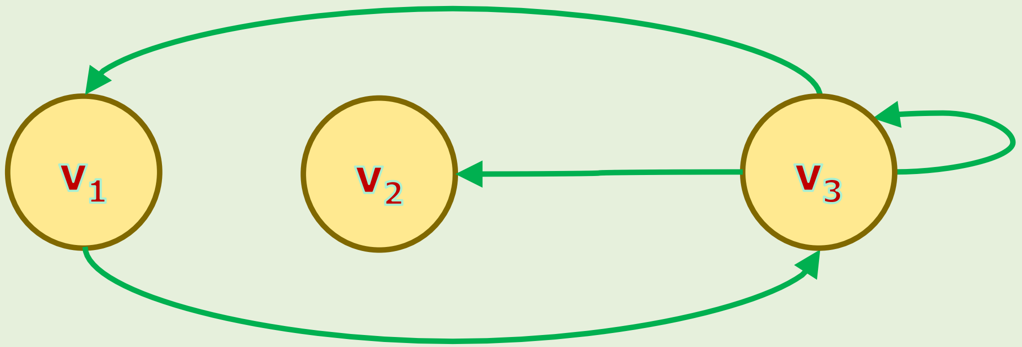
$$V = \{v_1 , v_2 , v_3\} , E = \{(v_1 , v_3), (v_3 , v_1), (v_3 , v_2), (v_3 , v_3)\}$$



Graphs Terminologies

Walk

- A sequence of edges like $(v_i, v_j), (v_j, v_k), \ldots, (v_m, v_n)$, is called a walk from v_i to v_n.
  - Note that the end vertex of e_i is the start vertex of e_{i+1}.
  - In other words, in a walk we cannot jump!

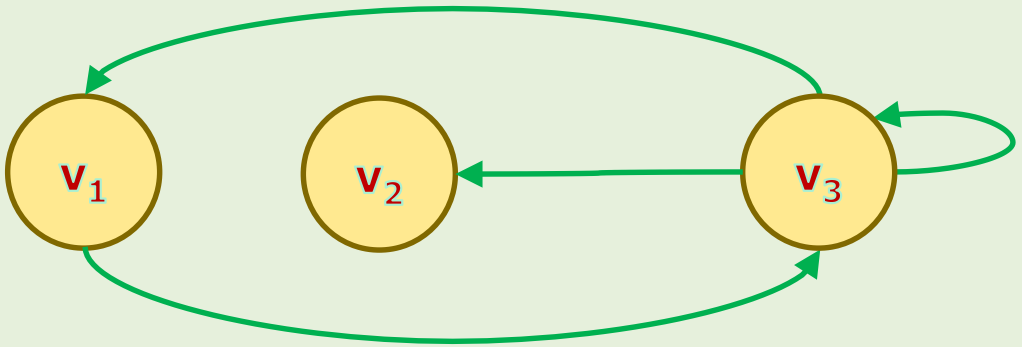


Example 33

- Each of the following sequences are a walk from v_1 to v_3:
  - Walk 1: (v_1, v_3)
  - Walk 2: $(v_1, v_3), (v_3, v_3)$
  - Walk 3: $(v_1, v_3), (v_3, v_1), (v_1, v_3)$
  - …

Length of Walks

- It is the total number of edges traversed.

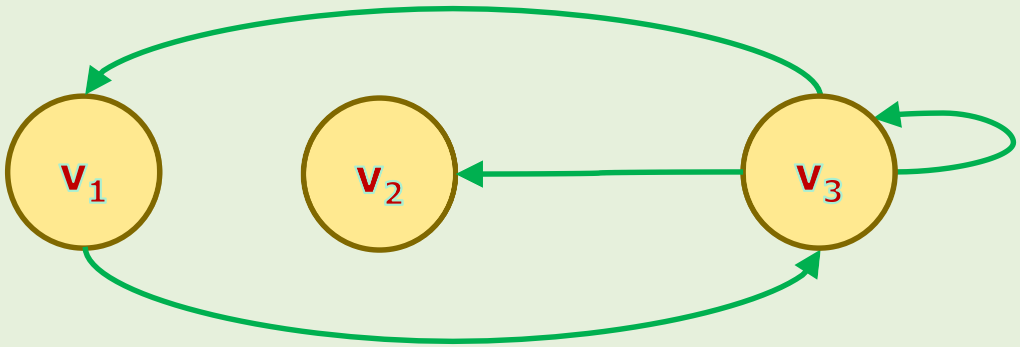


Example 33 (cont'd)

- Walk 1: (v_1, v_3) ; length = 1
- Walk 2: $(v_1, v_3), (v_3, v_3)$; length = 2
- Walk 3: $(v_1, v_3), (v_3, v_1), (v_1, v_3)$; length = 3

Path

- A walk that no edges are repeated.

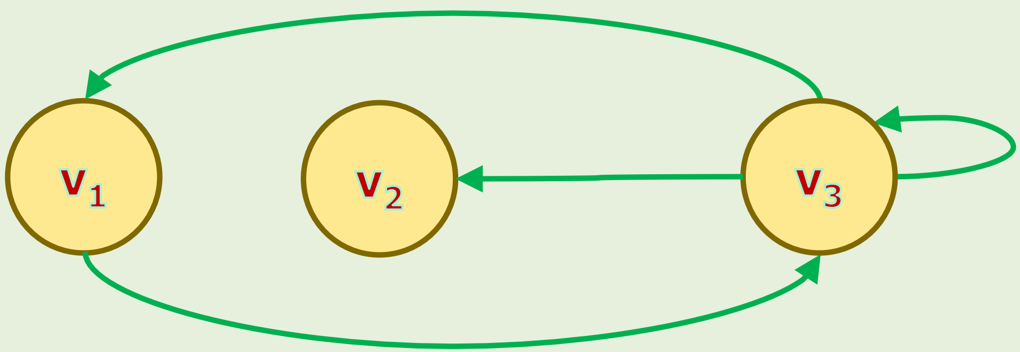


Example 34

- Which one is a path?
  - ✓ Walk 1: (v_1, v_3)
  - ✓ Walk 2: $(v_1, v_3), (v_3, v_3)$
  - ✗ Walk 3: $(v_1, v_3), (v_3, v_1), (v_1, v_3)$

Simple Path

- A path that no vertices are repeated.
  - In other words, no vertex should be visited more than once.

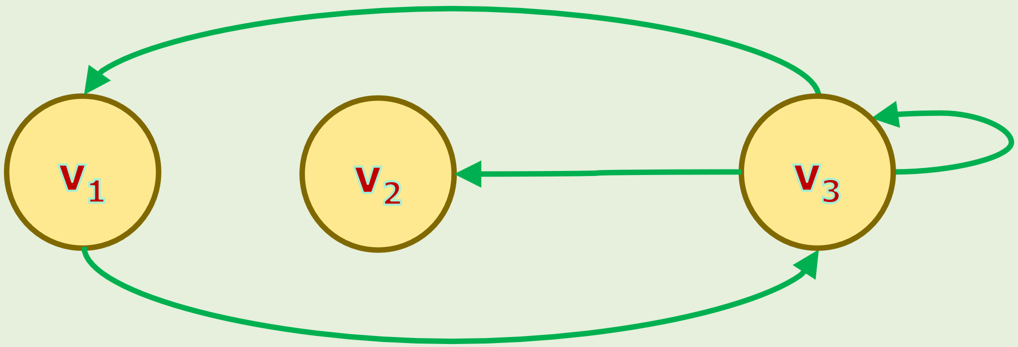


Example 35

- Which one is a simple path?
  - ✓ Walk 1: (v_1, v_3)
  - ✗ Walk 2: $(v_1, v_3), (v_3, v_3)$
  - ✗ Walk 3: $(v_1, v_3), (v_3, v_1), (v_1, v_3)$

Loop

- An edge from a vertex to itself.



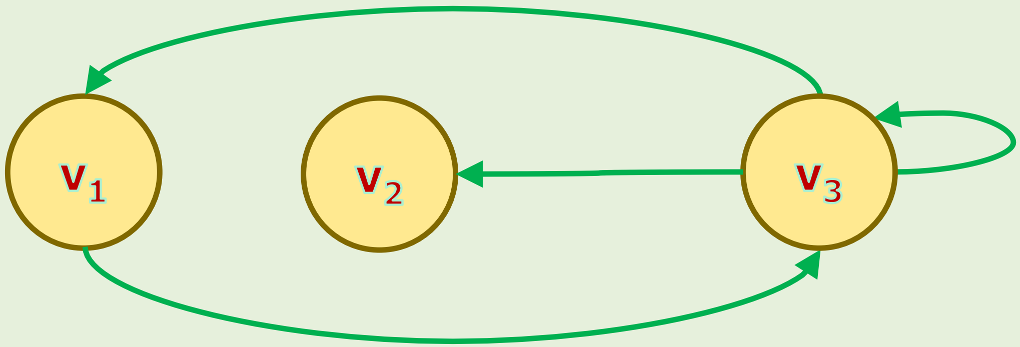
Example 36

- Which one is a loop?
  - ✓ – Walk 1: (v_3 , v_3)

- Is there any other loop in this graph?

Cycle

- A walk from a vertex (called base) to itself with no repeated edges.
- Recall that: Walk + No repeated edges = path
- Rewording: A cycle is a path from a vertex (called base) to itself.



Example 37

- Which one is a cycle?

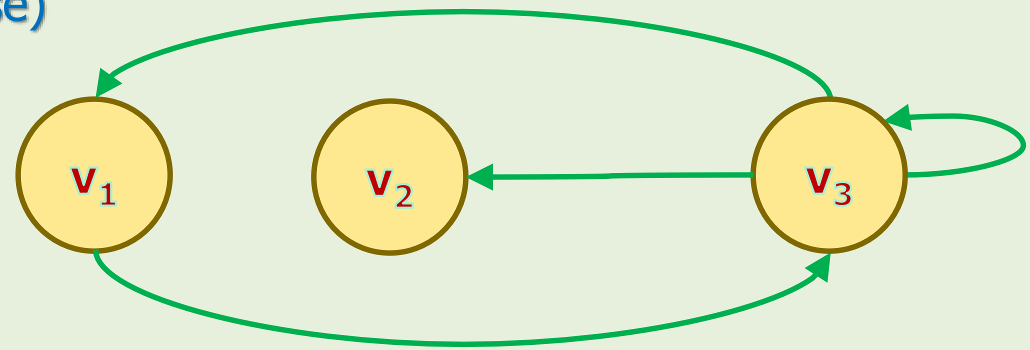
✗ – Walk 1: $(v_1, v_3), (v_3, v_1), (v_1, v_3), (v_3, v_1)$

✓ – Walk 2: $(v_1, v_3), (v_3, v_1)$

✓ – Walk 3: $(v_1, v_3), (v_3, v_3), (v_3, v_1)$

Simple Cycle

- A cycle that no vertices other than the base is repeated.
  - Note that the walk starts from the base and ends to the base.
  - During the walk, the base should not be repeated too.
  - In other words, in a simple cycle, all vertices (except the base) and the edges are visited uniquely.



Example 38

- Which one is a simple cycle?

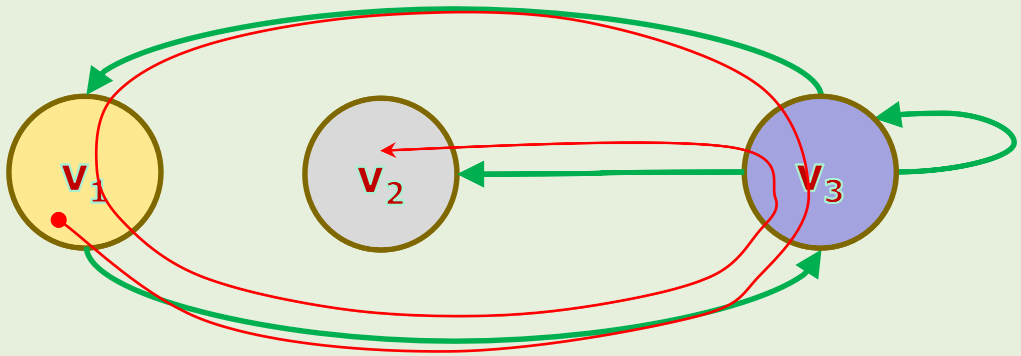
✗ – Walk 1: $(v_1 , v_3) , (v_3 , v_1) , (v_1 , v_3) , (v_3 , v_1)$

✓ – Walk 2: $(v_1 , v_3) , (v_3 , v_1)$

✗ – Walk 3: $(v_1 , v_3) , (v_3 , v_3) , (v_3 , v_1)$

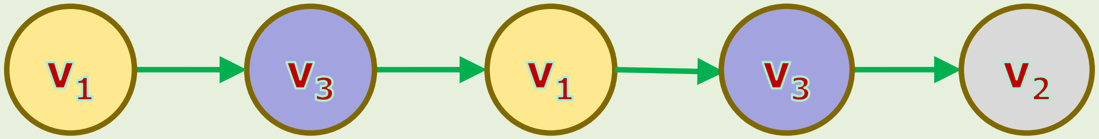
One-Dimensional Projection of a Walk

- One-dimensional projection (or just projection) is another way of representing a walk.



Example 39

- Represent the following walk as a one-dimensional projection.
- Walk from v_1 to v_2: $(v_1 , v_3) , (v_3 , v_1) , (v_1 , v_3) , (v_3 , v_2)$



- The length of this walk (= the number of edges) is clearly shown.

References

1. Linz, Peter, "An Introduction to Formal Languages and Automata, 5th ed.," Jones & Bartlett Learning, LLC, Canada, 2012
2. Kenneth H. Rosen, "Discrete Mathematics and Its Applications, 7th ed.," McGraw Hill, New York, United States, 2012
3. Sipser, Michael, "Introduction to the Theory of Computation, 3rd ed.," CENGAGE Learning, United States, 2013
   ISBN-13: 978-1133187790