CS-1005
DISCRETE STRUCTURES

INTRODUCTION

Class code:b7xpd4s

Instructor: Dr Khadija Farooq

BOOKS

Textbook:

- Discrete Mathematics with applications (5th edition) by Susanna S. Epp

Reference book:

- Practical Discrete Mathematics by Ryan T. White, Archana Tikayat Ray Released February 2021 Publisher(s): Packt Publishing
  ISBN: 9781838983147
- https://github.com/PacktPublishing/Practical-Discrete-Mathematics
- Discrete Mathematics and its applications by Kenneth H. Rosen

Class code:b7xpd4s

ASSESSMENTS

Grading: Absolute grading

Assessment Item	Number	Weight (%)
Assignments	4	10
Quizzes	5	15
Midterm Exam	1	20
Project	1	15
Final Exam	1	40

Class code:b7xpd4s

Missed Assessments

- Retake of missed assessment items (other than midterm/ final exam) will not be held.
- For a missed midterm/ final exam, an exam retake/ pretake application along with necessary evidence are required to be submitted to the department secretary. The examination assessment and retake committee decides the exam retake/ pretake cases.

Plagiarism

- Plagiarism in project or midterm/ final exam may result in F grade in the course.
- Plagiarism in an assignment will result in zero marks in the whole assignments category.

Discrete Mathematics

- Is the language of computer science
- No much mathematical background, just logical reasonings
- Is the collection for different fields of mathematics combined to make a compact course for computer scientists. These includes Logic, Set Theory, Graph Theory, Number Theory, Probability etc.
- Discrete mathematics is the study of countable, distinct, or separate mathematical structures

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

There are two men. One of them is wearing a red shirt, and the other is wearing a blue shirt. The two men are named Andrew and Bob, but we do not know which is Andrew and which is Bob.

The guy in the blue shirt says, “I am Andrew.”
The guy in the red shirt says, “I am Bob.”

If we know that at least one of them lied, then what color shirt is Andrew wearing?

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

There are two men. One of them is wearing a red shirt, and the other is wearing a blue shirt. The two men are named Andrew and Bob, but we do not know which is Andrew and which is Bob.

The guy in the blue shirt says, "I am Andrew."
The guy in the red shirt says, "I am Bob."

If we know that at least one of them lied, then what color shirt is Andrew wearing?

1) One of them is Bob and the other one is Andrew

At least one of them is lying

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

There are two men. One of them is wearing a red shirt, and the other is wearing a blue shirt. The two men are named Andrew and Bob, but we do not know which is Andrew and which is Bob.

The guy in the blue shirt says, "I am Andrew."
The guy in the red shirt says, "I am Bob."

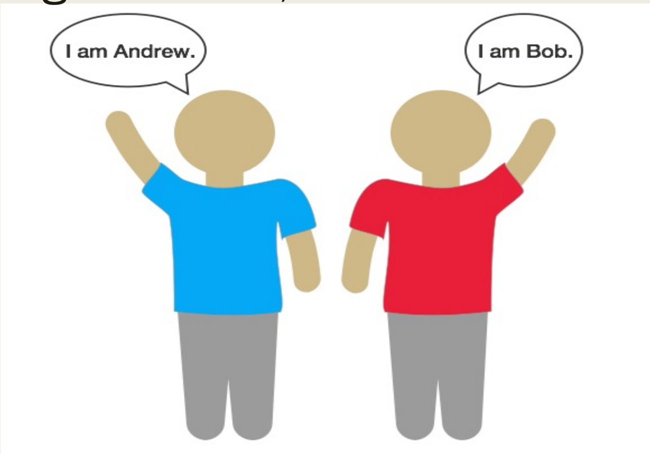
If we know that at least one of them lied, then what color shirt is Andrew wearing?

1) One of them is Bob and the other one is Andrew
2) At least one of them is lying

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.



There are 4 cases to consider depending upon who lied or spoke the truth:

Case 1: *Both spoke the truth.* Since one of them lied, this is not possible.

Case 2: *Red shirt told the truth and Blue shirt lied.* If only the person in the blue shirt lied, then he would be Bob, and the person in the red shirt would be Bob. Since neither of them are Andrew, this is not possible.

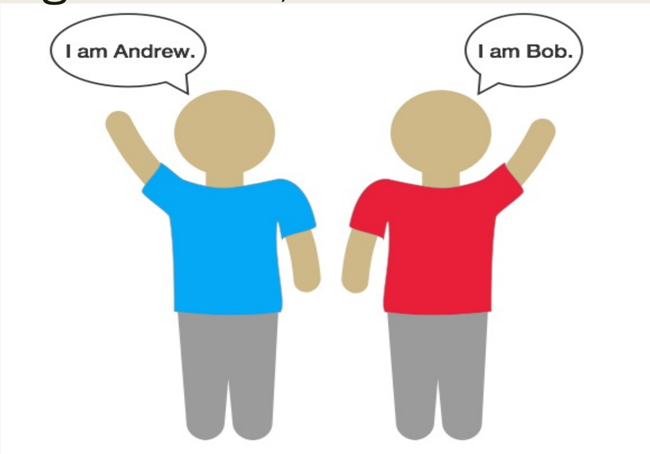
Case 3: *Blue shirt spoke the truth and Red lied.* If only the person in the red shirt lied, then he would be Andrew, and the person in the blue shirt would be Andrew. Since neither of them are Bob, this is not possible.

Case 4: *Both of them lied.* Hence the person in blue shirt is Bob and the person in red shirt is Andrew.

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.



There are 4 cases to consider depending upon who lied or spoke the truth:

Case 1: *Both spoke the truth.* Since one of them lied, this is not possible.

Case 2: *Red shirt told the truth and Blue shirt lied.* If only the person in the blue shirt lied, then he would be Bob, and the person in the red shirt would be Bob. Since neither of them are Andrew, this is not possible.

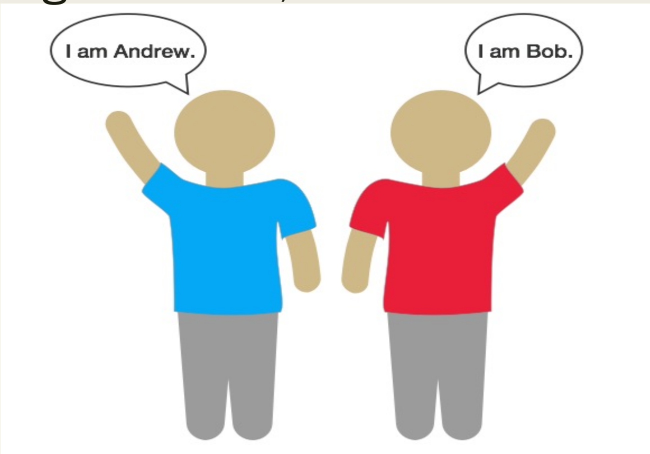
Case 3: *Blue shirt spoke the truth and Red lied.* If only the person in the red shirt lied, then he would be Andrew, and the person in the blue shirt would be Andrew. Since neither of them are Bob, this is not possible.

Case 4: *Both of them lied.* Hence the person in blue shirt is Bob and the person in red shirt is Andrew.

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.



There are 4 cases to consider depending upon who lied or spoke the truth:

Case 1: *Both spoke the truth.* Since one of them lied, this is not possible.

Case 2: *Red shirt told the truth and Blue shirt lied.* If only the person in the blue shirt lied, then he would be Bob, and the person in the red shirt would be Bob. Since neither of them are Andrew, this is not possible.

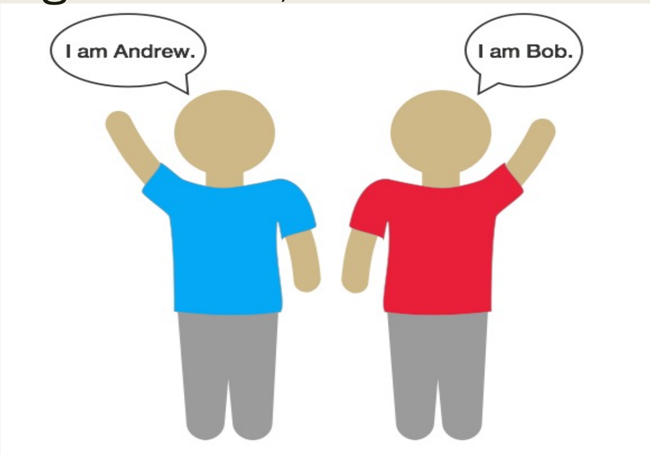
Case 3: *Blue shirt spoke the truth and Red lied.* If only the person in the red shirt lied, then he would be Andrew, and the person in the blue shirt would be Andrew. Since neither of them are Bob, this is not possible.

Case 4: *Both of them lied.* Hence the person in blue shirt is Bob and the person in red shirt is Andrew.

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.



There are 4 cases to consider depending upon who lied or spoke the truth:

Case 1: *Both spoke the truth.* Since one of them lied, this is not possible.

Case 2: *Red shirt told the truth and Blue shirt lied.* If only the person in the blue shirt lied, then he would be Bob, and the person in the red shirt would be Bob. Since neither of them are Andrew, this is not possible.

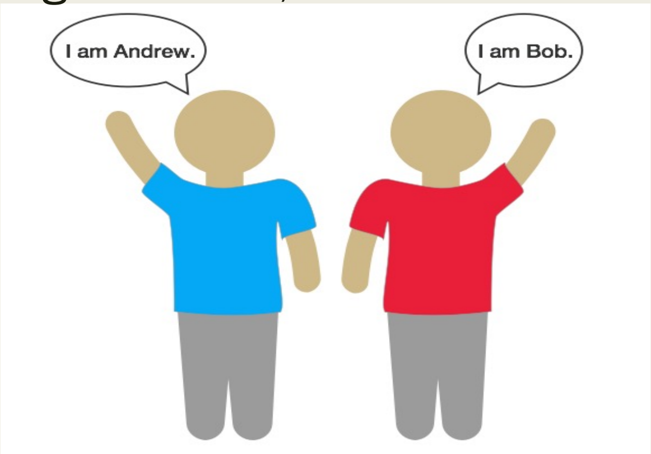
Case 3: *Blue shirt spoke the truth and Red lied.* If only the person in the red shirt lied, then he would be Andrew, and the person in the blue shirt would be Andrew. Since neither of them are Bob, this is not possible.

Case 4: *Both of them lied.* Hence the person in blue shirt is Bob and the person in red shirt is Andrew.

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.



There are 4 cases to consider depending upon who lied or spoke the truth:

Case 1: *Both spoke the truth.* Since one of them lied, this is not possible.

Case 2: *Red shirt told the truth and Blue shirt lied.* If only the person in the blue shirt lied, then he would be Bob, and the person in the red shirt would be Bob. Since neither of them are Andrew, this is not possible.

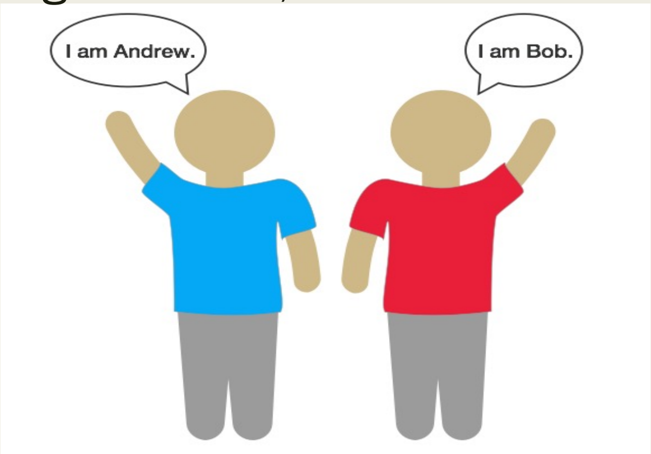
Case 3: *Blue shirt spoke the truth and Red lied.* If only the person in the red shirt lied, then he would be Andrew, and the person in the blue shirt would be Andrew. Since neither of them are Bob, this is not possible.

Case 4: *Both of them lied.* Hence the person in blue shirt is Bob and the person in red shirt is Andrew.

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.



The method of exhaustion

There are 4 cases to consider depending upon who lied or spoke the truth:

Case 1: *Both spoke the truth.* Since one of them lied, this is not possible.

Case 2: *Red shirt told the truth and Blue shirt lied.* If only the person in the blue shirt lied, then he would be Bob, and the person in the red shirt would be Bob. Since neither of them are Andrew, this is not possible.

Case 3: *Blue shirt spoke the truth and Red lied.* If only the person in the red shirt lied, then he would be Andrew, and the person in the blue shirt would be Andrew. Since neither of them are Bob, this is not possible.

Case 4: *Both of them lied.* Hence the person in blue shirt is Bob and the person in red shirt is Andrew.

Goals of the Discrete Structures Course

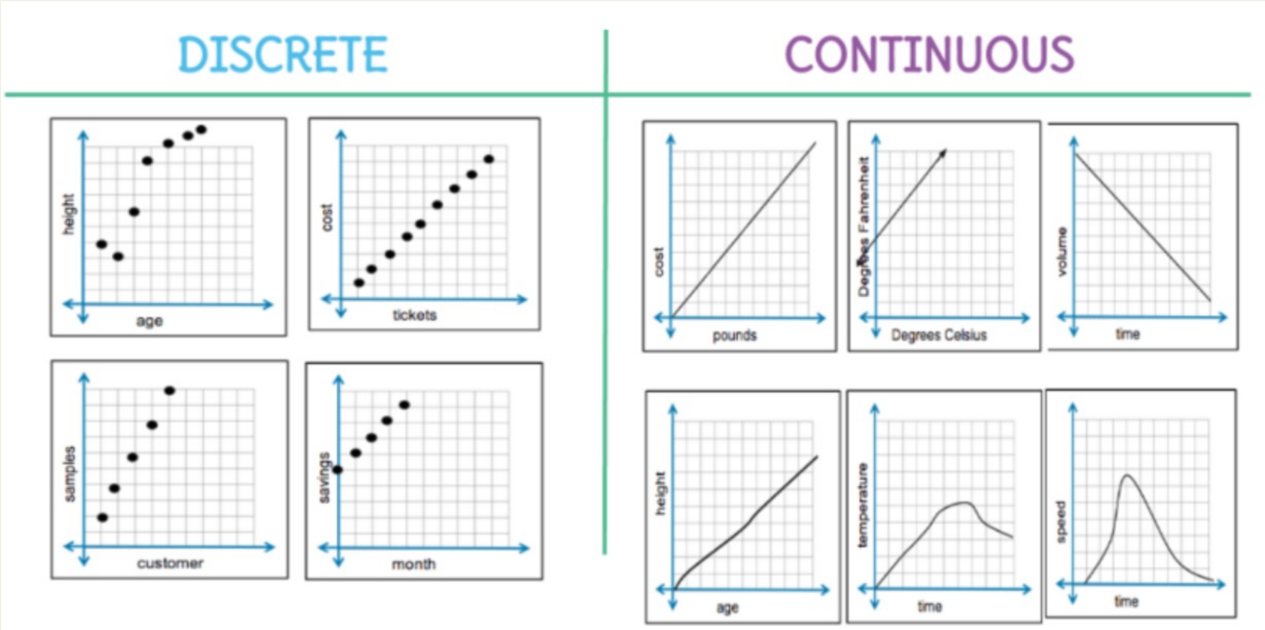
- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

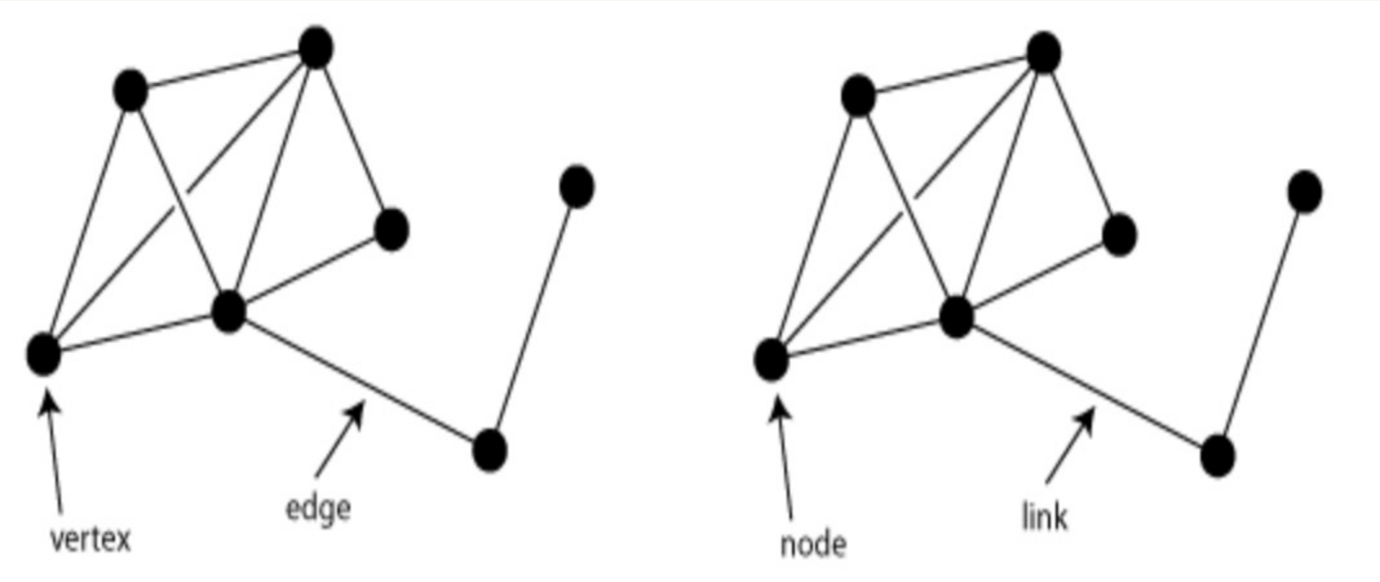
- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.



Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

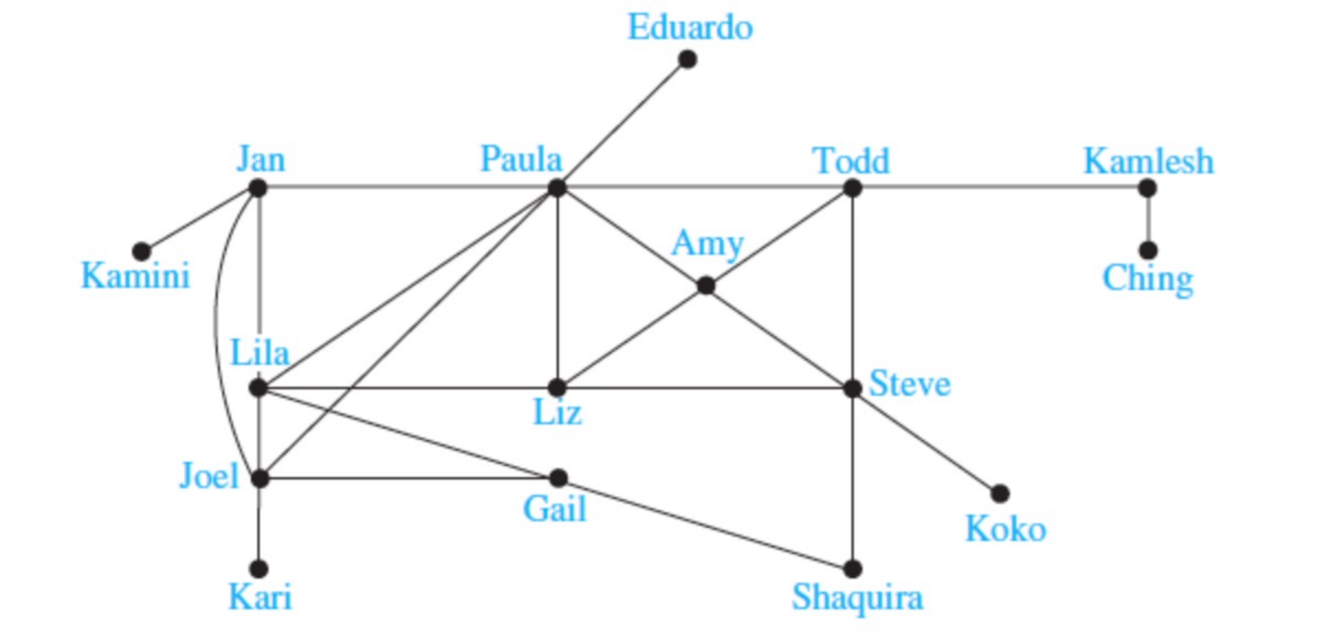


Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

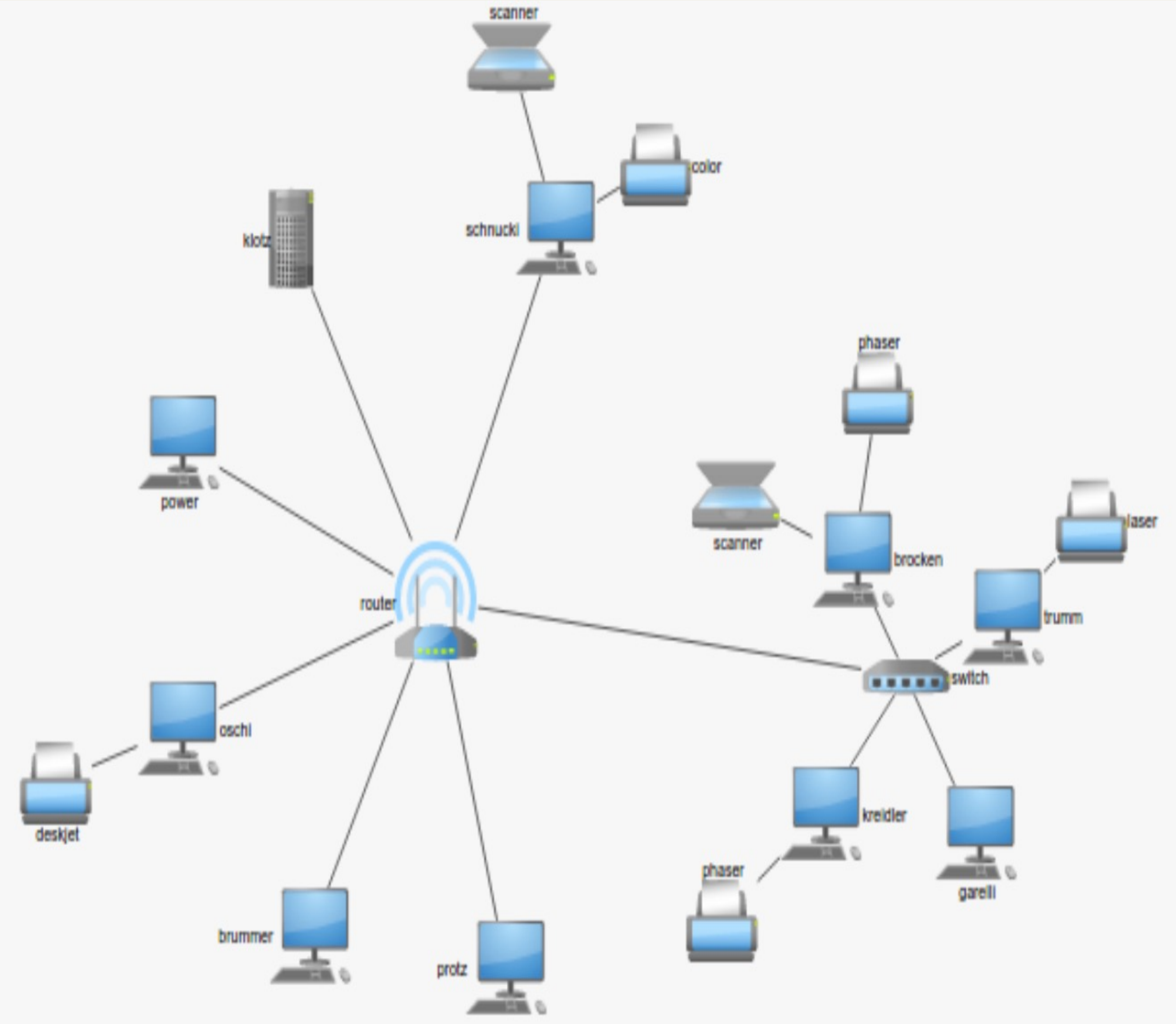
- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.



Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.



Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Any amount greater or equal to 8cents can be obtained using 3cents and 5cents coins.

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Any amount greater or equal to 8cents can be obtained using 3cents and 5cents coins.

$$8 \text{ cents} = 3 + \boxed{5}$$

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Any amount greater or equal to 8cents can be obtained using 3cents and 5cents coins.

8 cents = 3 + $\boxed{5}$

9 cents =

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Any amount greater or equal to 8cents can be obtained using 3cents and 5cents coins.

8 cents = 3 + $\boxed{5}$

9 cents = 3 +

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Any amount greater or equal to 8cents can be obtained using 3cents and 5cents coins.

8 cents = 3 + $\boxed{5}$

9 cents = 3 + 3 + 3

Class code:b7xpd4s

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Any amount greater or equal to 8cents can be obtained using 3cents and 5cents coins.

8 cents = 3 + 5

9 cents = 3 + 3 + 3

10 cents =

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

Any amount greater or equal to 8cents can be obtained using 3cents and 5cents coins.

8 cents = 3 + 5

9 cents = 3 + 3 + 3

10 cents = 5 + 5

Goals of the Discrete Structures Course

- **Mathematical Reasoning:**
  This will help students to think abstractly. This means learning to use logically valid forms of argument and avoid common logical errors, derive new results form already known to be true.

- **Discrete Structures:**
  Discrete mathematical structures are the abstract structures that describe, categorize, and reveal the underlying relationships among discrete mathematical objects. Those studied in this course are the sets of integers and rational numbers, general sets, Boolean algebras, functions, relations, graphs and trees.

- **Induction and Recursion:**
  In the analysis of algorithms when a recursive relation give rise to some formulas then these formulas are then verified in mathematical induction.

- **Algorithms and Their Analysis:**
  To solve a problem on a computer, it is necessary to find an algorithm or step-by-step sequence of instructions for the computer to follow. Designing an algorithm requires an understanding of the mathematics underlying the problem to be solved. Determining whether an algorithm is correct requires a sophisticated use of mathematical induction.

Class code:b7xpd4s

By the end of this course we will be able to:

- Think mathematically.
- Express a logic sentence in terms of predicates, quantifiers and logical connectives.
- Apply rules of logic in computational problems.
- Apply rules of inference and methods of proof to prove or disprove a mathematical statement.
- Use mathematical induction to prove properties of sequences.
- Define and solve recursive relations.
- Determine correctness of algorithms.
- Use number theory to discuss some methods of cryptography.
- Use graph theory to solve computational problems.

Why do we need this?

- In programming we need to give every single details to the computer. So we need to think mathematically and covert our statements into logical expressions.
- After developing an algorithm, we need to prove that the algorithm is correct; that is, it terminates after a finite number of loops and that it gives us our required output. For that we need proofs.

Proof

- Proof exists beyond mathematics. It is a method of verifying the truth. For example, experimentation and observation, examining the evidences.
- A mathematical proof is an inferential argument for a mathematical statement, showing that the stated assumptions logically guarantee the conclusion.

Uses of Discrete Mathematics in Computer Science

- Advanced algorithms and data structures
- Programming language compilers and interpreters
- Computer networks
- Operating systems
- Computer architecture
- Database management systems
- Cryptography
- Error correction codes, etc.

ELEMENTARY SET THEORY

"A set is a Many that allows itself to be thought of as a One." – Georg Cantor

Definition

- **Set:** A set is a collection of objects. If a set A is made up of objects $a_1, a_2, \dots,$ we write it as
$$A = \{a_1, a_2, \dots\}.$$

- **Elements of sets:** Each object in a set A is called an element of A, and we write
$$a_n \in A.$$

- **Empty set:** The empty set is denoted ϕ.

EXAMPLE

- The set of prime numbers less than 10 is $A = \{2, 3, 5, 7\}$.
- The set of the three largest cities in the world is *{Tokyo, Delhi, Shanghai}*.
- The natural numbers are a set $N = \{1, 2, 3, \ldots\}$.
- The integers are a set $Z = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$.
- If B, C, and D are sets, $A = \{B, C, D\}$ is a set of sets.
- The real numbers are written $R = (-\infty, \infty)$, which consists of the entire number line. Note that it is not possible to list the real numbers within braces, as we can with N or Z.

SUBSETS AND SUPERSETS

- A set A is a **subset** of B if all elements in A are also in B, and we write it as $A \subseteq B$.
- We call B a superset of A.
- If A is a subset of B, but not the same set, we call A a proper subset of B, and write $A \subset B$.

Set-builder notation

- A set may be written as $\{x \in A \mid Conditions\}$, which consists of the subset of A such that the given conditions are true.

- Sometimes, sets will be expressed as $\{x \mid Conditions\}$ when it is obvious what kind of mathematical object x is from the context.

Using set-builder notation

Examples of sets constructed by set-builder notation include the following.

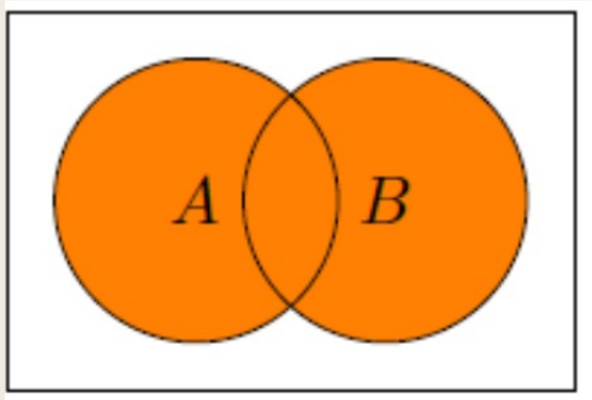
- The set of even natural numbers is
  $$\{2, 4, 6, \dots\} = \{n \mid n = 2k \; for \; some \; k \in N\}.$$
  This is an infinite set where each element n is $2 * k$, where k is some natural number belonging to the set $\{1, 2, 3 \dots\}$.
- The closed interval of real numbers from *a* to *b* is $\{x \in R \mid a \leq x \leq b\} = [a, b]$.
- The open interval of real numbers from *a* to *b* is $\{x \in R \mid a < x < b\} = (a, b)$.
- The set $R^2 = \{(x, y) \mid x, y \in R\}$ consists of the entire 2D coordinate plane.
- The line with slope 2 and y-intercept 3 is the set $\{(x, y) \in R^2 \mid y = 2x + 3\}$.
- The open ball of radius r and center $(0, 0)$ is $\{(x, y) \in R^2 \mid x^2 + y^2 < r^2\}$, which is the interior, but not the boundary of a circle.
- A circle of radius r and center $(0, 0)$ is $\{(x, y) \in R^2 \mid x^2 + y^2 = r^2\}$, which is the boundary of the circle.
- The set of all African nations is $\{x \in Nations \mid x \; is \; in \; Africa\}$.

Basic set operations

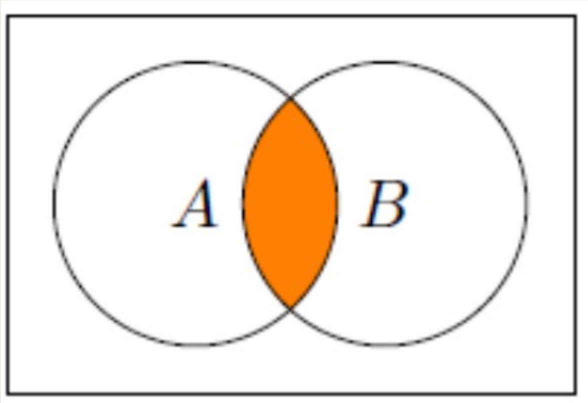
Let A and B be sets. Then:

- The **union** of sets A and B is the set of all elements in A or B (or both) and is denoted
$$A \cup B = \{x \mid x \in A \text{ } or \text{ } x \in B\}.$$
- A **union** of sets $A_1, A_2, \ldots$ is denoted
$$\bigcup_{n=1}^{\infty} A_n .$$
- The **intersection** of sets A and B is the set of all elements in both A and B. It is
$$A \cap B = \{x \mid x \in A \text{ } and \text{ } x \in B\}.$$
- An **intersection** of sets $A_1, A_2, \ldots$ is denoted
$$\bigcap_{n=1}^{\infty} A_n .$$
- The **complement** of set A is all elements in the set that are not in A and is denoted
$$A^c = \{x \mid x \notin A\}.$$
- The **difference** between sets A and B is the set of all elements in A, but not B, denoted
$$A - B = \{x \in A \mid x \notin B\}.$$

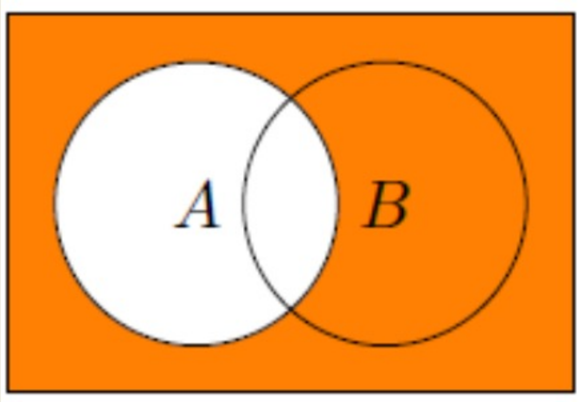
Venn diagrams



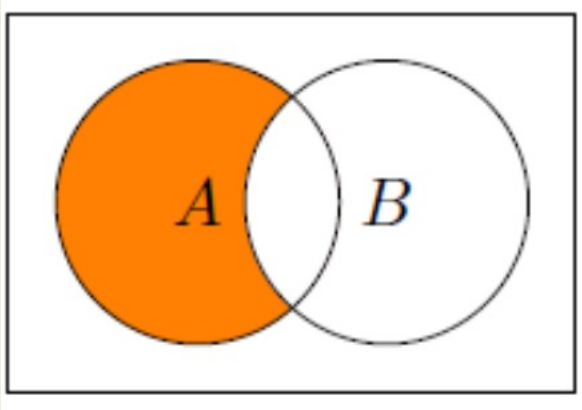
$A \cup B$



$A \cap B$



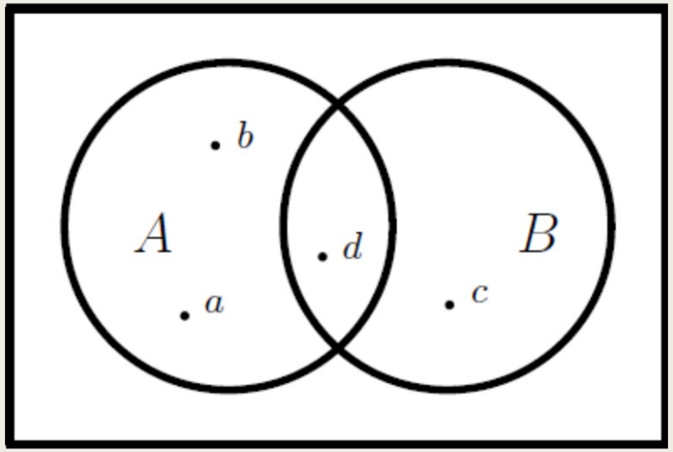
A^c



$A - B$

Class code:b7xpd4s

Example



- Elements a, b, and d are in set A, which we can write as $a, b, d \in A$.
- Elements c and d are in set B, and $c, d \in B$.
- Element c is not in A, so we could write $c \notin A$ or $c \in A^c$.
- Element d is in both A and B, or $d \in A \cap B$.
- All four elements are in A or B (or both), so we could say $a, b, c, d \in A \cup B$.

Disjoint Set

Sets *A* and *B* are disjoint (or mutually exclusive) if both sets share no elements in common. In other words,

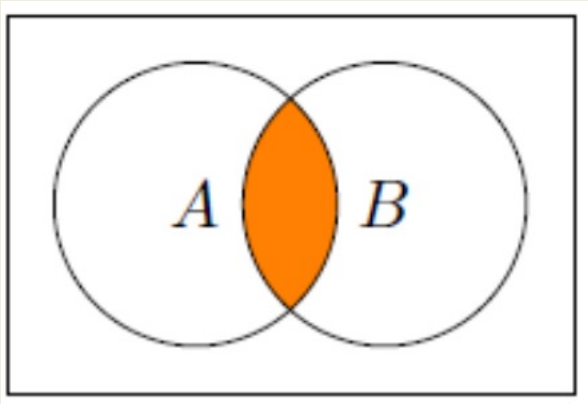
$$A \cap B = \phi$$

De Morgan’s Laws

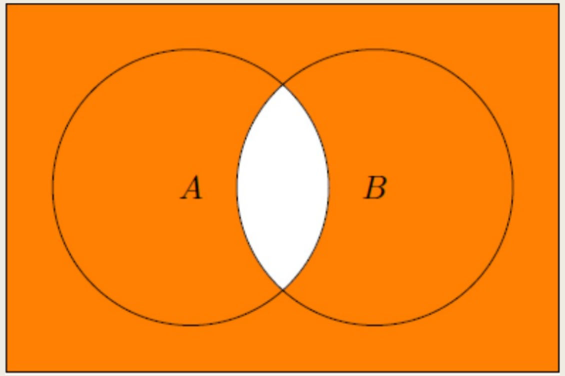
De Morgan's laws state how mathematical concepts are related through their opposites. In set theory, these laws make use of complements to address the intersection and union of sets.

De Morgan's laws can be written as follows:

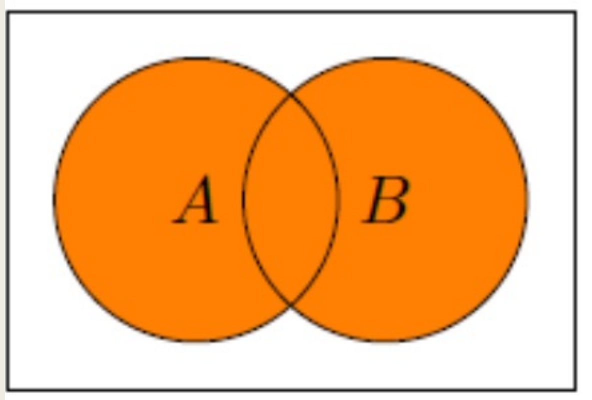
1. $(A \cap B)^c = A^c \cup B^c$
2. $(A \cup B)^c = A^c \cap B^c$



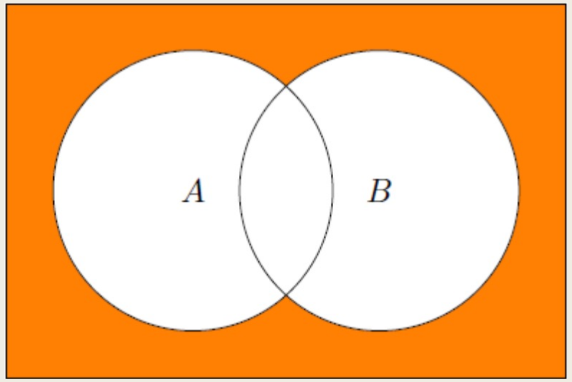
$A \cap B$



$(A \cap B)^c$



$A \cup B$



$(A \cup B)^c$

Class code:b7xpd4s

Cardinality

The cardinality, or size, of a set A is the number of elements in the set and is denoted $|A|$.

FUNCTIONS AND RELATIONS

Definition

- **Relation:** A relation r between sets X and Y is a set of ordered pairs (x, y) where $x \in X$ and $y \in Y$.
- **Domain:** The set $\{x \in X \mid (x, y) \in r \ for\ some\ y \in Y\}$ is the domain of r.
- **Range:** The set $\{y \in Y \mid (x, y) \in r \ for\ some\ x \in X\}$ is the range of r.

Function:

- A **function** f from X to Y, denoted $f: X \to Y$, is a relation that maps each element of X to exactly one element of Y.
- X is the **domain** of f.
- Y is the **codomain** of f.
- Elements of the function (x, y) are sometimes written $(x, f(x))$.
- Set of all $f(x)$ is called **range**

$$range\ of\ f = \{y \in Y |\ y = f(x)\ for\ some\ x \in X\}.$$

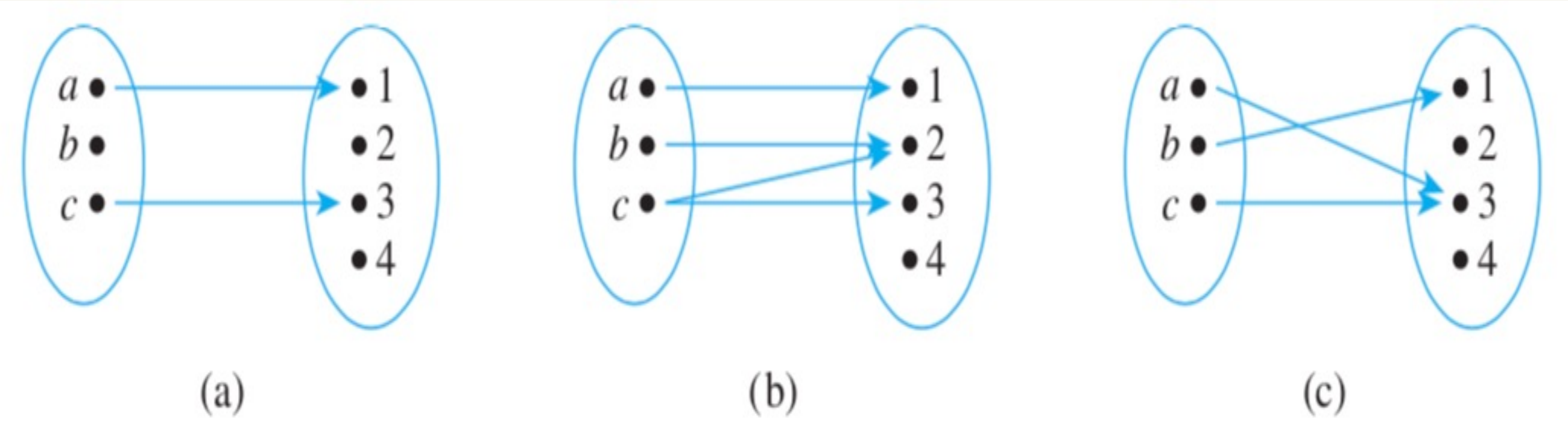
- If $f(x) = y$, then x is called **a preimage of $\boldsymbol{y}$** or **an inverse image of $\boldsymbol{y}$**.
- The set of all inverse images of y is called **the inverse image of y**. Symbolically,

$$\textbf{the inverse image of } \boldsymbol{y} = \{x \in X |\ f(x) = y\}.$$

Class code:b7xpd4s

Example:

Which of the following define functions from set $X = \{a, b, c\}$ to the set $Y = \{1,2,3,4\}$.



Relations versus functions

Let's look at $X = \{1, 2, 3, 4, 5\}$ and $Y = \{2, 4, 6, \dots\}$. Consider two relations between X and Y:

$$r = \{(3, 2), (3, 6), (5, 6)\}$$
$$s = \{(1, 4), (2, 4), (3, 8), (4, 6), (5, 2)\}$$

The domain of r is $\{3,5\}$ and the range of r is $\{2, 6\}$ while the domain of s is all of X and the range of s is $\{2, 4, 6, 8\}$.

Functions in Python

- In Python and most other programming languages, there are blocks of code known as "functions," which programmers give names and will run when you call them.
- These Python functions may or may not take inputs (referred to as "parameters") and return outputs, and each set of input parameters may or may not always return the same output.
- As such, it is important to note Python functions are not necessarily functions in the mathematical sense, although some of them are.

Example: sort()

Consider the sort() Python function, which is used for sorting lists. See this function applied to two lists – one list of numbers and one list of names:

```
numbers = [3, 1, 4, 12, 8, 5, 2, 9]
names = ['Wyatt', 'Brandon', 'Kumar', 'Eugene', 'Elise']

# Apply the sort() function to the lists
numbers.sort()
names.sort()

# Display the output
print(numbers)
print(names)
```

The output is as follows:

```
[1, 2, 3, 4, 5, 8, 9, 12]
['Brandon', 'Elise', 'Eugene', 'Kumar', 'Wyatt']
```

Example: shuffle()

```
[ ]: import random
     # Run the random.shuffle() function 5 times and display the outputs
     for i in range(0,5):
         numbers=[3, 1, 4, 12, 8, 5, 2, 9]
         random.shuffle(numbers)
         print(numbers)

     [2, 12, 1, 3, 4, 9, 8, 5]
     [1, 3, 2, 5, 9, 12, 8, 4]
     [1, 2, 5, 12, 8, 4, 3, 9]
     [12, 2, 9, 5, 3, 1, 8, 4]
     [3, 12, 1, 5, 9, 4, 8, 2]
```

Boolean Function:

An (n-place) Boolean function f is a function whose domain is the set of all ordered n-tuples of 0's and 1's and whose co-domain is the set $\{0,1\}$.

More formally, the domain of a Boolean function can be described as the Cartesian product of n copies of the set $\{0,1\}$, which is denoted $\{0,1\}^n$. Thus $f\colon \{0,1\}^n \to \{0,1\}$.

Boolean Function:

Example: Consider the three-place Boolean function defined from the set of all 3-tuples of 0's and 1's to {0, 1} as follows: For each triple (x_1, x_2, x_3) of 0's and 1's,

$$f(x_1, x_2, x_3) = (x_1 + x_2 + x_3)\ mod\ 2.$$

Describe f using an input/output table

Boolean Function:

Example: $f(x_1, x_2, x_3) = (x_1 + x_2 + x_3)\ mod\ 2.$

Describe f using an input/output table

Input			Output
x_1	x_2	x_3	
1	1	1	
1	1	0	
1	0	1	
1	0	0	
0	1	1	
0	1	0	
0	0	1	
0	0	0	

Boolean Function:

Example: $f(x_1, x_2, x_3) = (x_1 + x_2 + x_3)\ mod\ 2$.

Describe f using an input/output table

Input			Output
x_1	x_2	x_3	
1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	1
0	0	0	0

INJECTIVE, SURJECTIVE AND BIJECTIVE

- For a **one-to-one function (injective)**, each element of the co-domain is the image of at most one element of the domain.
- When a function is **onto (surjective)**, its range is equal to its co-domain. That is every element in the codomain has a preimage in the domain.
- A **one-to-one correspondence** (or **bijection**) from a set X to a set Y is a function $F: X \to Y$ that is both one-to-one and onto.

Example:

Let $X = \{1,2,3\}$ and $Y = \{a, b, c, d\}$.

Define $H\colon\ X \ \rightarrow\ Y$ as follows: $H(1)\ =\ c, H(2) = a$, and $H(3) = d$.

Define $K\colon\ X\ \rightarrow Y$ as follows: $K(1) = d, K(2) = b$, and $K(3)\ =\ d$.

Is either H or K one-to-one?

Example:

Let $X = \{1,2,3,4\}$ and $Y = \{a, b, c\}$.

Define $H\colon X \to Y$ as follows:

$$H(1) = c, H(2) = a, H(3) = c, H(4) = b.$$

Define $K\colon X \to Y$ as follows:

$$K(1) = c, K(2) = b, K(3) = b, \text{ and } K(4) = c.$$

Is either H or K onto?