VE203
Discrete Math
RC1

Yucheng Huang

University of Michigan
Shanghai Jiao Tong University
Joint Institute

February 25, 2022

1. Introduction
2. Sets
3. Logic
4. Induction
5. Q&A

What will you learn in VE203

Set I

1. Sets (Naive)
2. Logic
3. Induction $\star\star\star$
4. Relations and Functions
5. Numbers and Equinumerosity
6. Finite Sets and Pigeonhole Principle
7. Partial Order $\star\star\star$
8. Cardinality $\star\star\star$

What will you learn in VE203

Set II

1. Prime Numbers
2. Euclidean Algorithm
3. Additive Group of Integers
4. Cyclic Groups and Symmetric Groups
5. Homomorphism and Cosets $\star\star\star$
6. Modular Arithmetic $\star\star\star$
7. Chinese Remainder Theorem $\star\star\star$
8. Public Key Cryptography $\star\star\star$

What will you learn in VE203

Set III

1. Binomial Coefficients $\star\star\star$
2. Multichoosing
3. Inclusion-Exclusion Principle $\star\star\star$
4. Matrix Chain Multiplication
5. Linear Recurrence Equations $\star\star\star$
6. Asymptotic Notations
7. Master Method

What will you learn in VE203

Set IV $\star\star\star$

1. Basic Graph Theory
2. Connectivity
3. Bipartite Graph
4. Matching
5. Trees
6. Spanning Trees
7. Kruskal's Algorithm
8. Dijkstra's Algorithm

Set

Definition

A set is an unordered collection of distinct objects, called elements or members of the set. A set is said to contain its elements. We write

$a \in A$ if a is an element of the set A.

$a \notin A$ if a is not an element of the set A.

Multisets

Elements in a set are distinct and unordered.

Set

Number Systems

$\mathbb{N}$, the natural numbers
$\mathbb{Z}$, the integers
$\mathbb{Q}$, the rational numbers
$\mathbb{R}$, the real numbers
$\mathbb{C}$, the complex numbers
$\mathbb{D}$, the decimal numbers
$\mathbb{I}$, the pure imaginary numbers

$$\mathbb{D} = \left\{ \frac{a}{10^p}, a \in \mathbb{Z}, p \in \mathbb{N} \right\}$$

Set

Set Operations

A is a subset of B, denoted by $A \subset B$, if every element of A is an element of B.
B is called a superset of A, denoted by $B \supset A$.
$A = B$ if and only if $A \subset B$ and $B \subset A$. (cf., $x = y$ iff $x \leq y$ and $y \leq x$.)

Set

Union

The union of A and B is the set of elements in either A or B, denoted by

$$A \cup B := \{x \mid x \in A \text{ or } x \in B\}$$

Intersection

The intersection of A and B is the set of elements in both A and B, denotedy by

$$A \cap B := \{x \mid x \in A \text{ and } x \in B\}$$

Set

Set Difference

The set difference of A and B, denoted by $A - B$, or $A \backslash B$, is the set of elements in A but not in B, that is,

$$\begin{aligned} A - B :&= \{x \mid x \in A \text{ and } x \notin B\} \\ &= \{x \in A \mid x \notin B\} \end{aligned}$$

Symmetric Difference

The symmetric difference of A and B is the set of elements that are in exclusively one of A and B, but not the other.

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

Set

Power Set

The power set of a set A is the set of all subsets of A, denoted by $\mathcal{P}(A)$ or 2^A.

Vien Graph and Eule Graph

See Blackboard

Set

Set Algebras

Commutative Laws

$\rightarrow A \cup B = B \cup A$

$\rightarrow A \cap B = B \cap A$

Associative Laws

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

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

(Left) Distributive Laws

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

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

De Morgan's Laws

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

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

Set

Cartesian Product

The Cartesian product of sets A and B is the set of ordered pairs, such that

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

Exercise

See Worksheet 1

Logic

Definition

A proposition or statement is a declarative sentence that is either true or false, but not both.

Logic

True or False

True: $\mathrm{T}, 1, \top$

False: $\mathrm{F}, 0, \perp$

Connectives

$\neg$, negation/not

$\wedge$, and

$\vee$, or (inclusive or)

$\rightarrow$, implies

$\leftrightarrow$, if and only if (iff)

Truth Table

AND

p	q	$p \wedge q$
0	0	0
0	1	0
1	0	0
1	1	1

OR

p	q	$p \vee q$
0	0	0
0	1	1
1	0	1
1	1	1

Truth Table

NOT

p	$\neg p$
0	1
1	0

XOR

p	q	$p \oplus q$
0	0	0
0	1	1
1	0	1
1	1	0

Truth Table

Conditional statement

p	q	$p \to q$
0	0	1
0	1	1
1	0	0
1	1	1

if and only if (iff)

p	q	$p \to q$	$q \to p$	$p \leftrightarrow q$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	1	1	1

Tautology and Contradiction

Tautology: All cases evaluates to 1 . (e.g., $p \vee \neg p$)
Contradiction: All cases evaluates to 0 . (e.g., $p \wedge \neg p$)

Tautological Equivalence

Commutativity

$$p \wedge q \Leftrightarrow q \wedge p$$

$$p \vee q \Leftrightarrow q \vee p$$

Associativity

$$(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$$

$$(p \vee q) \vee r \Leftrightarrow p \vee (q \vee r)$$

$$(p \leftrightarrow q) \leftrightarrow r \Leftrightarrow p \leftrightarrow (q \leftrightarrow r)$$

$$(p \oplus q) \oplus r \Leftrightarrow p \oplus (q \oplus r)$$

Distributivity

$$p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$$

$$p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \vee r)$$

Tautological Equivalence

Negation

$$\neg\neg p \Leftrightarrow p$$

$$p \rightarrow q \Leftrightarrow \neg q \rightarrow \neg p$$

$$\neg(p \vee q) \Leftrightarrow (\neg p) \wedge (\neg q)$$

$$\neg(p \wedge q) \Leftrightarrow (\neg p) \vee (\neg q)$$

Identity

$$p \vee 0 \Leftrightarrow p \quad (\max\{p, 0\} = p)$$
$$p \wedge 1 \Leftrightarrow p \quad (\min\{p, 1\} = p)$$

Null

$$p \wedge 0 \Leftrightarrow 0 \quad (\min\{p, 0\} = 0)$$
$$p \vee 1 \Leftrightarrow 1 \quad (\max\{p, 1\} = 1)$$

Tautological Equivalence

Idempotent

$$p \wedge p \Leftrightarrow p$$
$$p \vee p \Leftrightarrow p$$

Absorption

$$p \wedge (p \vee q) \Leftrightarrow p$$
$$p \vee (p \wedge q) \Leftrightarrow p$$

Cases

$$(p \rightarrow q) \wedge (p \rightarrow r) \Leftrightarrow p \rightarrow (q \wedge r)$$
$$(p \rightarrow q) \vee (p \rightarrow r) \Leftrightarrow p \rightarrow (q \vee r)$$
$$(p \rightarrow r) \wedge (q \rightarrow r) \Leftrightarrow (p \vee q) \rightarrow r$$
$$(p \rightarrow r) \vee (q \rightarrow r) \Leftrightarrow (p \wedge q) \rightarrow r$$

Added premise

$$\begin{aligned}(p \wedge q) \rightarrow r &\Leftrightarrow p \rightarrow (q \rightarrow r)\\ &\Leftrightarrow q \rightarrow (p \rightarrow r)\end{aligned}$$

Tautological Equivalence

DeMorgan's Law

$$\neg(p \vee q) \Leftrightarrow \neg p \wedge \neg q$$

$$\neg(p \wedge q) \Leftrightarrow \neg p \vee \neg q$$

CNF and DNF

CNF

For any proposition φ, there is a proposition φ_{cnf} over the same Boolean variables and in CNF such that $\varphi \Leftrightarrow \varphi_{\text{cnf}}$.

$$
\begin{array}{ll}
\varphi = p \vee q & \varphi_{cnf} = (p \vee q) \\
\varphi = p \wedge q & \varphi_{cnf} = (p) \wedge (q) \\
\varphi = p \rightarrow q & \varphi_{cnf} = (\neg p \vee q) \\
\varphi = p \leftrightarrow q & \varphi_{cnf} = (\neg p \vee q) \wedge (\neg q \vee p) \\
\varphi = p \oplus q & \varphi_{cnf} = (p \vee q) \wedge (\neg q \vee \neg p)
\end{array}
$$

CNF and DNF

DNF

For any proposition φ, there is a proposition φ_{dnf} over the same Boolean variables and in DNF such that $\varphi \Leftrightarrow \varphi_{\text{dnf}}$.

$$
\begin{array}{ll}
\varphi = p \vee q & \varphi_{dnf} = (p) \vee (q) \\
\varphi = p \wedge q & \varphi_{dnf} = (p \wedge q) \\
\varphi = p \rightarrow q & \varphi_{dnf} = (\neg p) \vee (q) \\
\varphi = p \leftrightarrow q & \varphi_{dnf} = (p \wedge q) \vee (\neg q \wedge \neg p) \\
\varphi = p \oplus q & \varphi_{dnf} = (\neg p \wedge q) \vee (p \wedge \neg q)
\end{array}
$$

Example (From Mid1 FA2021)

Given the logical proposition $\varphi = p \to (q \wedge r)$

1. Write the truth table for φ.

p	q	r	$p \to (q \wedge r)$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Example (From Mid1 FA2021)

2. Express φ in disjunctive normal form (i.e., sum of products)

φ_{dnf}.

By the truth table, we can write

$$\varphi_{\text{dnf}} = (\neg p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \vee (p \wedge q \wedge r)$$

3. Express φ in conjunctive normal form (i.e., product of sums)

φ_{cnf}.

Based on the truth table, let

$$\neg \varphi_{\text{cnf}} = (p \wedge q \wedge \neg r) \wedge (p \wedge \neg q \wedge r) \wedge (p \wedge \neg q \wedge \neg r)$$

then

$$\begin{aligned} \varphi_{\text{cnf}} &= \neg[(p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge r) \vee (p \wedge \neg q \wedge \neg r)] \\ &= \neg(p \wedge q \wedge \neg r) \wedge \neg(p \wedge \neg q \wedge r) \wedge \neg(p \wedge \neg q \wedge \neg r) \\ &= (\neg p \vee \neg q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (\neg p \vee q \vee r) \end{aligned}$$

Exercise

See Worksheet 1

Induction

Typically one wants to show that some statement frame $P(n)$ is true for all $n \in \mathbb{N}$ with $n \geq n_0$ for some $n_0 \in \mathbb{N}$. Mathematical induction works by establishing two statements:
(I) the base case: $P\left(n_0\right)$ is true.
(II) the inductive case: $P(n+1)$ is true whenever $P(n)$ is true for $n \geq n_0$, i.e.,

$$(\forall n \in \mathbb{N}, n \geq n_0)\,(P(n) \Rightarrow P(n+1))$$

In the inductive case, $P(n)$ is called inductive hypothesis, often abbreviated as *IH*.
Note that (II) does not make a statement on the situation when $P(n)$ is false; it is permitted for $P(n+1)$ to be true even if $P(n)$ is false.
The principle of mathematical induction now claims that $P(n)$ is true for all $n \geq n_0$ if (I) and (II) are true.

Induction

Algorithm 1 Factorial

Input: n, a positive integer

Output: $n!$

```
Function: fact(n)
if n = 1 then
    return 1
else
    return n·fact(n − 1)
end if
```

Induction

base case $(n = 1)$: Observe that f act (1) returns 1 immediately, and $1! = 1$.
inductive case $(n \geq 1)$: Assume that $f\,\mathrm{act}(n)$ returns $n!$. We want to show that fact $(n + 1)$ returns $(n + 1)!$. Indeed, by induction hypothesis,

$$\mathrm{fact}(n + 1) = n \cdot \mathrm{fact}(n) = (n + 1) \cdot n! = (n + 1)!$$

Sort (Optional: you will see it in VE281)

Exchange Family

Bubble Sort $\star$
Cocktail Sort
Gnome Sort
Comb Sort
Quick Sort $\star$

Selection Family

Selection Sort $\star$
Heap Sort $\star$
Smooth Sort
Tournament Sort

Sort (Optional: you will see it in VE281)

Insertion Family

Insertion Sort $\star$
Shell Sort
Cycle Sort

Merge Family

Merge Sort $\star$
In-Place Merge Sort $\star$

Sort (Optional: you will see it in VE281)

Non-comparison Family

Bucket Sort $\star$
Bead Sort
Counting Sort $\star$
Pigeonhole Sort
Flash Sort

Hybrid

TimSort
std::sort
GrailSort

For fun

Bogo Sort

Q&A

Q&A