Speed Running Discrete Mathematics

Minh Nguyen

November 14, 2023

Contents

Author’s Notes

This actually isn’t the first copy of my discrete math notes…I had an earlier copy that got corrupted, so now I’m just going to constantly upload to GitHub because I’m afraid of them getting wiped again 😭

These notes are based off of the textbook *Discrete Mathematics and Its Applications* by *Kenneth H. Rosen*, as well as from the lectures of *Serdar Erbatur* from the *Fall 2023* semester.

If you have any complaints, or suggestions regarding these notes, please email me at
mdn220004@utdallas.edu

1 Propositional Logic

Definition **Proposition**

Declarative statements that are either `true` or `false`

Table 1: Examples of propositions

Statement	Proposition?
$1 + 1 = 2$	✔️
What class are ya takin?	⛔
I am happy	✔️

Propositons can be combined to form another proposition with the use of *logical operators*

Note

Common labels for propositions in discrete mathematics include letters like P, Q, R, S...

Table 2: Logical Operators in Discrete Mathematics

Symbol	Meaning	Expression
$\neg$	negation / not	$\neg P$
$\land$	conjunction / and	$P \land Q$
$\lor$	disjunction / or	$P \lor Q$
$\oplus$	exclusive disjunction/ xor	$P \oplus Q$
$\implies$	implication / conditional	$P \implies Q$
$\iff$	biconditional	$P \iff Q$

Note **Logical Operator Precedence**

Order matters when it comes to evaluating logical operators:

1. negaiton
2. conjunction
3. disjunctions
4. conditionals

Where negation is evaluated first, and conditionals last.

Theorem More on Conditionals

Conditionals can also be expressed in three other ways:

Contrapositive: $\neg Q \implies \neg P$

Converse: $Q \implies P$

Inverse: $\neg P \implies \neg Q$

Example

Conditional: If it is raining, then I am not going to town

Contrapositive: If I go to town, then it is not raining

Converse: If I do not go to town, then it is raining

Inverse: If it is not raining, then I am going to town

Definition Truth Table

They are used as a way of seeing all possible values of a proposition

Table 3: Truth Table of $P \implies Q$

P	Q	$P \implies Q$
F	F	T
F	T	T
T	F	F
T	T	T

1.1 Propositional Equivalencies

Definition Equivalencies

A proposition is a...

- **Tautology** if it is `true` in *every case*
- **Contradiction / Fallacy** if it is `false` in *every case*
- **Contingency** if *neither* is the case

Table 4: Example of Tautology and Contradiction

P	$\neg P$	$P \vee \neg P$	$P \wedge \neg P$
T	F	T	F
F	T	T	F

Definition **Logical Equivalency** ($\equiv$)

Compound propositions P and Q are *logically equivalent* if $P \iff Q$ is a *tautology*. This is expressed as

$$P \equiv Q$$

Below is a table of useful logical equivalencies

Table 5: Logical Equivalencies

Name	**Equivalence**
Identity	$P \wedge \mathrm{T} \equiv P$
	$P \vee \mathrm{F} \equiv P$
Idempotent	$P \wedge P \equiv P$
	$P \vee P \equiv P$
Domination	$P \vee \mathrm{T} \equiv \mathrm{T}$
	$P \wedge \mathrm{F} \equiv \mathrm{F}$
Negation	$P \vee \neg P \equiv \mathrm{T}$
	$P \wedge \neg P \equiv \mathrm{F}$
Double Negation	$\neg(\neg P) \equiv P$
Commutative	$P \wedge Q \equiv Q \wedge P$
	$P \vee Q \equiv Q \vee P$
Associative	$(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$
	$(P \vee Q) \vee R \equiv P \vee (Q \vee R)$
Distributive	$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$
	$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$
De Morgan's	$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$
	$\neg(P \vee Q) \equiv \neg P \wedge \neg Q$
Absorption	$P \wedge (P \vee Q) \equiv P$
	$P \vee (P \wedge Q) \equiv P$

The use of showing the equivalencies between two compound propositions is called a *conditional-disjunction equivalence*

Example **Conditional-Disjunction Equivalence**

$$P \implies Q \equiv \neg P \vee Q$$

This can be proven with the use of a truth table 🥰

Continued on Next Page...

Table 6: Conditional-Disjunction Equivalence Proof

P	Q	$\neg P$	$\neg P \lor Q$	$P \implies Q$
F	F	T	T	T
F	T	T	T	T
T	F	F	F	F
T	T	F	T	T

1.2 Predicates and Quantifiers

Note

So far, propositional logic can only handle *singular* subjects. It can't handle statements such as:

- All computer science students can program well
- $3x + 4 \geq 0$

Definition **Predicate Logic**

When you have a proposition that contains a variable. It is typically written as a *propositional function*, $P(x)$, where x is the subject for the predicate P

Example

- $F(x) =$ "$x > 3$"
- $P(x) =$ "x looks beautiful!"

Note **Multi-variable Propositional Functions**

Propositional functions may also contain more than just one argument:

$$P(x, y)$$

Definition **Quantifiers**

They define the range of which a proposition holds `true`, and can be nested to produce *nested quantifiers*

Table 7: Quantifiers in Discrete Mathematics

Quantifier	**Expression**	**In English**
Universal Quantifier	$\forall x.P(x)$	$P(x)$ is `true` for every x in its domain
Existential Quantifier	$\exists x.P(x)$	There exists x where $P(x)$ is `true`
	$\nexists x.P(x)$	There does not exist x where $P(x)$ is `true`
	$\exists! x.P(x)$	There exists *only one* x where $P(x)$ is `true`

$\forall x.P(x)$ may also be written as $\forall x P(x)$, and the same holds for the other quantifiers

Example **Nested Quantifiers**

$$\forall x \exists y.(x + y = 0)$$

Translates to: "For *every* x *there exists* a y such that $x + y = 0$"

1.3 Inference Rules and Proofs

Definition **Argument**

A sequence of statements that have a conclusion

Definition **Valid**

The conclusion, the final statement of the argument, must follow from its premises
(i.e. premises $\implies$ conclusion)

Definition **Premise**

The preceeding statements of a mathematical argument that lead to a conclusion

Definition **Fallacy**

Incorrect reasoning in discrete mathematics that leads to an invalid argument

Example **Argumentative Form**

Arguments may be written as this: $\big((P \implies Q) \wedge P\big) \implies Q$, or in argumentative form:

$$\begin{array}{l} P \implies Q \\ \underline{P \qquad\quad\;} \\ \therefore Q \end{array}$$

The next page contains a table of inference rules

Table 8: Rules of Inference

Rule	Expression	Tautology
Modus ponens	P $\underline{P \implies Q}$ $\therefore\ Q$	$\big(P \land (P \implies Q)\big) \implies Q$
Modus tollens	$\neg Q$ $\underline{P \implies Q}$ $\therefore\ \neg P$	$\big(\neg Q \land (P \implies Q)\big) \implies \neg P$
Hypothetical syllogism	$P \implies Q$ $\underline{Q \implies R}$ $\therefore\ P \implies R$	$\big((P \implies Q) \land (Q \implies R)\big) \implies (P \implies R)$
Disjunctive syllogism	$P \lor Q$ $\underline{\neg P}$ $\therefore\ Q$	$\big((P \lor Q) \land \neg P\big) \implies Q$
Addition	$\underline{P}$ $\therefore\ P \lor Q$	$P \implies (P \lor Q)$
Simplification	$\underline{P \land Q}$ $\therefore\ P$	$(P \land Q) \implies P$
Conjunction	P $\underline{Q}$ $\therefore\ P \land Q$	$\big((P) \land (Q)\big) \implies P \land Q$
Resolution	$P \lor Q$ $\underline{\neg P \lor R}$ $\therefore\ Q \lor R$	$\big((P \lor Q) \land (\neg P \lor R)\big) \implies (Q \lor R)$

Table 9: Rules of Inference for Quantified Statements

Rule	Expression
Universal Instantiation	$\underline{\forall x.P(x)}$ $\therefore\ P(c)$
Universal Generalization	$\underline{P(c) \text{ for an arbitrary } c}$ $\therefore\ \forall x.P(x)$
Existential Instantiation	$\underline{\exists x.P(x)}$ $\therefore\ P(c) \text{ for some element } c$
Existential Generalization	$\underline{P(c) \text{ for an arbitrary } c}$ $\therefore\ \forall x.P(x)$

Definition **Proof**

Valid arguments that establish the `truth` of mathematical statements

Definition **Theorem**

A statement or claim that can be proven using:

- definitions
- other theorems
- axioms
- inference rules

They are also referred to as "Lemma", "Proposition", or "Corollary"

There's many different ways of proving a theorem. Let's assume a conditional statement $P \implies Q$...

Definition **Direct Proof**

When the first step is the assumption that P is `true` and the following steps that lead up to Q is also `true`

Definition **Indirect Proof**

Proofs that do not start with the premimses and end with conclusion (the opposite of a direct proof). Ways of doing direct proofs involve *proof by contraposition*, *proof by contradiction*, and much more...

Example **Proof by Contraposition**

$P \implies Q$ can be proved `true` if $\neg Q \implies P$, its contraposition, can also be proved. This is because a contraposition and a conditional proposition are tautologies

Example **Proof by Contradiciton**

To prove P, you must assume $\neg P$ and derive that $\neg P$ is `false`. If $\neg P$ is `false`, then that must mean that $\neg\neg P$, or P, must be `true`

2 Set Theory

Definition **set**

A collection of *unique* objects or elements

- $a \in A$ to state that a is *contained* in the set A
- $a \notin A$ to indicate that a *is not* contained in A

Note **Defining Sets**

Roster method

$$\begin{aligned} A &= \{a, b, c, d\} \\ B &= \{4, b, c, a\} \\ C &= \big\{1, 2, \{3.0, 3.5\}, 4, 5\big\} \\ D &= \{1, 2, C\} \end{aligned}$$

Set Builder Notation

$$\begin{aligned} E &= \{x \in \mathbb{N} \big| x = 2k \text{ for some } k \in \mathbb{N}\} \\ F &= \{\alpha \big| P(\alpha) \text{ is } \texttt{true}\} \end{aligned}$$

Table 10: Important Sets

Set	Expansion	Description
$\mathbb{N}$	$\{0, 1, 2, 3, \cdots\}$	Natural numbers
$\mathbb{Z}$	$\{\cdots, -1, 0, 1, \cdots\}$	Integers
$\mathbb{Z}^+$	$\{1, 2, 3, \cdots\}$	Positive integers
$\mathbb{Q}$	$\{\frac{P}{Q} \big| P \in \mathbb{Z}, Q \in \mathbb{Z}, Q \neq 0\}$	Rational numbers
$\mathbb{R}$		Real numbers
$\mathbb{R}^+$	$\{x \big| x > 0\}$	Positive real numbers
$\mathbb{C}$		Complex numbers
$\varnothing$		Empty set

2.1 Subsets

Theorem **Set Equality**

Given two sets A and B:

$$A \equiv B \iff \forall x.(x \in A \iff x \in B)$$

is `true`

Definition **Subset ($\subseteq$)**

A is a subset of B if *all elements of* A are also contained in B:

$$A \subseteq B \iff \forall x.(x \in A \implies x \in B)$$

Example **Subsets**

Let $A = \{2, 3\}$, $B = \{a, b, c, 2, 3\}$

- $A \subseteq B$ `true`
- $A \subseteq A$ `true`
- $\varnothing \subseteq A$ `true`
- $B \subseteq A$ `false`

This notation is seen in Section 2.2

Definition **Power Set**

The set of all subsets of the set

$$|A| = n, |P(A)| = 2^n$$

Example

Let $S = \{1, 2\}$:

$$P(S) = \{\varnothing, \{1\}, \{2\}, \{1, 2\}\}$$

Definition **Proper Subset ($\subset$)**

It's a subset where $A \neq B$:

$$A \subset B \iff \forall x.(x \in A \implies x \in B) \land A \neq B$$

2.2 Cardinality

Definition **Cardinality**

It's the number of unique elements in a set. It's denoted by $|A|$

Example

$\{\{1, 2\}, 3\}$	2 elements
$\{1, 2, 3\} = \{1, 2, 3, 3\}$	3 elements
$\{\varnothing\}$	1 element
$\{\}$	0 elements
$\{\varnothing\}$	1 element

Definition **Intervals**

Sets of numbers *between* two numbers a and b if $a, b \in \mathbb{R} \land a \leq b$

- $[a, b] = x|a \leq x \leq b$
- $[a, b) = x|a \leq x < b$
- $(a, b] = x|a < x \leq b$
- $(a, b) = x|a < x < b$

Definition **Ordered Tuples**

$$(a_0, a_1, a_2, \cdots, a_n)$$

An *ordered collection* of unique elements

$$(a, b, c) \neq (b, c, a)$$

Theorem **Cartesian Products**

Let A and B be sets:

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

Example **Cartesian Product**

Let $A = \{1, 2\}, B = \{a, b, c\}$:

$$A \times B = \{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c)\}$$

2.3 Set Operations

Let $A = \{1, 2, 3\}, B = \{2, 3, 4, 5\}$:

Table 11: Set Operations

Expression	**Meaning**	**Result**
$A \cup B$	A union B	$\{1, 2, 3, 4, 5\}$
$A \cap B$	A intersection B	$\{2, 3\}$
$\bar{A}$	A complement B	$\{x \notin A\}$

Definition **Membership Table**

It's very similar to a truth table, where `true` means that an element exists within the set, and `false` means that the element isn't.

Table 12: Membership Table Example

A	B	$A \cup B$	$A \cap B$
`false`	`false`	`false`	`false`
`false`	`true`	`true`	`false`
`true`	`false`	`true`	`false`
`true`	`true`	`true`	`true`

2.4 Functions

Definition **Function**

Function f is denoted from set A to set B:

$$f : A \to B$$

representing a relation that assigns each element of A to *exactly one* element of B

Example

Let $A = \{1, 2\}, B = \{a, b\}$

$$A \times B = \{(1, a), (1, b), (2, a), (2, b)\}$$

These are functions:

- $f = \{(1, a), (2, b)\}$
- $f = \{(1, a), (2, a)\}$

This isn't:

- $f = \{(1, a), (1, b)\}$

Theorem **Equality of Functions**

Two functions f and g are equal if:

- They share the same domains / co-domains
- They assign the same element from the domain to the same element

Let $f : A \to B, g : C \to D$:

- $A \neq C \lor B \neq D \implies f \neq g$
- $A = C \lor B = D \implies f = g$

Definition **One-to-One Functions (Injective)**

A function $f : A \to B$ where all values A must correspond to *one and only one* element in B:

$$\forall a \in A, \forall b \in B.\big(f(a) \neq f(b)\big)$$

Definition **Onto Functions (Subjective)**

A function $f : A \to B$ where all values B must correspond to *at least an* element in A:

$$\forall b \in B, \exists a \in A.\big(f(a) = b\big)$$

Definition **Bijection**

A function that is *both* one-to-one *and* onto

Definition **Inverse of Function**

$$f^{-1}(x) = y \iff f(y) = x$$

Definition **Compisite Functions**

$$\text{domain}(f) = \text{range}(g) \implies f \circ g(x) = f(g(x))$$

If the composite of the function exists, then:

$$(f \circ g)^{-1} = g^{-1} \circ f^{-1}$$

Definition **Floor** ($\lfloor x \rfloor$)

The biggest integer $n \leq x$

Definition **Ceiling** ($\lceil x \rceil$)

The smallest integer $n \geq x$

3 Algorithms

Definition **Algorithm**

A sequence of *precise* steps used to solve computational problems

Note **Input-Output**

For every input instance, a correct output *must* be produced

---

Algorithm 1: Binary Search Algorithm

Input: $A \leftarrow \{A_1, A_2, \ldots, A_n\}$ where $A_i \in \mathbb{Z}$, $k \in \mathbb{Z}$
Output: i where $A_i = k$, or $i = 0 \implies k \notin A$

left$\leftarrow 1$
right$\leftarrow n$
while $left < right$ **do**
    $i \leftarrow \lfloor (\text{left} + \text{right}) \div 2 \rfloor$
    if $A_i = k$ **then**
        return i
    else if $A_i < k$ **then**
        left$\leftarrow i + 1$
    else
        right$\leftarrow i$
return 0

Algorithm 2: Insertion Sort

Input: $A \leftarrow \{A_1, A_2, \ldots, A_n\}$ where $n \geq 2$
Output: $A \leftarrow \{A_1, A_2, \ldots, A_n\}$ where $n \geq 2 \wedge A_i \leq A_{i+1}$

for $i \leftarrow 2$ **to** n **do**
    key$\leftarrow A_{\text{i}}$
    index$\leftarrow \text{i} - 1$
    while $index \geq 1 \wedge A_{index} > key$ **do**
        $A_{\text{index}+1} \leftarrow A_{\text{index}}$
        index $\leftarrow$ index $- 1$
    $A_{\text{index}+1} \leftarrow$ key

3.1 Growth of Functions

This can be used to determine the growth of functions, and is also mainly used in the field of *computer science* to analyze algorithms.

Note **Proving Big-O**

To prove that $f(x) = O\big(g(x)\big)$, you must prove the existence of a C and k that fulfill the definition:

$$\exists C, k. \big(\forall x > k. (|f(x)| \leq C \times |g(x)|)\big)$$

Definition **Big-O Notation**

Let f and g be functions. $f(x)$ is $O\big(g(x)\big)$ (read "$f(x)$ is Big-O of $g(x)$") when there are constants C and k (the witnesses) such that:

$$|f(x)| \leq C|g(x)|, x > k$$

This is called the worst-case runtime of an algorithm

Example **Identify the Witnesses**

Find C, k such that $f(x) = 4x^2 + 2x + 1$ is $O(x^2)$

$$\begin{aligned} g(x) &= n^2 + n^2 + n^2 \\ &= 3n^2 \end{aligned}$$

$$f(x) < g(x) \text{ when } x > 1 \therefore C = 3, k = 1$$

Definition **Big-Ω**

Let f and g be functions. $f(x)$ is $\Omega\big(g(x)\big)$ when there are cosntants C and k such that:

$$|f(x)| \geq C|g(x)|$$

This is called the best-case runtime of an algorithm

Definition **Big-θ**

Let f and g be functions. $f(x)$ is $\theta\big(g(x)\big)$ when:

$$f(x) = O\big(g(x)\big) \wedge f(x) = \Omega\big(g(x)\big)$$

This is called the average-case runtime of an algorithm