

MATH1064: Discrete Mathematics for Computation
Complete Exam Preparation Notes with Worked Examples and Practice Examination

MATH1064

Discrete Mathematics for Computation

Complete Exam Preparation Notes with Worked Examples and Practice Examination

Independent study document compiled from the supplied 2025 unit materials

Contents

1	Sets and Set Operations	3
1.1	Sets, elements, and subsets	3
1.2	Set-builder notation and operations	3
1.3	Set identities and proofs	4
1.4	Power sets and Cartesian products	4
2	Functions and Sequences	4
2.1	Function terminology	5
2.2	Injective, surjective, and bijective functions	5
2.3	Composition	5
2.4	Sequences and recurrences	6
2.5	Summation notation	6
3	Number Theory	6
3.1	Divisibility	6
3.2	Quotient-remainder theorem	7
3.3	Congruence	7
3.4	Primes, greatest common divisors, and least common multiples	7
3.5	Euclidean algorithm and Bezout identity	8
4	Counting and Catalan Structures	8
4.1	Sum, product, subtraction, and division rules	8
4.2	Permutations and combinations	8
4.3	Repetition and stars and bars	9
4.4	Binomial theorem and identities	9
4.5	Inclusion-exclusion and pigeonhole principle	10
4.6	Catalan structures	10
5	Discrete Probability	10
5.1	Sample spaces and events	11
5.2	Conditional probability and independence	11
5.3	Total probability and Bayes' theorem	12
5.4	Random variables, expectation, and variance	12
6	Graph Theory	12
6.1	Basic terminology and degrees	13
6.2	Representations and isomorphism	13
6.3	Walks, paths, circuits, and connectivity	13
6.4	Adjacency-matrix powers	13
7	Relations and Orders	14
7.1	Relations and their properties	14
7.2	Equivalence relations and partitions	14
7.3	Partial and total orders	14
7.4	Composition and closure	15
8	Logic and Proof	15
8.1	Propositions and connectives	15
8.2	Logical equivalence	16
8.3	Predicates and quantifiers	16
8.4	Rules of inference	16
8.5	Proof techniques	16
9	Induction, Recursion, and Complexity	17
9.1	Mathematical induction	17
9.2	Strong induction and recursion	17
9.3	Linear recurrences	18
9.4	Growth notation	18
9.5	Algorithm analysis	18
10	Formal Languages and Grammars	19
10.1	Alphabets, words, and languages	19
10.2	Grammars and derivations	19

10.3 Grammar hierarchy	19
11 Finite-State Models and Regular Languages	20
11.1 Finite-state machines with output	20
11.2 Finite-state automata	20
11.3 Regular languages	20
12 Cardinality of Infinite Sets	21
12.1 Comparing cardinalities	21
12.2 Countable sets	21
12.3 Diagonal argument and power sets	21
13 Consolidated Formula Sheet	22
13.1 Sets and functions	22
13.2 Sequences and sums	22
13.3 Number theory	22
13.4 Counting	22
13.5 Probability	22
13.6 Graphs and relations	23
13.7 Logic, induction, and complexity	23
13.8 Languages, automata, and cardinality	23
14 Glossary	24
15 MATH1064 Practice Examination	25
15.1 Candidate instructions	25
15.2 Section A: Short response - 32 marks	25
15.3 Section B: Extended response - 40 marks	26
15.4 Section C: Synthesis - 28 marks	27
16 Worked Answer Key	28
16.1 Section A solutions	28
16.2 Section B solutions	30
16.3 Section C solutions	31

1 Sets and Set Operations

Set theory supplies the language used throughout the unit. Functions, relations, graphs, probability spaces, and formal languages are all built from sets, so precise notation at this stage prevents logical errors later. This chapter consolidates the set notation and proof methods used in the supplied lecture, tutorial, practice, and revision materials.

1.1 Sets, elements, and subsets

Definition: A **set** is a collection of distinct objects called **elements**. If an object x belongs to a set A , write $x \in A$; otherwise write $x \notin A$.

Here, x denotes an object and A denotes a set.

Definition: A set A is a **subset** of a set B , written $A \subseteq B$, when every element of A is also an element of B .

Here, A and B denote sets.

The empty set, written $\emptyset$, contains no elements. It is a subset of every set because there is no element of $\emptyset$ that can violate the subset condition. Set equality is stronger than equal cardinality: two sets are equal only when they contain exactly the same elements.

The standard proof of set equality is

$$A = B \iff A \subseteq B \text{ and } B \subseteq A. \quad (1.1)$$

where:

- A and B are the sets being compared;
- $\subseteq$ denotes the subset relation.

Membership and containment must not be confused. If $A = \{1, 2, 3\}$, then $2 \in A$, while $\{2\} \subseteq A$. The statement $\{2\} \in A$ is false because the elements of A are numbers, not singleton sets.

1.2 Set-builder notation and operations

Set-builder notation describes elements by a property:

$$A = \{x \in X : P(x)\}. \quad (1.2)$$

where:

- X is the ambient set or domain;
- x is a candidate element of X ;
- $P(x)$ is the property required for membership in A .

Let A and B be subsets of a universal set U . The main operations are summarised below.

Operation	Notation	Membership condition
Union	$A \cup B$	$x \in A$ or $x \in B$
Intersection	$A \cap B$	$x \in A$ and $x \in B$
Difference	$A \setminus B$	$x \in A$ and $x \notin B$
Complement	A^c	$x \in U$ and $x \notin A$
Symmetric difference	$A \triangle B$	x belongs to exactly one of A and B

The complement depends on the chosen universal set U . A statement involving A^c is incomplete if the ambient universe has not been established.

1.2.1 Worked example 1.1: computing set operations

Let

$$U = \{1, 2, 3, 4, 5\}, \quad A = \{1, 2, 4\}, \quad B = \{2, 3\}.$$

where U is the universal set and $A, B \subseteq U$. Then

$$A \cup B = \{1, 2, 3, 4\}, \quad A \cap B = \{2\},$$

$$A \setminus B = \{1, 4\}, \quad A^c = \{3, 5\}.$$

The union collects elements occurring in at least one set. The intersection retains only common elements. The difference begins with A and removes elements of B , while the complement removes A from the declared universe.

1.3 Set identities and proofs

Law	Union form	Intersection form
Identity	$A \cup \emptyset = A$	$A \cap U = A$
Domination	$A \cup U = U$	$A \cap \emptyset = \emptyset$
Idempotent	$A \cup A = A$	$A \cap A = A$
Commutative	$A \cup B = B \cup A$	$A \cap B = B \cap A$
Associative	$(A \cup B) \cup C = A \cup (B \cup C)$	$(A \cap B) \cap C = A \cap (B \cap C)$
Distributive	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
De Morgan	$(A \cup B)^c = A^c \cap B^c$	$(A \cap B)^c = A^c \cup B^c$

1.3.1 Worked example 1.2: an element-chasing proof

Prove

$$A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C). \quad (1.3)$$

where A , B , and C are subsets of the same universe. Choose arbitrary x in that universe. Then

$$\begin{aligned}
 x \in A \setminus (B \cup C) &\iff x \in A \text{ and } x \notin B \cup C \\
 &\iff x \in A \text{ and } x \notin B \text{ and } x \notin C \\
 &\iff x \in A \setminus B \text{ and } x \in A \setminus C \\
 &\iff x \in (A \setminus B) \cap (A \setminus C).
 \end{aligned}$$

Because the membership conditions are equivalent for every x , the sets are equal.

1.4 Power sets and Cartesian products

Definition: The **power set** of A , written $\mathcal{P}(A)$, is the set of all subsets of A .

If A has n elements, then

$$|\mathcal{P}(A)| = 2^n. \quad (1.4)$$

where $n = |A|$ is the number of elements of the finite set A . Each element has two independent states: included or excluded.

Definition: The **Cartesian product** of A and B is

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

where a is an element of A and b is an element of B .

For finite sets,

$$|A \times B| = |A| |B|. \quad (1.5)$$

where $|A|$ and $|B|$ are the cardinalities of the factor sets. Ordered pairs retain position, so (a, b) and (b, a) need not be equal. This distinction becomes essential when Cartesian products define functions and relations.

Chapter recap: Prove set equality by double inclusion or a chain of equivalent membership statements.

Keep $\in$ separate from $\subseteq$, state the universe before taking complements, and identify whether the elements under discussion are objects, sets, or ordered pairs.

2 Functions and Sequences

Functions formalise assignments between sets and reappear in relations, countability, and finite-state models. Sequences are functions with integer index sets, so the sequence material extends the same language. The central exam habit is to work from the declared domain and codomain rather than from a formula alone.