CS1231S Discrete Structures
AY23/24, Y1S1
Notes

Sim Ray En Ryan

December 3, 2025

Contents

1 Speaking Mathematically

1.1 Mathematical Notation

- **Sets**: N, Z, Q, R, C
- **Statements**:
  - **Universal**: All, every, any ($\forall$)
  - **Conditional**: If, then ($\rightarrow$)
  - **Existential**: There exists ($\exists$)
- **Terms**
  - **Definition**
  - **Axiom/Postulate**
  - **Theorem**
  - **Lemma**
  - **Corollary**
  - **Conjecture**

1.2 Properties of Integers

- **Closure**
- **Commutativity**
- **Associativity**
- **Distributivity**
- **Trichotomy**
- **Definition of Even and Odd Integers**
  - n is even $\leftrightarrow \exists k \in Z$ s.t. $n = 2k$
  - n is odd $\leftrightarrow \exists k \in Z$ s.t. $n = 2k + 1$
  - **Assumption 1**: Every integer is even or odd, but not both.
- **Divisibility**
  - $d|n \leftrightarrow \exists k \in Z$ s.t $n = dk$, $n, d \in Z, d \neq 0$.
- **Definition of Rational and Irrational**
  - r is rational $\leftrightarrow \exists a, b \in Z$ s.t. $r = \frac{a}{b}, b \neq 0$.

- r is not rational $\leftrightarrow r$ is irrational

- **Definition of Fraction in Lowest Term**
  - Largest integer that divides both a and b is 1.
  - **Assumption 2**: Every rational can be reduced to a fraction in lowest term.
- **Definition of Colorful**: $n = 3k$

1.3 Proofs

- **Direct Proof**
- **By Construction**
- **By Counterexample**
- **Exhaustion**
- **Contradiction**
- **Contraposition**
- **Mathematical Induction**

2 Logic of Compound Statements

2.1 Signs and Precedence

- $\sim$ not (negation)
- $\wedge$ and (conjunction)
- $\vee$ or (disjunction)
- $\oplus$ xor (exclusive or)
- **Precedence**: Not, then left to right, then conditionals.

2.2 More Terms

- **t**, **Tautology** (always true)
- **c**, **Contradiction** (always false)
- **Conditional** ($p \to q$, "If p then q")
  - **Converse** ($q \to p$, "if q then p")
  - **Inverse** ($\sim p \to \sim q$, "q only if p")

- **Contrapositive** ($\sim q \to \sim p$, "p only if q")

- **Biconditional** ($p \leftrightarrow q$, "p If and only if q")

2.3 Valid Arguments

- Valid: If and only if statements are all true $\to$ conclusion is true.
- **Critical Row**: All statements are true.
- If there is a critical row in which the conclusion is false, the argument form is **invalid**.
- If the conclusion in every critical row is true, then the argument form is **valid**.
- If there is no critical row, the argument is **valid**.

2.4 Properties of Argument

- **Modus Ponens**: $(p \to q) \wedge p \implies q$
- **Modus Tollens**: $(p \to q) \wedge \sim q \implies \sim p$
- **Generalisation**: $p \implies p \vee q$
- **Specialisation**: $p \wedge q \implies p$
- **Elimination**: $(p \vee q) \wedge \sim q \implies p$
- **Transitivity**: $(p \to q) \wedge (q \to r) \implies p \to r$
- **Division into cases**: $(p \vee q) \wedge (p \to r) \wedge (q \to r) \implies r$

2.5 Errors

- **Converse Error**: $(p \to q) \wedge q \implies p$ (Invalid)
- **Inverse Error**: $(p \to q) \wedge \sim p \implies \sim q$ (Invalid)

2.6 Soundness

- Argument is **valid** AND all premises are **true**.

2.7 Necessary and Sufficient

- r is **sufficient** for $s \leftrightarrow (r \to s)$, if r then s.
- r is **necessary** for $s \leftrightarrow (s \to r)$, if not r then not s.
- r is **necessary and sufficient** for $s \leftrightarrow (r \leftrightarrow s)$.

3 Logic of Quantified Statements

3.1 Definitions

- **Predicate**: A predicate contains a finite number of variables and becomes a statement when specific values are assigned to the variables.
- **Truth Set**: Set of elements that make $P(x)$ true.
- **Universal statement**: $\forall$
- **Existential statement**: $\exists$
- $\exists!$: There exists only 1.

3.2 Negation of Quantified Statements

- Negation of a Universal Statement: $\sim (\forall x P(x)) \equiv \exists x \sim P(x)$
- Negation of an Existential Statement: $\sim (\exists x P(x)) \equiv \forall x \sim P(x)$

3.3 Related Conditional Statements

- Contrapositive, Converse, Inverse definitions are the same as for non-quantified statements.
- Necessary, sufficient, Only if definitions are the same as for non-quantified statements.

4 Methods of Proof

4.1 Proof Methods

- Proof by construction
- Disproof by counterexample
- Proof by method of exhaustion
- Proof by generalising from the generic particular
- Proof by contradiction
- Proof by contraposition

4.2 Definitions

- **Prime**: n is prime $\leftrightarrow (n > 1) \wedge (\forall r, s \in Z((r > 1) \wedge (s > 1) \rightarrow rs \neq n))$.
- **Composite**: Composite $\rightarrow$ not prime.

5 Set Theory

5.1 Notations and Definitions

- **Set Roster Notation**: $\{1, 2, 3, \ldots\}$
- **Membership**: $\in$
- **Cardinality**: $|S| = \#$ of elements
- **Set Builder Notation**: $\{x \in U : P(x)\}$
- **Replacement Notation**: $\{P(x) : x \in A\}$
- **Subset**: $A \subseteq B$ if and only if $x \in A \rightarrow x \in B$ (B is a superset of A).
  - The empty set ($\emptyset$) is a subset of every set.
  - **Proper Subset**: $A \subset B$ if $A \subseteq B$ and $A \neq B$.
- **Ordered Pair**: (a, b)
- **Cartesian Product**: $A \times B = \{(a, b) : a \in A \wedge b \in B\}$
- **Set Equality**: $A = B \Leftrightarrow A \subseteq B \wedge B \subseteq A \Leftrightarrow \forall x(x \in A \Leftrightarrow x \in B)$.

5.2 Set Operations

- **Union**: $A \cup B = \{x \in U : x \in A \vee x \in B\}$
- **Intersection**: $A \cap B = \{x \in U : x \in A \wedge x \in B\}$
- **Difference**: $B \setminus A = \{x \in U : x \in B \wedge x \notin A\}$
- **Complement**: $A^c = \{x \in U : x \notin A\}$
- **Disjoint**: $A \cap B = \emptyset$.
- **Mutually Disjoint**: True for all sets in a collection.
- **Power Set**: Set of all subsets of A, denoted $\mathcal{P}(A)$.
  - Cardinality of Power Set of Finite Set: $|\mathcal{P}(A)| = 2^{|A|}$.

5.3 Partition

- A collection of non-empty subsets of A forms a partition if they are mutually disjoint and their union is A.
- $\forall x \in A \exists ! S \in \mathcal{C}(x \in S)$, where $\mathcal{C}$ is the collection of subsets.

5.4 Set Identities and Properties

- **Subset Relations** (Inclusion of Intersection, Inclusion in Union, Transitive Property of Subsets)
- $A \subseteq B \iff A \cup B = B$
- $A \oplus B = (A \cup B) \setminus (A \cap B)$ (**Symmetric Difference**)

6 Relations

6.1 Definition

- A relation R from set A to set B is a subset of $A \times B$.
- x is related to y by R, written xRy, iff $(x, y) \in R$.
- **Domain**, **Codomain**, and **Range** are defined as usual.

6.2 Relation Operations

- **Inverse of Relation**: $R^{-1} = \{(y, x) \in B \times A : (x, y) \in R\}$.
- **Composition of Relations**: $S \circ R$: $\forall x \in A, \forall z \in C(x(S \circ R)z \Leftrightarrow \exists y \in B(xRy \wedge ySz))$.
- Composition is **Associative**.

6.3 Properties of Relations

Let R be a relation on set A.

- **Reflexive**: $\forall x \in A(xRx)$.
- **Symmetric**: $\forall x, y \in A(xRy \implies yRx)$. R is symmetric $\leftrightarrow R = R^{-1}$.
- **Transitive**: $\forall x, y, z \in A(xRy \wedge yRz \implies xRz)$.
- **Antisymmetric**: $\forall x, y \in A(xRy \wedge yRx \implies x = y)$.
- **Asymmetric**: $\forall x, y \in A(xRy \implies \sim (yRx))$. Asymmetry $\rightarrow$ Antisymmetry.

6.4 Equivalence Relation

- A relation that is **Reflexive**, **Symmetric**, and **Transitive**.
- **Equivalence Class**: $[a]_\sim = \{\text{all elements that are } \sim \text{ related to } a\}$.
- $x \sim y \Leftrightarrow [x] = [y] \Leftrightarrow [x] \cap [y] \neq \emptyset$.

- **Congruence**: $a \equiv b \pmod{n} \iff n|(a-b)$. This is an equivalence relation.

6.5 Partial Order Relation

- A relation that is **Reflexive**, **Antisymmetric**, and **Transitive**.
- **Common Partial Order**: Subset relation ($\subseteq$).
- **Comparability**: x and y are comparable if $x \preceq y$ or $y \preceq x$ (where $\preceq$ denotes the partial order relation).
- **Total Order (Linear Order)**: A partial order where every two elements are comparable.
- **Well Ordered Set**: Every non-empty subset of A contains a smallest element. $\forall S \in \mathcal{P}(A), S \neq \emptyset \implies \exists x \in S \forall y \in S(x \preceq y)$.

6.6 Order Extremal Elements

Let A be a set with partial order $\preceq$.

- **Maximal Element** c: $\forall x \in A, c \preceq x \to c = x$.
- **Minimal Element** c: $\forall x \in A, x \preceq c \to c = x$.
- **Largest Element** c (Greatest Element): $\forall x \in A, x \preceq c$. (Must be comparable to everything).
- **Smallest Element** c (Least Element): $\forall x \in A, c \preceq x$. (Must be comparable to everything).
- All largest/smallest elements are also maximal/minimal.

7 Functions

7.1 Definition

A function $f : X \to Y$ is a relation that satisfies:

- (F1) $\forall x \in X \exists y \in Y (x, y) \in f$. (Every element in the domain has an image).
- (F2) $\forall x \in X \forall y_1, y_2 \in Y ((x, y_1) \in f \land (x, y_2) \in f \to y_1 = y_2)$. (The image is unique).
- Equivalently, $\forall x \in X \exists! y \in Y (x, y) \in f$.

7.2 Terminology

- X: **Domain**

- Y: **Codomain**
- x: **Argument**, **Input**, **Preimage** of $f(x)$
- $f(x)$: **Image** of x, **Output** of f for input x
- **Range** (Image of X): $f(X) = \{y \in Y : \exists x \in X(y = f(x))\}$. A subset of the Codomain.

7.3 Setwise Image and Preimage

- **Setwise Image**: If $A \subseteq X$, $f(A) = \{f(x) : x \in A\}$.
- **Setwise Preimage**: If $B \subseteq Y$, $f^{-1}(B) = \{x \in X : f(x) \in B\}$.

7.4 Function Equality

f and g are equal if:

- Domains are the same.
- Codomains are the same.
- $f(x) = g(x)$ for all x in the Domain.

7.5 Properties of Functions

Let $f : X \to Y$.

- **Injective** (One-to-One): $\forall x_1, x_2 \in X(f(x_1) = f(x_2) \implies x_1 = x_2)$.
  - Not Injective: $\exists x_1, x_2 \in X(f(x_1) = f(x_2) \wedge x_1 \neq x_2)$.
  - A function is injective iff it has a left inverse.
- **Surjective** (Onto): $\forall y \in Y \exists x \in X(y = f(x))$. (Range equals Codomain).
  - Not Surjective: $\exists y \in Y \forall x \in X(y \neq f(x))$.
  - A function is surjective iff it has a right inverse.
- **Bijective**: Injective and Surjective. $\forall y \in Y \exists! x \in X(y = f(x))$.

7.6 Inverse Function and Composition

- **Inverse Function** $g = f^{-1}$: $\forall x \in X \forall y \in Y(y = f(x) \Leftrightarrow x = g(y))$.
- f^{-1} is unique.
- f is bijective iff f^{-1} exists.

- **Composition**: $g \circ f(x) = g(f(x))$ for all $x \in X$.
- **Identity Function**: $\text{id}_X(x) = x$ for all $x \in X$. $f \circ \text{id}_X = f$ and $\text{id}_Y \circ f = f$.
- If f is bijective, $f^{-1} \circ f = \text{id}_X$ and $f \circ f^{-1} = \text{id}_Y$.
- Composition is **associative** but not **commutative**.
- If f and g are injective, then $g \circ f$ is injective.
- If g is injective and $g \circ f$ is injective, then f is injective.
- If f and g are surjective, then $g \circ f$ is surjective.
- $(g \circ f)^{-1} = f^{-1} \circ g^{-1}$.

7.7 Sequences and Strings

- **Sequence**: $a(n) = a_n$ for every $n \in Z_{\geq 0}$.
- **Fibonacci Sequence**: $F_0 = 0, F_1 = 1$, and $F_{n+2} = F_{n+1} + F_n$ for $n \in Z_{\geq 0}$.
- **String**: A word over set A.
- **Empty String**: ϵ (Length 0).
- Two sequences are equal if $a(n) = b(n)$ for all n.
- Two strings are equal if $a_i = b_i$ for all i from 0 to length $- 1$.

8 Mathematical Induction

8.1 Notations

- **Summation Notation**: $\sum_{k=m}^{n} a_k$
- **Product Notation**: $\prod_{k=m}^{n} a_k$
- $\sum a_k + \sum b_k = \sum (a_k + b_k)$
- $c \prod a_k = \prod c a_k$ (Generalised Distributive law, if c is a constant multiplier outside the product)
- $\prod a_k \times \prod b_k = \prod (a_k \times b_k)$

8.2 Mathematical Induction (MI) Format

- **Base Case**: Prove $P(a)$ is true.
- **Inductive Hypothesis**: Assume for all $k \geq a$, $P(k)$ is true.

- **Inductive Step**: Prove $P(k+1)$ is true using the Inductive Hypothesis.
- **Conclusion**: Then for all $n \geq a$, $P(n)$ is true.

8.3 Strong Mathematical Induction

- **Inductive Hypothesis**: Assume $P(i)$ is true for all $a \leq i \leq k$.

8.4 Sequences

- **Arithmetic Progression (AP)**: $a_n = a_0 + dn$.
- **Geometric Progression (GP)**: $a_n = a_0 r^n$.
- **Closed Form**: No summation or ellipsis $(\ldots)$.
- **Recurrence Relation**: Each sequence term is based on previous terms (e.g., Fibonacci).

9 Cardinality

9.1 Definitions

- **Finite**: S is empty or there is a bijection from S to $Z_n = \{1, 2, \ldots, n\}$.
- **Infinite**: If not finite. A set A is infinite if there exists a set B such that $B \subset A$, $B \neq A$, and $|B| = |A|$.
- **Cardinality**: $|S|$.
- $\aleph_0 = |Z^+|$.
- **Countably Infinite**: Same cardinality as $\aleph_0$.
- **Countable**: Finite or Countably Infinite.
- **Uncountable**: Not countable (e.g., R).

9.2 Pigeonhole Principle (PHP)

- If $f : A \to B$ is injective, then $|A| \leq |B|$.
- **Dual PHP**: If $f : A \to B$ is surjective, then $|A| \geq |B|$.
- If f is injective and surjective, $|A| = |B|$ (bijective).
- If $f : X \to Y$ and $|X| = n, |Y| = m$. If $\lfloor \frac{n-1}{m} \rfloor < k$, then there is some $y \in Y$ such that y is the image of at least $k+1$ distinct elements of X.

- Equivalently, if for each $y \in Y$, $|f^{-1}(\{y\})| \leq k$, then $|X| \leq km$.

9.3 Properties of Cardinality

- If A and B are both countably infinite, then $A \times B$ is countable. (Works for n countably infinite sets too).
- $A \cup B$ is countable if A and B are countable.
- Any subset of a countable set is countable.
- Any set with an uncountable subset is uncountable.
- Every infinite set has a countably infinite subset.
- If A is countably infinite, $\mathcal{P}(A)$ is uncountable.

10 Counting

10.1 Basic Probability

- **Sample Space** (S): Set of all possible outcomes.
- **Event** (E): A subset of the sample space.
- $P(E) = \frac{|E|}{|S|}$ (for equally likely outcomes).

10.2 Counting Principles

- Number of integers between m to n inclusive is $n - m + 1$.
- **Multiplication Rule**: If an operation consists of k steps with n_i options for step i, total options is $n_1 \times n_2 \times \ldots \times n_k$.
- **Permutations** of a set of n elements: $n!$.
- r-**Permutations**: $P(n, r) = \frac{n!}{(n-r)!}$.
- r-**Combinations**: $C(n, r) = \frac{P(n,r)}{r!} = \frac{n!}{r!(n-r)!}$.
- **Combinations with Repetition** (Stars and Bars): The number of ways r objects can be selected from n categories is $C(r + n - 1, r)$.

10.3 Set Counting Formulas

Assuming A_i are mutually disjoint:

- **Addition Rule**: $|A| = |A_1| + |A_2| + \ldots$.

- $|A \setminus B| = |A| - |A \cap B|$.
- **Inclusion-Exclusion Principle**:
  - $|A \cup B| = |A| + |B| - |A \cap B|$.
  - $|A \cup B \cup C| = |A| + |B| + |C| - (|A \cap B| + |A \cap C| + |B \cap C|) + |A \cap B \cap C|$.
  - Generalised: $\sum$ (single) - $\sum$ (double) + $\sum$ (triple) - ...

10.4 Properties of $C(n, r)$

- $C(n+1, r) = C(n, r-1) + C(n, r)$ (for $r \leq n$).
- $C(n, r) = C(n, n-r)$.
- **Binomial Theorem**: $(a+b)^n = \sum_{k=0}^{n} C(n, k) a^{n-k} b^k$.

10.5 Conditional Probability and Independence

- **Probability Axioms**: $0 \leq P(A) \leq 1$; $P(\emptyset) = 0$ and $P(S) = 1$; If $A \cap B = \emptyset$, then $P(A \cup B) = P(A) + P(B)$.
- $P(A \cup B) = P(A) + P(B) - P(A \cap B)$.
- **Conditional Probability**: $P(B|A) = \frac{P(A \cap B)}{P(A)}$.
- $P(A \cap B) = P(B|A)P(A)$.
- **Independence**: $P(A \cap B) = P(A)P(B)$.
- **Mutual Independence** (for multiple sets): $P(A_1 \cap A_2 \cap \ldots) = P(A_1)P(A_2)\ldots$.
- **Expected Value**.
- **Bayes' Theorem**.

11 Graphs

11.1 Definitions

- **Graph** $G = (V, E)$, where V are **Vertices** (Points, Nodes) and E are **Edges** (Lines connecting vertices).
- **Undirected Edge**: $e = \{v_1, v_2\}$ (Set).
- **Directed Edge**: $e = (v_1, v_2)$ (Ordered pair).
- **Simple Graph**: No loops (self-related) and no parallel edges (double related).

- **Complete Graph** (K_n): Simple graph with n vertices, where every pair of distinct vertices is connected by an edge. Has $C(n, 2)$ edges.
- **Bipartite Graph**: Vertex set V can be partitioned into two disjoint and independent sets, V_1 and V_2, such that every edge connects a vertex in V_1 to one in V_2.
- **Complete Bipartite Graph** ($K_{m,n}$): Every vertex in V_1 is connected to every vertex in V_2.
- **Subgraph**: Subset of vertices and edges of the original graph, with the same endpoint relations for the edges.

11.2 Degree

- **Degree of a Vertex** ($\deg(v)$): The number of edges incident to v (loops count twice).
- **Total Degree of Graph**: Sum of degrees of each vertex.
- **Handshake Theorem**: Total Degree of $G = 2 \times |E|$.
- Total Degree is **Even**.
- There is an even number of vertices with **odd degrees**.
- **Directed Graph**:
  - **In-degree** ($\deg^-(v)$): Number of edges going into v.
  - **Out-degree** ($\deg^+(v)$): Number of edges going out of v.

11.3 Traversal

- **Walk**: $v_0 e_1 v_1 e_2 v_2 \ldots e_n v_n$. n is the length.
- **Trail**: Walk with no repeated edges.
- **Path**: Trail with no repeated vertices.
- **Closed Walk**: Walk that starts and ends with the same vertex ($v_0 = v_n$).
- **Circuit**: Closed walk of length ≥ 3 and no repeated edges.
- **Simple Circuit**: Circuit with no repeated vertices other than $v_0 = v_n$.

11.4 Connectivity and Cycles

- **Connected**: If there is a walk (or path) from every vertex to every other vertex.
- **Connected Component**: A maximal connected subgraph of G.

- **Cyclic**: If there exists a loop or a cycle (simple circuit).
- **Acyclic**: Otherwise.

11.5 Euler and Hamiltonian

- **Euler Circuit**: A circuit that contains every vertex and traverses each edge exactly once.
- **Eulerian Graph**: Contains an Euler Circuit.
- **Conditions for Euler Circuit**: G is connected AND every vertex has an even degree.
- **Euler Trail**: A trail that contains every vertex and traverses each edge exactly once.
- **Conditions for Euler Trail**: G is connected AND has exactly zero or two vertices of odd degree (starts/ends at the odd degree vertices).
- **Hamiltonian Circuit**: A simple circuit that visits every vertex exactly once (except the start/end vertex).
- **Travelling Salesman Problem**: Finding the shortest Hamiltonian circuit in a weighted graph.
- All complete graphs K_n, where $n > 2$, contain a Hamiltonian circuit.

11.6 Adjacency Matrix

- A_{ij} is the number of ways to go from vertex i to vertex j in one step.
- A is symmetric if the graph is undirected.
- $(A^n)_{ij}$ is the number of walks from i to j in n steps.

11.7 Isomorphism

- Two graphs are **isomorphic** if they look the same (there exists a bijection between their vertex sets and between their edge sets that preserves incidence).
- The set induced by the relation of isomorphic graphs is an equivalence relation.

11.8 Planar Graphs

- A graph is **planar** if it can be drawn on a 2D plane without edges crossing.
- **Euler's Formula** (for connected planar graphs): $|V| - |E| + |F| = 2$, where $|F|$ is the number of faces (including the outside face).

- **Kuratowski's Theorem**: A graph is planar iff it does not contain a subgraph that is a subdivision of K_5 or $K_{3,3}$.

12 Trees

12.1 Definitions and Properties

- **Tree**: A connected, acyclic (no circuits) graph.
- **Trivial Tree**: One vertex.
- **Forest**: Circuit-free but not necessarily connected (a collection of trees).
- **Leaf**: A vertex of degree 1.
- **Internal Vertex**: A vertex that is not a leaf.
- A tree with n vertices has $n - 1$ edges.
- A tree with $n > 2$ vertices has at least 2 vertices with degree 1.
- If a graph G has n vertices and $n - 1$ edges and is connected, then G is a tree.

12.2 Binary Trees

- **Rooted Tree**: A tree where one vertex is designated as the root.
- **Binary Tree**: A rooted tree where every parent has at most two children (at most one left and one right).
- **Full Binary Tree**: A binary tree where every parent has exactly two children. Arithmetic expressions can be represented by a Full Binary Tree.
- A full binary tree with k internal vertices has $k + 1$ leaves.
- If h is the height, the maximum number of leaves is 2^h, so $\log_2(\text{leaves}) \leq h$.

12.3 Tree Traversal

- **Breadth-First Search (BFS)**: Visit all neighbours before moving to the next level.
- **Depth-First Search (DFS)**:
  - **Pre-Order**: Root $\rightarrow$ Left $\rightarrow$ Right.
  - **In-Order**: Left $\rightarrow$ Root $\rightarrow$ Right.
  - **Post-Order**: Left $\rightarrow$ Right $\rightarrow$ Root.

12.4 Spanning Trees

- **Spanning Tree**: A subgraph of a connected graph G that includes every vertex of G and is a tree. It uses the minimal amount of edges to stay connected.
- Every connected graph has a spanning tree.
- Any two spanning trees for a graph have the same number of edges ($n - 1$).
- Number of spanning trees in a complete graph K_n is n^{n-2} (Cayley's Formula, for $n > 2$).

12.5 Algorithms

- **Kruskal's Algorithm** (Minimum Spanning Tree):
  1. Find edge with lowest weight.
  2. If it does not create a circuit, add it.
  3. End when $n - 1$ edges are added.
- **Prim's Algorithm** (Minimum Spanning Tree):
  1. Start from an arbitrary vertex.
  2. Add the edge with the lowest weight that connects a vertex not yet in the growing tree to a vertex in the growing tree.
- **Dijkstra's Algorithm** (Shortest Path):
  1. From the starting node, find the shortest path to all adjacent unvisited nodes.
  2. Select the unvisited node with the shortest known distance as the new starting node.