

LAPT

LONDON ACADEMY OF PROFESSIONAL TRAINING

CERTIFICATE

Logic & Computational Thinking

Workshop Level



LAPT — London Academy of Professional Training

Address 85 Great Portland Street, W1W 7LT, London, United Kingdom

Email info@lapt.org

Telephone +44 7513 283044

Web lapt.org

Reference IT-SDA-W • IT Centres

UKPRN 10067351 • Registered in England and Wales, company 4984798

AT A GLANCE

REFERENCE	IT-SDA-W
AWARD	Certificate
ASSESSMENT	440 marks — 264 to pass (60%)
PREREQUISITES	None
VALIDITY	Lifetime
AWARDING BODY	LAPT — London Academy of Professional Training

CURRICULUM

1 Basic Algorithms 3 chapters · 12 classes 65 marks	
• 1 Fundamentals of Basic Algorithms — 4 classes › 1.1 Understanding Basic Algorithm Concepts › 1.2 Exploring Simple Sequence Structures › 1.3 Implementing Conditional Logic in Algorithms › 1.4 Building Iterations for Problem Solving • 2 Algorithm Design and Efficiency — 4 classes › 2.1 Understanding Basic Algorithm Structures › 2.2 Analyzing Algorithm Efficiency: Time and Space Complexities › 2.3 Designing Simple Algorithms: Step-by-Step Approach › 2.4 Improving Algorithm Performance through Optimization	• 3 Implementing and Adapting Basic Algorithms — 4 classes › 3.1 Understanding Algorithm Structures › 3.2 Implementing Simple Algorithms › 3.3 Adapting Algorithms for Different Scenarios › 3.4 Testing and Optimizing Algorithm Efficiency
2 Computational Systems Analysis 3 chapters · 12 classes 65 marks	
• 1 Understanding Computational System Structures and Components — 4 classes › 1.1 Identifying Key Components of Computational Systems › 1.2 Analyzing the Structure of System Architectures › 1.3 Exploring Interactions Between System Components › 1.4 Evaluating System Performance and Optimization • 2 Analyzing System Operations and Data Flow — 4 classes › 2.1 Understanding System Components and Functions › 2.2 Mapping Data Flow in Computational Systems › 2.3 Identifying Operational Challenges in Systems › 2.4 Applying Data Flow Analysis to Optimize Performance	• 3 Applying Analytical Tools and Techniques for System Optimization — 4 classes › 3.1 Understanding Key Analytical Tools for System Optimization › 3.2 Exploring Data Flow Analysis Techniques › 3.3 Implementing Optimization Algorithms in Systems › 3.4 Evaluating System Performance Post-Optimization

3

Data Structures

3 chapters · 12 classes **65 marks**

• 1 Chapter 1 — Understanding Basic Data Structures — 4 classes

- › 1.1 Exploring Fundamental Data Structures
- › 1.2 Comparing Arrays and Linked Lists
- › 1.3 Implementing Stacks and Queues
- › 1.4 Analyzing Performance in Data Structure Operations

• 2 Chapter 2 — Complex Data Structures and Their Uses — 4 classes

- › 2.1 Understanding the Characteristics of Complex Data Structures
- › 2.2 Exploring Applications of Trees and Graphs
- › 2.3 Implementing Hash Tables in Real-World Scenarios
- › 2.4 Analyzing Performance and Efficiency of Complex Data Structures

• 3 Chapter 3 — Data Structures in Software Development — 4 classes

- › 3.1 Understand the Role of Data Structures in Software Development
- › 3.2 Explore Common Data Structures: Arrays, Lists, and Maps
- › 3.3 Implement Data Structures in Programming: Hands-On Exercises
- › 3.4 Analyze the Efficiency of Data Structures in Real-World Applications

4

Introduction to Logical Reasoning

3 chapters · 12 classes **90 marks**

• 1 Chapter 1 — Fundamentals of Propositional Logic — 4 classes

- › 1.1 Understanding Basic Propositions and Their Components
- › 1.2 Exploring Logical Connectives: AND, OR, NOT
- › 1.3 Constructing Truth Tables and Evaluating Statements
- › 1.4 Applying Propositional Logic in Real-World Scenarios

• 2 Chapter 2 — Techniques for Logical Deduction — 4 classes

- › 2.1 Understanding Logical Statements
- › 2.2 Identifying Premises and Conclusions
- › 2.3 Applying Deductive Reasoning
- › 2.4 Solving Problems with Logical Deduction

• 3 Chapter 3 — Introduction to Predicate Logic and Applications — 4 classes

- › 3.1 Understanding the Basics of Predicate Logic
- › 3.2 Exploring Quantifiers and Their Roles
- › 3.3 Translating English Sentences into Predicate Logic
- › 3.4 Applying Predicate Logic to Real-World Scenarios

5

Logical Problem Solving

3 chapters · 12 classes **90 marks**

• 1 Understanding Fundamental Logical Concepts — 4 classes

- › 1.1 Grasping Basic Logical Operators
- › 1.2 Applying Truth Tables in Problem Solving
- › 1.3 Analyzing Logical Statements and Arguments
- › 1.4 Implementing Logic in Computational Thinking

• 2 Building Structured Problem-Solving Steps — 4 classes

- › 2.1 Understanding the Problem: Defining Objectives and Constraints
- › 2.2 Breaking Down the Problem: Creating Sub-Tasks
- › 2.3 Planning Solutions: Developing Logical Sequences
- › 2.4 Evaluating Outcomes: Testing and Refining Solutions

• 3 Applying Logic to Real-World Computational Problems — 4 classes

- › 3.1 Understanding Logic in Everyday Contexts
- › 3.2 Analyzing Computational Problems with Logical Frameworks
- › 3.3 Creating Logical Models for Problem Solving
- › 3.4 Applying Logical Models to Real-World Scenarios

6

Practical Applications

3 chapters · 12 classes **65 marks**

• 1 Chapter 1 — Problem Solving with Logical Deduction — 4 classes

- › 1.1 Understanding Logical Deduction: Recognizing Patterns
- › 1.2 Analyzing Problems: Breaking Down Scenarios
- › 1.3 Applying Deductive Reasoning: Solving Real-World Problems
- › 1.4 Evaluating Outcomes: Reflecting on Problem-Solving Strategies

• 2 Chapter 2 — Algorithm Design and Analysis — 4 classes

- › 2.1 Understanding Algorithm Basics
- › 2.2 Constructing Simple Algorithms
- › 2.3 Analyzing Algorithm Efficiency
- › 2.4 Applying Algorithms to Real-World Problems

• 3 Chapter 3 — Implementing Computational Solutions — 4 classes

- › 3.1 Understanding Problem Decomposition
- › 3.2 Designing Algorithms with Flowcharts
- › 3.3 Translating Algorithms into Pseudocode
- › 3.4 Testing and Debugging Computational Solutions