KOENIG
step forward

KOENIG SOLUTIONS · COURSE OUTLINE

Python DSA: Trees, Graphs & Sorting Algorithms

Learn to design, implement and analyse core data structures and algorithms in Python, from arrays and trees to graphs and sorting.

5 Days	16	89
DURATION	MODULES	TOPICS

COURSE OVERVIEW

This course offers a comprehensive, definitive introduction to data structures and algorithms, including their design, analysis, and implementation in Python. Utilizing a consistent object-oriented viewpoint throughout the book, it provides detailed algorithmic strategies for producing efficient realizations of common data structures such as arrays, stacks, queues, linked lists, trees, maps, hash tables, search trees, and graphs. This course also provides an in-depth analysis of algorithmic performance that helps readers to recognize common trade-offs between competing strategies. This course incorporates a host of pedagogical features, including illustrations, code fragments, and end-of-Module exercises.

WHO THIS COURSE IS FOR

- **Python Developers**
- **Software Engineers**
- **Computer Science Students**
- **Algorithm Designers**
- **Backend Developers**

LEARNING OBJECTIVES

- Explain object-oriented programming concepts and class design in Python
- Analyse algorithm efficiency using asymptotic notation techniques
- Apply recursion to solve and analyse computational problems
- Implement stacks, queues and linked lists as abstract data types
- Build tree structures and traverse them using standard algorithms
- Implement priority queues, heaps and hash-based maps and sets
- Compare balanced search trees such as AVL, splay and red-black trees
- Evaluate sorting algorithms such as merge-sort and quick-sort

COURSE SYLLABUS

16 Modules, 89 Topics

01

Introduction to Python

TOPICS

- Python Overview
- Objects in Python
- Expressions, Operators, and Precedence
- Control Flow
- Functions
- Simple Input and Output
- Exception Handling
- Iterators and Generators
- Additional Python Conveniences
- Scopes and Namespaces
- Modules and the Import Statement

02

Object-Oriented Programming

TOPICS

Goals, Principles, and Patterns · Software Development · Class Definitions · Inheritance · Namespaces and Object-Orientation · Shallow and Deep Copying

03

Introduction to Data Structures and Algorithms

TOPICS

Data Structures · Experimental Studies · The Seven Functions Used in this Book · Analysis of Algorithms · Simple Justification Techniques

04

Recursion

TOPICS

Examples Illustrating Recursion · Analysing Recursive Algorithms · Further Examples of Recursion · Designing Recursive Algorithms · Eliminating Tail Recursion

05

Array-Based Sequences

TOPICS

Python's Sequence Types · Low-Level Arrays · Dynamic Arrays and Amortization · Efficiency of Python's Sequence Types · Using Array-Based Sequences · Multidimensional Data Sets

06

Stacks

TOPICS

Stacks · The Stack Abstract Data Type · Simple Array-Based Stack Implementation

07

Queues

TOPICS

Queues · The Queue Abstract Data Type · Array-Based Queue Implementation · Double-Ended Queues · Circular Queues

08

Linked Lists

TOPICS

Singly Linked Lists · Circularly Linked Lists · Doubly Linked Lists · The Positional List ADT · Sorting a Positional List · Link-Based vs. Array-Based Sequences

09

Trees

TOPICS

General Trees · Binary Trees · Implementing Trees · Tree Traversal Algorithms · **Case Study:** An Expression Tree

10

Priority Queues

TOPICS

The Priority Queue Abstract Data Type · Implementing a Priority Queue · Heaps · Adaptable Priority Queues

11

Maps, Hash Tables, and Sets

TOPICS

Maps and Dictionaries · Hash Tables · Sorted Maps · Sets, Multisets, and Multimaps

12

Search Trees

TOPICS

Binary Search Trees · Balanced Search Trees · AVL Trees · Splay Trees · (2, 4) Trees · Red-Black Trees

13

Sorting Algorithms

TOPICS

Why Study Sorting Algorithms? · Merge-Sort · Quick-Sort · Studying Sorting through an Algorithmic Lens · Sorting with a Priority Queue · Comparing Sorting Algorithms · Python's Built-In Sorting Functions

14

Graph Algorithms

TOPICS

Graphs · Data Structures for Graphs · Graph Traversals · Transitive Closure · Directed Acyclic Graphs · Shortest Paths · Minimum Spanning Trees

15

Text Processing

TOPICS

Abundance of Digitized Text · Pattern-Matching Algorithms · Dynamic Programming · Text Compression and the Greedy Method · Tries

16

Memory Management and B-Trees

TOPICS

Memory Management · Memory Hierarchies and Caching · External Searching and B-Trees · External-Memory Sorting

WHAT IS INCLUDED

Everything You Need to Succeed

- Koenig course material for every module
- Delivery by certified instructors and subject matter experts
- Hands-on exercises throughout the course
- Course completion certificate from Koenig

ABOUT KOENIG

Who We Are

Established in 1993, Koenig Solutions is a globally recognised training provider with 35+ technology partnerships, 5,000+ training programmes and a network of certified instructors and subject matter experts, supporting individuals and enterprise teams worldwide.

FLEXIBLE DELIVERY

Learn the Way That Suits You

Classroom Training | Live Online Classes (ILO) | Fly-Me-A-Trainer (FMAT) | Flexi (Self-Paced) | 1-on-1 Training

Ready to Get Started?

Talk to a learning advisor about schedules, group training and customised delivery.

Visit koenig-solutions.com | info@koenig-solutions.com