Data Structures with C++

Worksheets

Worksheets – Data Structures with C++

01 – Introduction

01 – Introduction – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

02 – Arrays

02 – Arrays – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

03 – Strings

03 – Strings – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).

2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

04 – Singly Linked List

04 – Singly Linked List – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

05 – Doubly & Circular Linked List

05 – Doubly & Circular Linked List – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

06 – Stack

06 – Stack – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.

2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

07 – Queue

07 – Queue – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

08 – Recursion

08 – Recursion – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

09 – Binary Trees & BST

09 – Binary Trees & BST – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

10 – Advanced Trees (AVL, Heap)

10 – Advanced Trees (AVL, Heap) – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

11 – Graph Basics

11 – Graph Basics – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

12 – Graph Algorithms

12 – Graph Algorithms – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

13 – Hashing

13 – Hashing – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises
1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge
- Build a tiny application using this week’s DS/algorithms.

14 – Review & Practice

14 – Review & Practice – Worksheet

Part A – Quick Questions
1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises

1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge

- Build a tiny application using this week’s DS/algorithms.

15 – Sorting & Searching

15 – Sorting & Searching – Worksheet

Part A – Quick Questions

1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises

1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge

- Build a tiny application using this week’s DS/algorithms.

16 – Applications & Final Project

16 – Applications & Final Project – Worksheet

Part A – Quick Questions

1. Define the main concept of this week’s topic and one real-world analogy.
2. List two advantages and two disadvantages.
3. State the typical time complexity of the core operations.

Part B – Coding Exercises

1. Implement create/insert/delete for the core structure (or a key algorithm demo).
2. Implement a search (or traversal) and print routine.
3. Provide a small menu-driven CLI wrapper.

Part C – Mini Challenge

- Build a tiny application using this week’s DS/algorithms.