Syllabus and Readings

Required Reading

The following table outlines the reading you are expected to do for this course. You do not have to read every single example in a chapter if the first example clarifies the concept for you. Even though you are required to read a chapter, we may not cover all of the topics at length in class. That does not excuse you from doing the reading, as the material might appear on an exam.

Topic and Reading	Week
Chapter 1 Principles of Programming and Software Engineering Problem Solving and Software Engineering Achieving a Better Solution A Summary of Key Issues in Programming	Week 1
Chapter 2 Recursion: The Mirrors Recursive Solutions Counting Things Searching an Array Organizing Data Recursion and Efficiency	Week 2
Chapter 3 Data Abstraction: The Walls Abstract Data Types Specifying ADTs Implementing ADTs	Week 3
Chapter 4 Linked Lists Preliminaries Programming with Linked Lists Variations of the Linked List Templates and the Standard Template Library	Weeks 4-5
Chapter 5 Recursion as a Problem-Solving Technique Backtracking Defining languages Mathematical induction revisited	Week 6
Chapter 6 Stacks The Abstract Data Type Stack Simple Applications of the ADT Stack Implementations of the ADT Stack Application: Algebraic Expressions	Weeks 7 -8
Chapter 7 Queues The Abstract Data Type Queue	Weeks 8-9

Topic and Reading	Week
Simple Applications of the ADT Queue Implementations of the ADT Queue A Summary of Position-Oriented ADTs	
Chapter 8 Class Relationships Inheritance Virtual Functions and Late Binding Friends The ADTs List and Sorted List Revisited Overloaded Operators	Weeks 9-10
Chapter 9 Algorithm Efficiency And Sorting Measuring the Efficiency of Algorithms Sorting Algorithms and Their Efficiency A Comparison of Sorting Algorithms	Weeks 11-12
Chapter 10 Trees Terminology The ADT Binary Tree The ADT Binary Search Tree General Trees	Weeks 13-14