CIS 22C.42Z: CRN: 13624, Summer 2025, Synchronous
CIS 22C.63Z: CRN: 13613, Summer 2025, Asynchronous
Data Abstraction and Structures
Credit-Degree Applicable
4.5 quarter units (3 semester units)
June 30, 2025 – August 08, 2025

Class 13624 meets Online synchronously, 3:00 – 3:50 pm, Mon, Wed, while 13613 is asynchronous, and the lectures are also recorded.
Class 13613 can also join the live lectures/recordings if they wish.
Classes are recorded at 3:00 pm – 3:50 (PST) Mon, Wed.
Office Hours in Zoom: Mon, Wed, 2:00 pm – 3:00 pm (PST), online, one for each section.
Instructor: Hussein Al-Hussein, Ph.D. (MS, Ph.D. from Stanford University)
Email: alhussein@fhda.edu
Course Registration Number (CRN): 13624 Synchronous, 13613 Asynchronous
Section: 42Z synchronous, 63Z asynchronous
Units: 4.5 quarter units (3-semester units)
Classes are given and also recorded at 3:30 pm – 5:20 (PST) Wed.
Join lectures Zoom meeting at: Link TBA
Meeting ID: TBA; no passcode

Lab: Online; TBA
Duration: 06/30/2025 – 08/08/2025

Office Hours via Zoom: Wed, 02:30 pm – 03:30 pm (PST), online
Join Office Hours Zoom Meeting: Link TBA.
Meeting ID: TBA; no passcode

Instructor: Hussein Al-Hussein, Ph.D. (MS, Ph.D. from Stanford University)
Email: alhussein@fhda.edu

Textbook:
zyBook: CIS 22C: Data Abstraction and Structures
zyBook code: DEANZACIS22CAl-HusseinSummer2025
zyBook ISBN: 979-8-203-99194-2

Instructions for students

Students must access zyBooks through links in Canvas

1. Click any zyBooks assignment link in your learning management system (Do not go to the zyBooks website and create a new account)
1. Subscribe

*A subscription is **$49.95**. Students may begin subscribing on Jun 17, 2025, and the cutoff to subscribe is Aug 09, 2025. Subscriptions will last until Aug 24, 2024.*

Description from Catalog:

Application of software engineering techniques to the design and development of large programs; data abstraction and structures and associated algorithms: stacks, queues, linked lists, trees, graphs, and hash tables; internal and external sorting; use of recursion; team project.

Student Learning Outcome Statements (SLO):

- Read, analyze, and explain advanced data structures programs.
- Design solutions for advanced problems using appropriate design methodology incorporating advanced data structures programming constructs.
- Create and analyze the efficiency of advanced-level data structures algorithms, code, document, debug, and test advanced data structures programs using multiple source and header files.

Advisory preparation:

- CIS 22B or CIS 35A.
- Advisory: Mathematics 212 or equivalent.

Work Required: 15 hours per week

Grading:

- ZyBooks & Labs: 40%
- Midterm: 30%
- Final: 30%

Grade average required:

- A+: 98-100·
- A: 92-97
- A-: 90-91
- B+: 88-89
- B: 82-87
- B-: 80-81
- C+: 78-79
- C: 70-77
- D+: 68-69
- D: 62-67
- D-: 60-61
- F: 59 and less

Expanded Description: Content and Form the zyBooks

ZyBook Sections

Table of contents (ZyBooks Sections)

Ch 01. Introduction to Data Structures and Algorithms
1.1 Data structures
1.2 Introduction to algorithms
1.4 Abstract data types
1.6 Algorithm efficiency
1.7 LAB: Introduction to data structures labs

Ch 02. Searching and Algorithm Analysis
2.1 Searching and algorithms
2.2 Binary search
2.3 Constant time operations
2.4 Growth of functions and complexity
2.5 O notation
2.6 Algorithm analysis
2.7 Recursive definitions
2.8 Recursive algorithms
2.9 Analyzing the time complexity of recursive algorithms
2.10 LAB: Binary search template function

Ch 03. Sorting Algorithms
3.1 Sorting: Introduction
3.2 Selection sort
3.3 Insertion sort
3.4 Shell sort
3.5 Quicksort
3.6 Merge sort
3.8 Overview of fast sorting algorithms
3.9 C++: Sorting with different operators
3.10 LAB: Natural merge sort

Ch 04. Lists
4.1 List abstract data type (ADT)
4.2 Singly-linked lists
4.3 Singly-linked lists: Search and insert
4.4 Singly-linked lists: Remove
4.5 Doubly-linked lists
4.6 Doubly-linked lists: Search and inserts
4.7 Doubly linked lists: Remove
4.8 Linked-list traversal
4.9 Sorting linked lists
4.11 Linked lists: Recursion

4.12 Array-based lists
4.13 LAB: Sorted number list implementation with linked lists

Ch 05. Stacks and Queues
5.1 Stack abstract data type (ADT)
5.2 Stacks using linked lists
5.3 Array-based stacks
5.4 Queue abstract data type (ADT)
5.5 Queues using linked lists
5.6 Array-based queues
5.7 Deque abstract data type (ADT)
5.8 C++ stack class
5.9 C++ queue class
5.9 LAB: Grocery list editor with undo stack

Ch 06. Hash Tables
6.1 Map ADT
6.2 Hash tables
6.3 Chaining
6.4 Linear probing
6.6 Double hashing
6.7 Hash table resizing
6.7 Common hash functions
6.8 Common hash functions
6.9 Direct hashing
6.11 C++ unordered_map class
6.12 LAB: Course gradebook with unordered_map

Ch 07. Trees
7.3 Binary search trees
7.4 BST: Search algorithm
7.5 BST: Insertion
7.6 BST: Remove
7.7 BST: Traversal
7.8 BST: Height and insertion order
7.9 BST: Recursion
7.10 BST: Parent node pointers
7.11 Set abstract data type (ADT)
7.12 Implementing a set ADT with a BST
7.13 C++ unordered_set class
7.15 LAB: BST validity checker

Ch 08. Balanced Trees
8.1 AVL: A balanced tree
8.2 AVL rotations

8.3 AVL insertions
8.4 AVL removals
8.9 LAB: AVL tree Nth largest operation
8.10 LAB: AVL tree Nth largest operation

Ch 10. Graphs

10.1 Graphs: Introduction
10.3 Graph representations: Adjacency lists
10.4 Graph representations: Adjacency matrices
10.5 Directed graphs
10.6 Weighted graphs
10.7 Vertex, Edge, and Graph classes
10.8 Graphs: Breadth-first search
10.9 Graphs: Depth-first search
10.10 Algorithm: Dijkstra's shortest path
10.11 Algorithm: Bellman-Ford's shortest path
10.13 Minimum spanning tree
10.14 All pairs shortest path
10.15 LAB: Graph representations

Ch 11. Algorithms

11.1 Huffman compression
11.4 Greedy algorithm
11.5 Dynamic programming
11.6 Lab: Longest common subsequence

Ch 14. Artificial Intelligence

ZB14 New chapter: Artificial Intelligence with five new sections
14.1 Artificial intelligence
14.2 Machine learning
14.3 Computer vision using the transformers module.
14.4 Natural language processing
14.5 Risks and ethics in AI

Assignments

A. Reading: Required reading from the online interactive text
B. Doing the homework zyBooks assignments online.

Compilers & IDE:

- **Windows & Mac:**

Visual Studio 2022: Community (Free):

https://visualstudio.microsoft.com/downloads

VSCode (Windows and Linux and Mac) (Free):

https://code.visualstudio.com/download

- **Mac:** Xcode, Neovim
- **Online Compiler:**

https://www.onlinegdb.com/online_c++_compiler (recommended, free)

https://www.tutorialspoint.com/compile_cpp_online.php

https://www.programiz.com/cpp-programming/online-compiler/

Useful Tutorials:

https://www.geeksforgeeks.org/cpp-tutorial

https://www.geeksforgeeks.org/cpp-tutorial/

https://www.tutorialspoint.com/cplusplus/index.htm

https://thispointer.com/c11-tutorial

Useful Interview Problems:

https://interview.leetcode.com/interview/login/

https://www.hackerrank.com/domains/cpp

C++ Uses:

https://www.simplilearn.com/tutorials/cpp-tutorial/top-uses-of-c-plus-plus-programming

https://www.codingninjas.com/blog/2021/07/29/c-vs-java-vs-python-which-one-to-choose/

Student Resources:

The college has gathered all Canvas Resources for Students into a library; here is the link:
https://deanza.instructure.com/courses/3382

Dates and Deadlines:

June 30
Spring classes begin

TBA
Last day to add 12-week classes
TBA
Last day to drop classes without a W
July 4
Independence Day
TBA
Last day to drop classes with a W
TBA
Independence Day - no classes, offices closed
August 04-08
Final exams
August
Graduation