INSTRUCTOR: Cynthia Lee-Klawender
          (http://www.deanza.edu/faculty/leeklawendercynthia/)
OFFICE HOURS, OFFICE: Mondays and Wednesdays 1:30-2:20 PM (in F41C or AT203);
          Tuesdays and Thursdays 12:50 - 1:15 PM (in F41C and 3:25 - 3:50 (in F41C or
          AT203), or by appointment in F41C (in the L quad between L4 & L6 buildings)
PHONE & EMAIL: (408) 864-8609, E-mail for questions or help:
          LeeKlawenderCynthia@deanza.edu

PREREQUISITE: Computer Information Systems 22B or equivalent.

ADVISORY: Mathematics 212 or equivalent.

COURSE DESCRIPTION: 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 OUTCOMES:

   ● 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 efficiency of advanced level data structures algorithms, code,
   document, debug, and test advanced data structures programs using multiple source
   and header files.

TEXTBOOK: **Data Abstraction & Problem Solving with C++: Walls and Mirrors,
   6th Edition**, by Frank M. Carrano, (ISBN-13: 978-0-13-292372-9, ISBN-10: 0-13-
   292372-6)

LESSONS: Will be provided online on Catalyst

CLASS NOTES: Many of the class exercise answers and announcements will be posted in
          Catalyst (https://catalyst.deanza.edu) on a week-by-week basis.

COMPUTER LAB: You may use our computer lab or your own (or another) computer
          and compiler. If you don't use our computer lab, you need to have a C++
          compiler in order to do homework assignments. If you're enrolled in this class,
          you will automatically have an account in our classroom and AT203 Open
          Computer Lab (if you're adding, add online in Admissions office, wait about 20
          min. before using the open lab). Bring a flash drive to the Computer Lab to back
          up your programs or remember to email to yourself.

COURSE OUTLINE (subject to change): This is an online class. Most of the class will be
          presented in Catalyst. The midterm and final will be held on campus (see
          schedule below for exact dates, times and location). There will be weekly online
          meetings. Information about the online meetings will be given in Catalyst.

Dates	Lessons	Resources

Week	Dates	Topics	Readings
Week 1	Apr. 6 - 12	Review of Linked Lists, Templates, 22C Node and Linked List classes	Catalyst Lesson 1, Textbk. Ch. 1 (C++ Interlude)
Week 2	Apr. 13 - 19	Stacks Queues	Catalyst Lesson 2 & 3, Textbk Ch. 6&7, Ch. 13&14
Week 3	Apr. 20 - 26	Complex Linked Lists	Catalyst Lesson 4
Week 4	Apr. 27 - May 3	Recursion Algorithm Efficiency	Catalyst Lesson 5, Textbk Ch. 2, Ch. 5 Catalyst Lesson 6, Textbk Ch. 10
Week 5	May 4 - 10	Trees, Binary Trees, Binary Search Trees	Catalyst Lesson 7, Textbk Ch. 15&16
Week 6	May 11 - 17	Hashing MIDTERM Thur. May 14 6:30-8:30 PM in Adm119	Catalyst Lesson 8, Textbk Ch. 18
Week 7	May 18 - 24	AVL Trees	Catalyst Lesson 9, Textbk Ch. 19
Week 8	May 25 - 31	*HOLIDAY May 25* Heaps	Catalyst Lesson 10, Textbk Ch. 17
Week 9	Jun. 1 - 7	Graphs	Catalyst Lesson 11, Textbk Ch. 20
Week 10	Jun. 8 - 14	Sorting: Shell Sort, Quick Sort, External Sort	Catalyst Lesson 12, Textbk Ch. 11&21
Week 11	Jun. 15 - 21	Project Presentations Review for Final	Info. on Catalyst On Catalyst
Week 12	FINAL	**FINAL EXAM** Thur. June 25 6:15 - 8:15 PM in Adm119	Comprehensive

EVALUATION: Prog. Assignments (5)   30% (Each = 6%)
            Participation   10% (Class exercises)
            Midterm   25%
            Team Project   10%
            Final Exam   25%
                                          100%

MAKE-UP TESTS: NO MAKE-UP TESTS WILL BE GIVEN! Please notify the instructor ASAP if you know ahead of time you will be missing a test.

EXTRA CREDIT: Maximum of 5 extra credit projects may be counted! If the project is other than given with the lab assignments, it must be approved by the instructor before they are attempted or it may not be accepted. This will be discussed later in the semester. The extra credit projects are due at the final exam, and *must include the source file(s) and* **output**. (Note: Extra credit will only be counted if the student is receiving less than an A+.)

WITHDRAWING FROM CLASS: I will not automatically drop anyone from class, even if you stop attending classes. If you wish to discontinue the class, you must go the Admissions Office and turn in a signed drop card or you may receive an 'F' in the class.

PROGRAMMING HOMEWORK GRADING: Each will be graded as follows:

      35 points: Does the program correctly & completely solve the problem?

      7 points: Is the listing <u>commented & indented</u>? Will I understand what the program is doing (good logic, variable, class and function names)? Is the program efficient?

      8 points: ON TIME! (1 point deducted starting day after due date + every other day late--CAN'T TURN IN 3 WEEKS after due date!)

    50 points possible (for each programming assignment)

    NOTE: NOTHING WILL BE ACCEPTED AFTER **Fri., June 26, 11:59 PM**!

GRADING BREAKDOWN (adding each score/max-points * weight):

Grade	Low	Condition	High
A+		Total Percent >=	97.0
A	90.5	<= Total %<	97.0
A-	87.5	<= Total %<	90.5
B+	84.5	<= Total % <	87.5
B	80.5	<= Total % <	84.5
B-	77.5	<= Total% <	80.5
C+	74.5	<= Total% <	77.5
C	69.5	<= Total% <	74.5
D+	65.5	<= Total% <	69.5
D	60.5	<= Total% <	65.5
D-	57.5	<= Total% <	60.5
F		Total Percent <	57.5