

**Instructor:** Delia Gârbacea

**Office Location** : F51c

**Phone** : (408) 864-8308

**Email** : [GarbaceaDelia@DeAnza.edu](mailto:GarbaceaDelia@DeAnza.edu)

**Office Hours** : MTWTh 3:20 – 4:10 p.m. AT203B

**Class meetings:** Tue, Thu 6:00 – 7:50p.m. Lecture AT205

The instructor will be available **Online:** Thu, 10:45 – 12:00 noon.

**Course Description:** Applications of advanced features of C and the C-library functions including: binary and random-access input/output, dynamic data structures, bit manipulation, string parsing and string-to-numeric conversion, event and error processing, function pointers, recursion, and variable-length argument list functions.

**Text required:** Perry, John. "Advanced C Programming by Example" Belmont, CA: PWS Publishing, 1998.

**Supporting Texts and References:** Harbison, Samuel and Guy L. Steele, Jr. "C: A Reference Manual," 5th Edition. Englewood Cliffs, NJ: Prentice Hall Software Series, 2002

**Attendance policy:** This 4.5 Units course consists of 4 lecture hours (4x50min = 200min) and 1.5 lab hours (75min) per week. You are expected to attend all lecture sessions and complete all online work. If you must be absent from class, arrange with another student to share class notes for that session. You should plan on spending approximately another 16 to 24 hours per week to study and do your homework. Remember, there is no such thing as a dumb question: you want to understand before you get lost. In addition, please be assured that I'm ready to assist you in any way possible as you meet the challenges ahead. If you wish to withdraw from class, it is your responsibility to do so. An unauthorized withdrawal from class without following official procedures will result in you being assigned a grade of "F" (or "NC" if you have selected the Credit /No Credit option).

**Student Learning Outcomes:** By the end of the course, students will:

- Read, analyze and explain advanced C programs.
- Design solutions for advanced problems using appropriate design methodology incorporating advanced programming constructs.
- Create algorithms, code, document, debug, and test advanced C programs.

**Syllabus**

**Course objectives:** Upon completion of the course, students will:

- Create programs which demonstrate mastery of common pointer manipulation.
- Create programs which demonstrate knowledge of memory management functions and their use in the creation of linear dynamic data structures.
- Create programs which use bitwise operators.
- Create programs which use advanced string functions.
- Create programs which employ advanced file input/output functions.
- Create programs which use recursion -- especially the algorithms involving non-linear dynamic data structures such as binary trees.
- Demonstrate usages of multi-dimensional arrays and the internal addressing formulae used by such data structures.
- Use the ANSI C time functions to timestamp events and perform program benchmarking.
- Demonstrate knowledge of advanced subprogram (function) features.
- Write code which interacts with the Operating System environment outside of a program for event, error, and exception handling.
- Demonstrate knowledge of sorting and searching within ordinary arrays using appropriate ANSI C functions.

**Scholarly conduct:** In order to be successful in this class you will have to make a commitment to studying, reading the text, doing your homework, writing your lab assignments, attending class, and taking notes. Worthwhile contribution and regular attendance can positively affect the grades. You are expected to do your own work. Cheating or plagiarism in any form will not be tolerated. Copying or cheating during a test will result in a zero being assigned for that test. Any copied assignments will result in a zero grade for all parties. Please check the current Schedule of Classes to learn more about academic integrity, other policies, and Student Standards of Conduct: <http://www.deanza.edu/policies/academicintegrity.html>

**Reading assignments and recommended Exercises:** The book exercises and problems are not to be run on the computer (unless you wish to). The purpose of these exercises and problems is to help clarify the material for you as we proceed and to prepare you for tests, therefore, although you do not have to turn them in, you are strongly encouraged to solve them.

**Tests:** There will be one midterm exam (100 points) and a comprehensive final (100 points). All tests are open book, open notes, no computers. Test dates are shown on the calendar. The midterm exam will be returned, giving students a good idea of their progress in the class. Final exams will be retained for a period of 90 days from the exam date.

**Class and online assignments:** There will be given between 12 to 24 assignments to be completed in class or on line, such as group work, quizzes, etc. They are open book, open notes. There is no make up for any such assignment. These are pass/no pass type of assignments.

## Syllabus

**Programming assignments (homework):** You will be given 5 programming assignments (100 points each), half completed in the computer lab, half completed as homework. They are to be run using the computer.

- Up to 5 points will be deducted for each day an assignment is late.
- No assignment will be accepted more than one week after the due date, unless there's an exceptional situation (email me or come and talk to me, preferable in advance).
- Partial credit will be given for incomplete assignments.
- There is no make up for any assignment.
- Assignments must be uploaded on Catalyst: all source and header files, input files and any output generated by the program.

**Extra-credit assignments** may occasionally be given throughout the course (up to 9 points).

**Grading:** To pass the class you have to do the following:

1. complete **80%** of the class assignments
2. avg = average of the 5 programming assignments should be **70** or greater
3. mt = midterm exam should be **60** or greater
4. final exam score should be **60** or greater

If all of the above are true, your final grade will be calculated as follows:

$$\text{score} = (\text{avg} + \text{mt} + \text{final} + \text{extra credit}) / 3$$

Your grade is based on the score you earn as shown below. Worthwhile contribution and regular attendance can positively affect the grades.

Class Assignments	: 12 - 24	P/NP	[97, 103] → <b>A+</b>
			[93, 97) → <b>A</b>
Midterm Exam	: 1 at 100pts.	100	[90, 93) → <b>A-</b>
			[87, 90) → <b>B+</b>
Programming Assignments:	5 at 100 pts. each		[83, 87) → <b>B</b>
	AVERAGE: 500 / 5 =	100	[80, 83) → <b>B-</b>
			[77, 80) → <b>C+</b>
Final Exam	: 1 at 100 pts.	100	[70, 77) → <b>C</b>
Extra Credit	:	9	[67, 70) → <b>D+</b>
			[63, 67) → <b>D</b>
			[60, 63) → <b>D-</b>
			[ 0, 60) → <b>F</b>
	SCORE: (100 + 100 + 100 + 9) / 3 = 103		

## Tentative Schedule

	Chapters	Mon	Tuesday	Wed	Thursday	Fri	<u>Important Dates</u> <i>All dates are enforced!</i>
<b>APRIL</b> Week 1	Ch 0: Intro to C Ch 1: Style Ch 2: Review Pointers and Arrays	6	7	8	9	10	<b>Saturday, Apr. 11</b> Last day to <a href="#">drop</a> for a <a href="#">refund</a> for out-of-state or foreign students
Week 2	Ch 2: Review Ch 3: Linked Lists	13	14	15	16 <b>Hw 1</b> Review	17	<b>Saturday, Apr 18</b> Last day to <a href="#">add</a> quarter-length classes, drop for refund <b>Sunday, Apr. 20</b> Last day to <a href="#">drop</a> a class with no record of grade
Week 3	Ch 3: Linked Lists	20	21	22	23	24	
Week 4 <b>MAY</b>	Ch 4: Strings	27	28	29	30 <b>Hw 2</b> Linked Lists	1	<b>Friday, May. 1</b> Last day to <a href="#">request pass/no pass</a> grade
Week 5	Ch 5: Advanced Input/Output	4	5	6	7	8	
Week 6	Ch 6: Bit Manipulation	11	12	13	14 <b>Hw 3</b> Advanced I/O	15	
Week 7	Ch 7: Recursion & Binary Trees	18	19	20	21 <b>Midterm</b>	22	
Week 8	Ch 7: Recursion & Binary Trees Ch 8: Multi- dimensional Arrays	25 <b>Mem orial Day</b>	26	27	28 <b>Hw 4</b> Bit Manipulation	29	<b>Friday, May 29</b> Last day to <a href="#">drop</a> with a "W"
<b>JUNE</b> Week 9	Ch 8: Multi- dimensional Arrays Ch 9: Pointers to Functions, ...	1	2	3	4	5	
Week 10	Ch 9: Pointers to Functions, ... Ch 10: Sorting /Searching...	8	9	10	11 <b>Hw 5</b> Trees, recursion, etc.	12	
Week 11	Ch 10: Sorting /Searching... Final Review	15	16	17	18 <b>REVIEW</b> (Last Lecture)	19	
Week 12		22	23 <b>Final Exam</b> 6:15 - 8:15pm	24	25	26	