University of New Orleans
ScholarWorks@UNO

University of New Orleans Syllabi

Fall 2015

# CSCI 5125

Shengru Tu
*University of New Orleans*

Follow this and additional works at: https://scholarworks.uno.edu/syllabi

**This is an older syllabus and should not be used as a substitute for the syllabus for a current semester course.**

Recommended Citation
Tu, Shengru, "CSCI 5125" (2015). *University of New Orleans Syllabi.* Paper 184.
https://scholarworks.uno.edu/syllabi/184

This Syllabus is brought to you for free and open access by ScholarWorks@UNO. It has been accepted for inclusion in University of New Orleans Syllabi by an authorized administrator of ScholarWorks@UNO. For more information, please contact scholarworks@uno.edu.

**CSCI 5101: Section 001 [M,W 3:30-4:45 p.m. MTH 229]**
***Instructor:*** N. Adlai A. DePano, Ph.D.

# *Analysis of Algorithms*
**Fall 2015**

## Preliminary Remarks:

Welcome to CSCI 5101, *Analysis of Algorithms*. This course delves into the analysis (and design) of algorithms which is at the heart of the software development process. The listed prerequisites for this course imply a prior knowledge of programming using an advanced structured language (like Java or C), familiarity with data structures and their use, and that much-sought but hard-to-define commodity needed for an advanced course like this – *mathematical maturity*.

## Catalog Description:

Precise definition of the concept of an algorithm; techniques for algorithm verification; analyzing algorithm performance; applications to practical algorithms.

## Prerequisites:

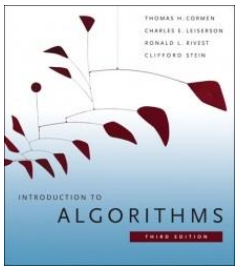
CSCI 2125, *Data Structures*, with a grade of C or better.

## Learning Objectives:

At the conclusion of this course, the successful student should be able to:

1. Define precisely what an algorithm is in the context of the software development process;
2. Apply this definition to problem instances to help identify problem elements that facilitate the design and/or choice of algorithmic solutions;
3. Apply techniques of analysis to assess the correctness and efficiency of an algorithm in a variety of applied problems;
4. Recognize and apply algorithmic patterns (recursion, divide-and-conquer, *etc*.) to a variety of problems; and
5. Use accepted terminology in describing algorithms and their measures of performance (correctness, efficiency, *etc*.).
6. Read extant literature about algorithms with a deeper understanding of the underlying metrics used in assessing algorithm correctness and efficiency.

## Text:




*Introduction to Algorithms*, 3rd ed., Cormen *et al*, MIT Press, 2009 (ISBN-13: 978-0-262-53305-8). This has become a classic in the CS literature and even though the textbook will certainly contain more material than we can possibly cover in the semester, it will be a welcome addition to your personal library for future reference. Supplementary notes will be made available on-line. The text should be available at the bookstore. The following links are intended to assist those who may not be able to avail of the bookstore's services:
Amazon.com page for the Cormen text; MIT Press page for the Cormen text

## Course Topics:

The following chapters from the textbook will be discussed during the course of the summer term:

1. Ch. 1: The Role of Algorithms in Computing
2. Ch. 2: Getting Started

3. Ch. 3: Growth of Functions
4. Ch. 4: Recurrences
5. Ch. 6: Heapsort
6. Ch. 8: Sorting in Linear Time
7. Ch. 11: Hash Tables
8. Ch. 12: Binary Search Trees
9. Ch. 22; Elementary Graph Algorithms
10. Ch. 23: Minimum Spanning Trees
11. Ch. 24; Single-Source Shortest Paths
12. Ch. 26: Maximal Flow
13. Ch. 34: NP-Completeness

## Course Format:

The course will be primarily in lecture format – questions and discussions are most welcome, however. Student participation is strongly encouraged. Problem sets will be assigned and may be solved in an in-class group work setting. Students are expected to read ahead. (This enhances their chance to participate meaningfully in any class discussion.) Programming exercises may be assigned as necessary. In such an event, programs must reflect the principles you acquired in your programming courses (*i.e.*, good documentation, clear logic design, and *integrity*). Students will be assigned additional readings from current literature and will be required to submit a written critique of these readings.

## Tests and Exams:

The test schedule is tentative. We will have two tests plus the final exam. Tentative dates for the tests are Sept. 23rd and Nov. 11th. **The final exam is firmly scheduled as follows: Wednesday, Dec. 9th, from 3:00 p.m. - 5:00 p.m.**

## Homework:

Homework assignments are a crucial part of the learning process in this course. It is by "getting one's hands dirty" that one absorbs the subtle points of theoretical issues. Only by being able to translate theory into practice can one truly say that the material has been learned. Policies that apply to homework are as follows:

1. Homework (but not group work) will (typically) be assigned at least a week prior to its due date.
2. "Late" work will be penalized. Every full day that an assignment is late doubles the penalty of the previous day, starting with one percentage point deducted on the first day. So in practice, an assignment that is turned in a week late automatically gets an F, having already incurred a penalty of -64! However, it is understood that during the course of the semester, life doesn't always follow a smooth path; therefore, everyone is entitled to a "free pass," that is, the lowest homework score will be dropped from the final computation of grades.
3. Unintelligible=*wrong*=no credit. You are expected to communicate your thoughts clearly. Submitted homework is expected to be neat, solutions (if required) appearing in order, and, in general, clearly explained by accompanying explanations in English. One suggestion is to work out problems on scratch paper and to recopy them for the final submission.
4. Homework (obviously, not group work) is expected to be *your own personal effort*. However, there is very little one can do to prevent you from "consulting" with each other on homework assignments. In a limited way, this can be beneficial to you. After all, teaching one another and working together are important skills. However, it is essential that your homework submissions reflect your own personal analysis and solution. It is suggested that you try to work on the problem on your own, and then only when you get stuck should you begin discussion with your colleagues. The interaction should be two-way – you contributing to it as well as profiting from it. When writing the final submission, try to recreate the arguments on your own. Only when you

can do this can you truly say that you have learned from the group effort. Needless to say, this joint work policy applies only to homework, and not to examinations!

## Grading:

The final grade will be computed as follows: Homework and group work will comprise 30%; tests make up the remaining 70%. The test component will be computed as follows: Two in-class tests and the final exam counted twice give four grades. The highest three will be used to compute the test component of your final grade. For instance if your in-class test grades are 70 and 80, and your final exam grade is 75, then the grades 80, 75, 75 (*i.e.*, highest three of 70, 80, 75, 75) will be used.
Letter grades will be assigned as [90 - 100] = A, [80 - 90) = B, [60 - 80) = C, [50 - 60) = D, [0 - 50) = F. Final grades may be curved up at the discretion of the instructor, but don't count on it.

## Office Hours:

My office is in MATH 308. I will be available at the following times: M,W, F 12:00 noon-2:00 p.m. More hours will be announced as the academic schedules firm up Other times by appointment only (504-280-7370 or by e-mail to *ndepano@uno.edu*).

## Attendance Policy

Attendance will be taken at each session. Although not specifically included as a basis for the final grade, attendance can have an impact on borderline cases. Good attendance is an indication of the dedication of the student to the learning enterprise. This will be taken by the instructor into consideration in such cases.

## Academic Dishonesty

As a matter of policy, we call your attention to the University's rules regarding academic dishonesty (http://www.studentaffairs.uno.edu/pdfs/AcademicDishonestyPolicy.pdf). Academic dishonesty includes cheating, plagiarism, and collusion. In particular, it includes "the unauthorized collaboration with another person in preparing an academic exercise" and "submitting as one's own any academic exercise prepared totally or in part for/by another." In the event of academic dishonesty, **the student will be assigned a grade of 0** on the exam or exercise, the student will be informed in writing of the action taken, and a copy of this letter will be sent to the Assistant Dean for Special Student Service

## Students With Special Needs

Finally, we provide here a link to the University's Office of Disability Services webpage (http://www.ods.uno.edu/index.cfm). As expressed therein, the University pursues two primary objectives: (1) to ensure compliance with Section 504 of the 1973 Rehabilitation Act and the Americans with Disabilities Act (ADA) in regard to equal access for qualified students to academic programs; and (2) to uphold the academic integrity of UNO. Part of this policy regulates the accommodative testing services. Due to impending construction/repair work in the Library, the Administrative Office for Disability Services will be temporarily located in Milneburg Hall Room 159. The Accommodative Testing and Adaptive Technology Center (ATATC) will be temporarily located in Sciences Bldg. Room 1046.

**Note that October 14th is the last day to drop classes for the Fall 2015 term.**