

CSCI/CMPE3333.03

Algorithms and Data Structures

Fall 2019

Syllabus

Department of Computer Science

University of Texas Rio Grande Valley

Instructor

Name: Liyu Zhang

Office: SETB 1.522, Brownsville Campus

Tel: (956) 882-6631

E-mail: liyu.zhang@utrgv.edu

Webpage: <http://faculty.utrgv.edu/liyu.zhang/>

Office hours: MW 10-12, TR 11-12, and by appointment.

Course Information

Credit hours: 3

Class times: MW 1230pm - 145pm, BLHSB 1.314.

WWW: <https://my.utrgv.edu/home>, login and click on the Blackboard Icon, and then click on the link for CSCI/CMPE3333.03, Fall 2019.

Required textbook:

M. Weiss, *Data Structures & Algorithm Analysis in C++, Hardcover, 4e*, Pearson, June 23, 2013, ISBN 978-0132847377.

Recommended Reference:

T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein. *Introduction to Algorithms, 3e*. The MIT Press, 2009. ISBN: 978-0262033848.

Course Description (Catalog) This course is a continuation of data structures topics covered in CSCI/CMPE 2380. Content includes theoretical topics in algorithmic efficiency and complexity, along with abstract data types, including graphs, networks, trees, and priority queues. Search topics, including hashing, trees, external search trees (B-trees), and sorting algorithms including external sorting are introduced and compared. (Computational complexity topics include the Class P and NP, NP-completeness and Reducibility, NP-completeness Proofs, and NP-complete Problems.)

Prerequisites You must have a C or higher in both CSCI/CMPE 2380 (Computer Science II) and CSCI3310 (Mathematical Foundation of Computer Science, used to be Discrete Data Structures). You must have the instructor's approval to take this course if you have not satisfied the prerequisites. In general, you are expected to have familiarity with advanced programming

techniques, using principles of software engineering, as well as concepts including files, two-dimensional arrays, stacks, queues, linear and circular linked lists, tree data structures, and some sorting and searching methods.

Covered Topics CSCI/CMPE 3333 teaches you algorithms, data structures, and software principles -- those being fundamentals needed as essential groundwork for making the rest journey of learning computer science. It emphasizes fundamental questions of computer science: how to organize large amount of data and how to design algorithms to process large amount of data efficiently in time and space. It covers the following topics:

- Algorithms design and analysis: greedy algorithms, divide and conquer, dynamic programming, randomized algorithms, backtracking algorithms, graph algorithms, and amortized analysis, computational complexity, P vs. NP problems.
- Sorting: various methods including heapsort, mergesort, quicksort, indirect sorting, bucket sorting, and external sorting.
- Data structures: lists, stacks, queues, sets, trees, and graphs.
- Hashing: hashing functions, separate chaining, open addressing, rehashing, and extendible hashing.

Course Objectives After completing this course, a student should be able to:

- Understand basic data structures and abstract data types.
- Gain an appreciation of the variety, theoretical nature, and practical uses of data structures.
- Select appropriate data structures for uses in computer programs.
- Understand the basic techniques of algorithm design and analysis.
- Understand the basic concepts of computational complexity
- Design and implement efficient algorithms based on the selected data structures.

Course Organization The class meet for lecture twice a week. Students must study the material assigned by the instructor and complete assignments. There is one mid-term exam during the semester, which will be held in class on **Thursday, Oct 10th, 2019**. There is also a final exam at the end of the semester. All the exams will be based on materials covered in lectures and assignments. Please do not plan to travel at the end of semester until the final exam is over.

Assignments There will be two types of assignments in this course: homework exercises and programming projects. Homework exercises are essay problems that will help you practice and learn the theoretical aspects of algorithms including writing formal algorithm descriptions in

pseudo-programming languages, and providing proofs of correctness and mathematical complexity analysis. Programming projects will help you practice and learn advanced programming using moderate to complex algorithms and data structures. Learning both aspects of the algorithms can be achieved only through working at their respective assignment problems. Therefore, it is very important for you to do each assignment seriously. Most homework exercise problems will be essay questions while programming projects will be implementing algorithms given in the textbook or your theoretical solutions to some homework exercise problems in order for you to reinforce your understanding of algorithms. Both types of assignment problems will be given on a weekly/bi-weekly basis throughout the semester. Please note the following assignment submission requirements:

- Homework exercise problems: Typed submissions are recommended although handwritten solutions are acceptable as long as they're written clearly and completely readable. The instructor cannot grade your solutions if they're unreadable. You must staple your submission if it is more than one page and write/type clearly your name, student ID number and due date on the first page. Assignments are usually due on Wednesdays. Assignments are due at the beginning of lectures on the due dates.
- Programming projects: You will need to document your programs well and include a readme file containing instructions of running your program for each project submission. Acceptable programming languages are C++ (strongly preferred) and Java. Please submit your projects through the project submission link provided in the BB portal for the course. You must follow the instructions provided at the submission link. No programming assignment will be graded if the submitted program does not run.

No more than two assignment submissions that are late or do not meet submission requirements as described above will be accepted for each student. In addition, no assignment submissions will be accepted regardless if the solutions to those assignments have been already posted or given in class, or if a majority of the submitted solutions have been graded and returned to students. All submitted assignments, homework exercises and programming projects, are subject to oral defenses, where students are required to explain to the instructor key steps and details of the submitted assignment solutions satisfactorily and demonstrate complete understanding of the submitted work. Unsatisfactory assignment defenses might lead to grades of relevant assignment problems or whole assignments voided at the instructor's discretion.

Attendance Attendance of lectures is taken and counts towards your final grade for this course. No excuse other than officially documented cases allowable by the university policies, which are usually only for family or extreme health emergencies, will be accepted for absences. You are not required to attend class on days listed in the university calendar as major religious holy days (although I assume that you practice at most one religion). In addition, you're allowed two absences without excuses or grade penalties. Students have the option to be exempted from attendance of lectures, in which case the percentage weight of attendance will be distributed proportionally among other grading components. To activate this option, however, students must notify the instructor no later than **Monday, September 9, 2019**.

Grading The assignments and exams will be graded on the correctness of both the answers to the questions and the process you show to obtain the answers. Your final grades for this course will be based on your attendance if not exempted, assignments and exams. A breakdown of weights for each grading component is as follows.

Attendance 10%, Assignments (homework and projects) 40%, Mid-term Exam 25%, Final Exam 25%.

I will not make changes in final grades unless the student can document an error on his grade records in a timely manner (See Regrading).

I will use the following number-to-letter grade mapping as dictated by UTRGV to assign final letter grades at the end of the course. I reserve the right to curve up (but not curve down) grades when and if I feel necessary.

100% >= A >= 90% > B >= 80% > C >= 70% > D >= 60% > F

Re-grading If you have a question about the grading of any piece of work, you should consult with the instructor of the course within one week of the date that the work was returned. In other words, if you do not pick up your work in a timely fashion, you may forfeit your right to question the grading of your work.

Office Hours Office hours offer you the opportunity to ask more individual questions about the course. Office hours are held on a first-come first-served drop-in basis. No appointment is necessary to attend office hours. Be aware that office hours become increasingly busy when it is close to a homework deadline and/or exam date. Plan your use of office hours accordingly. Individual appointments may be arranged, if needed, as schedules allow.

Study Outside of Class In this course, as in any other course, you are expected to put in additional time beyond the scheduled class times. Professors generally expect that for each credit hour a class carries a typical student will put in 2 – 3 hours of time each week outside of class. Since this is a 3 credit course that translates into 6 – 9 hours of time outside of lecture times, each week. During this time you should read the material before coming to class and then again in greater detail after class. You should also attend office hours as needed and digest course materials thoroughly by doing assignments.

Incompletes and Course Withdrawal I will not give incomplete grades except for the rare cases dictated by the University and Department policy. It is the student's responsibility, not the instructor's, to withdraw from the course in a timely manner if doing poorly. No incomplete grades will be granted because of a wrong withdrawal process. Please obtain due dates to withdraw from the course and also please read and be aware of the formal procedures to withdraw. This information is available in the course schedule and the student affairs office.

Online Blackboard We will use UTRGV online Blackboard as the place for making announcements and posting course materials/information such as course calendar, lecture notes, assignments and grades etc. So please check Blackboard regularly and at least once every 24 hours. You can also post your questions there so that I, or even your fellow classmates can answer them. It is YOUR responsibility to keep updated with class through online Blackboard.

General Notes If you don't understand something covered in class, ask about it right away. The only silly question is the one that is not asked. If you get a poor mark on an assignment, or exam, find out why right away. Don't wait a month before asking. I will be happy to answer your questions. Don't be afraid to ask questions, or to approach the instructor in class, during office hours, in the hallways, or through e-mail.

The material in this course is hard as it requires quite much mathematics as well as advanced programming skills. However, it could also be fun. Playing with proofs, formulas and data structures, you will have a chance to appreciate the beauty of use of mathematics in creating better and faster computer algorithms. You will also see those neat and elegant ideas in algorithm design, which are among the greatest ideas in computer science. We think algorithms are interesting and fundamental to computer science, and we want to convince you of this. Work hard, but have fun!

Student Integrity Cheating of any kind will not be tolerated. Any assignment or exam that is handed in must be your own work. However, talking with one another to understand the material better is strongly encouraged. Recognizing the distinction between cheating and cooperation is very important. If you copy someone else's solution, you are cheating. If you let someone else copy your solution, you are cheating. We will not distinguish between the person who copied a solution and the person whose solution was copied. Both people will be treated as cheaters. If someone dictates a solution to you, you are cheating. Everything you hand in must be in your own words, based on your own understanding of the solution. If someone helps you understand the problem during a high-level discussion, you are not cheating. We strongly encourage students to help one another understand the material presented in class, in the book, and general issues relevant to the assignments. When taking an exam, you must work independently. Any collaboration during an exam will be considered cheating. When a cheating is caught, zero marks

will be given the cheated work, and the case will be forwarded to the Department chair and beyond if necessary.

Student Learning Outcomes

Level 3: Synthesis and Evaluation

Level 3 outcomes are those in which the students can apply the materials in new situations. This is the highest level of mastery. Upon successful completion of this course, students will be able to

- specify data structures and operations associated with abstract data types
- define the signature and pre- and post-conditions for operations of an abstract data type
- Given a scenario, describe the abstract data types that could be created
- identify, implement, and use the following data structures as appropriate for a given problem:
 - lists implemented as arrays or linked lists
 - stacks
 - queues
 - binary trees and binary search trees
 - simple hashes
- implement binary trees and binary search trees, using pre-, post-, or in-order traversals as appropriate for a given situation
- judge which data model (list, tree, graph, or set) is appropriate for solving a problem
- justify the choice of a data structure to solve a problem based on issues such as time, space, and of the data structure.
- judge which implementations are best suited for an application that requires a list data model: lists, circular lists, circular queues, or generalized list
- judge whether an array or linked implementation is best suited or an application that requires a data model
- judge which graph representations (adjacency list, adjacency matrix, edge list) are appropriate for solving a problem
- develop algorithms that are based on depth- and breadth-first traversals of general trees, binary trees and graphs
- judge which sort algorithms (insertion, selection, mergesort, heapsort, quicksort, radix) is appropriate for solving a problem
- judge which search algorithm and data structure is appropriate for solving a problem
- implement a recursive solution to a problem

Level 2: Application and Analysis

Level 2 outcomes are those in which the students can apply the materials in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details.

Upon successful completion of this course, students will be able to

- use Big-O notation to express the best-, average-, and worst-case behaviors of an algorithm
- explain the structure and use of activation records
- determine the best-, average- and worst-case behaviors of an algorithm
- assess time and space trade-offs in algorithms
- explain, code, and use $O(n \log n)$ -time sorting algorithms
- implement recursive algorithms over natural numbers, lists, and trees
- define and use classes, subclasses, and inheritance
- describe the importance of encapsulation and information hiding
- implement applications and simulations using data structures identified above
- implement simple sequential and binary search algorithms
- implement a set of searching/sorting algorithms
- categorize algorithms based on programming strategy, i.e., divided-and-conquer, greedy, backtracking, and dynamic programming strategies
- analyze iterative and recursive algorithms with respect to time and space
- describe the applications for a dictionary/map ADT, e.g., the application of symbol table
- Give representations for and operations on a binary tree, general tree, threaded tree, heap, binary search tree, B-tree, quadtree, and graph
- determine the order for a B-tree based on memory issues

Level 1: Knowledge and Comprehension

Level 1 outcomes are those in which the students have been exposed to the terms and concepts at a basic level and can apply basic definitions. The materials have been presented only at a superficial level.

Upon successful completion of this course, students will be able to:

- recognize the standard terms associated with particular data structures, e.g., head/tail, push/pop
- understand the big-O, Theta, Omega notations
- understand the concepts of time/space complexity
- understand the basic tree concepts
- understand the basic searching/sorting methods
- recognize the standard terms of graphs

ABET Student Outcomes for CSCI 3333 The list of ABET student outcomes related to this class is:

- (1) An ability to analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
- (2) An ability to design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
- (6) An ability to apply computer science theory and software development fundamentals to produce computing-based solutions.

ABET Student Outcomes for CMPE 3333 TBA

Tentative Course Calendar By Week

Week Of	Lecture Topics	Homework and Exams
8/26/2019	Syllabus, Programming Review (Chapter 1)	Homework 1
9/2/2019	Programming Review, continued (Chapter 1)	Homework 2
9/9/2019	Algorithm Analysis (Chapter 2)	Homework 3
9/16/2019	Sorting, part I (sections 7.1, 7.2 and 7.6)	Homework 4
9/23/2019	Sorting, part II (sections 7.5 and 7.7-7.9)	Homework 5
9/30/2019	Lists, Stacks and Queues (Chapter 3)	Homework 6
10/7/2019	Midterm Review	Midterm Exam
10/14/2019	Trees (sections 4.1-4.3)	Homework 7
10/21/2019	Hashing (Chapter 5)	Homework 8
10/28/2019	Heaps and Priority Queues (sections 6.1-6.3)	Homework 9
11/4/2019	Graph Algorithms, part I (sections 9.1)	Homework 10
11/11/2019	Graph Algorithms, part II (sections 9.2 and 9.6)	Homework 11
11/18/2019	Graph Algorithms, part III (sections 9.5)	Homework 12
11/25/2019	Graph Algorithms, part IV (sections 9.3)	Homework 13
12/2/2019	Final Review	

Note: Listed chapters and sections in the parentheses are relevant to the topics to be covered in the course but they don't necessarily cover all knowledge you are required to learn.

UTRGV Policy Statements

The UTRGV disability accommodation, mandatory course evaluation statement and sexual misconduct statement are required on all syllabi. Additional policy statements are optional, such as those covering attendance, academic integrity, and course drop policies.

STUDENTS WITH DISABILITIES: *Required on all syllabi.* Do not modify.

Students with a documented disability (physical, psychological, learning, or other disability which affects academic performance) who would like to receive academic accommodations should contact **Student Accessibility Services (SAS)** as soon as possible to schedule an

appointment to initiate services. Accommodations can be arranged through SAS at any time, but are not retroactive. Students who experience a broken bone, severe injury, or undergo surgery during the semester are eligible for temporary services.

Pregnancy, Pregnancy-related, and Parenting Accommodations

Title IX of the Education Amendments of 1972 prohibits sex discrimination, which includes discrimination based on pregnancy, marital status, or parental status. Students seeking accommodations related to pregnancy, pregnancy-related condition, or parenting (reasonably immediate postpartum period) are encouraged to contact Student Accessibility Services for additional information and to request accommodations.

Student Accessibility Services:

Brownsville Campus: Student Accessibility Services is located in 1.107 in the Music and Learning Center building (BMSLC) and can be contacted by phone at (956) 882-7374 or via email at ability@utrgv.edu.

Edinburg Campus: Student Accessibility Services is located in 108 University Center (EUCTR) and can be contacted by phone at (956) 665-7005 or via email at ability@utrgv.edu.

MANDATORY COURSE EVALUATION PERIOD: *Required on all syllabi. Do not modify.*

Students are required to complete an ONLINE evaluation of this course, accessed through your UTRGV account (<http://my.utrgv.edu>); you will be contacted through email with further instructions. Students who complete their evaluations will have priority access to their grades. Online evaluations will be available on or about:

Module 1	October 2 nd – 8 th
Module 2	November 27 th – December 3 rd
Full Fall Semester	November 14 th – December 4 th

ATTENDANCE: Recommended on all syllabi; may be modified by the instructor as long as it is not inconsistent with UTRGV policy.

Students are expected to attend all scheduled classes and may be dropped from the course for excessive absences. UTRGV's attendance policy excuses students from attending class if they are participating in officially sponsored university activities, such as athletics; for observance of religious holy days; or for military service. Students should contact the instructor in advance of the excused absence and arrange to make up missed work or examinations.

SCHOLASTIC DISHONESTY: Recommended on all syllabi.

As members of a community dedicated to Honesty, Integrity and Respect, students are reminded that those who engage in scholastic dishonesty are subject to disciplinary penalties, including the possibility of failure in the course and expulsion from the University. Scholastic dishonesty includes but is not limited to: cheating, plagiarism (including self-plagiarism), and collusion; submission for credit of any work or materials that are attributable in whole or in part to another person; taking an examination for another person; any act designed to give unfair advantage to a student; or the attempt to commit such acts. Since scholastic dishonesty harms the individual, all students and the integrity of the University, policies on scholastic dishonesty will be strictly

enforced (Board of Regents Rules and Regulations and UTRGV Academic Integrity Guidelines). All scholastic dishonesty incidents will be reported to Student Rights and Responsibilities.

SEXUAL MISCONDUCT and MANDATORY REPORTING: *Required on all syllabi. Do not modify.*

In accordance with UT System regulations, your instructor is a “Responsible Employee” for reporting purposes under Title IX regulations and so must report to the Office of Institutional Equity & Diversity (oiie@utrgv.edu) any instance, occurring during a student’s time in college, of sexual misconduct, which includes sexual assault, stalking, dating violence, domestic violence, and sexual harassment, about which she/he becomes aware during this course through writing, discussion, or personal disclosure. More information can be found at www.utrgv.edu/equity, including confidential resources available on campus. The faculty and staff of UTRGV actively strive to provide a learning, working, and living environment that promotes personal integrity, civility, and mutual respect that is free from sexual misconduct, discrimination, and all forms of violence. If students, faculty, or staff would like confidential assistance, or have questions, they can contact OVAVP (Office for Victim Advocacy & Violence Prevention) at 665-8287, 882-8282, or OVAVP@utrgv.edu.

COURSE DROPS: Recommended on all syllabi; may be modified by the instructor as long as it is not inconsistent with UTRGV policy.

According to UTRGV policy, students may drop any class without penalty earning a grade of DR until the official drop date. Following that date, students must be assigned a letter grade and can no longer drop the class. Students considering dropping the class should be aware of the “3-peat rule” and the “6-drop” rule so they can recognize how dropped classes may affect their academic success. The 6-drop rule refers to Texas law that dictates that undergraduate students may not drop more than six courses during their undergraduate career. Courses dropped at other Texas public higher education institutions will count toward the six-course drop limit. The 3-peat rule refers to additional fees charged to students who take the same class for the third time.

STUDENT SERVICES: Recommended on all syllabi.

Students who demonstrate financial need have a variety of options when it comes to paying for college costs, such as scholarships, grants, loans and work-study. Students should visit the Students Services Center (U Central) for additional information. U Central is located in BMAIN 1.100 (Brownsville) or ESSBL 1.145 (Edinburg) or can be reached by email (ucentral@utrgv.edu) or telephone: (888) 882-4026. In addition to financial aid, U Central can assist students with registration and admissions.

Students seeking academic help in their studies can use university resources in addition to an instructor’s office hours. University Resources include the Advising Center, Career Center, Counseling Center, Learning Center, and Writing Center. The centers provide services such as tutoring, writing help, critical thinking, study skills, degree planning, and student employment. Locations are:

Center Name	Brownsville Campus	Edinburg Campus
Advising Center AcademicAdvising@utrgv.edu	BMAIN 1.400 (956) 665-7120	ESWKH 101 (956) 665-7120

Career Center CareerCenter@utrgv.edu	BCRTZ 129 (956) 882-5627	ESSBL 2.101 (956) 665-2243
Counseling Center Counseling@utrgv.edu	EUCTR 109 (956) 665-2574	BSTUN 2.10 (956) 882-3897
Learning Center LearningCenter@utrgv.edu	BMSLC 2.118 (956) 882-8208	ELCTR 100 (956) 665-2585
Writing Center WC@utrgv.edu	BUBLB 3.206 (956) 882-7065	ESTAC 3.119 (956) 665-2538

Disclaimer This syllabus does not contain all regulations that relate to students. Contents in the syllabus may be changed by the instructor with advanced notice and/or agreement with the students. Any change will be kept to a minimum.