Disclaimer: This syllabus is not a binding contract or a binding agreement. When necessary, changes may be made, but will be announced in class, as the semester progresses.

Instructor

Zhixiang Chen  
Office: ENGR 3.293, Phone: 3520. Email: zhixiang.chen@utrgv.edu. Web: faculty.utpa.edu/zchen/ (to be updated soon)

Grader: Dustin Torres, dustin.torres01@utrgv.edu

A Note on Blackboard

I will use Blackboard for this class. Please make sure that you assess Blackboard often for course materials, assignments, and especially submitting your assignments. Assignment deadlines will be enforced by Blackboard.

Homework and lab exercise grader: Ernesto Bochas Jauregui.  
Email: ernesto.bochasjauregui01@utrgv.edu

Office Hours:

<table>
<thead>
<tr>
<th></th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuesday</td>
<td>04:30 PM -- 05:30 PM</td>
</tr>
<tr>
<td>Wednesday</td>
<td>10:30 AM -- 11:30 AM</td>
</tr>
<tr>
<td>Thursday</td>
<td>10:30 AM -- 11:30 AM</td>
</tr>
</tbody>
</table>

Class Schedule

- All courseware, including homework assignments and lab exercises will be available at BlackBoard.

Evaluations - Office of the Vice Provost for Faculty Affairs

Mandatory Course Evaluations period: Students are required to complete an ONLINE evaluation of this course, accessed through your UTPA account (https://my.utrgv.edu/); you will be contacted through email with further instructions. The evaluation window closes at the middle
night of the last day of the class. Students who complete their evaluations by the day will have priority access to their grades.

Course Description (UTPA Undergraduate 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: CSCI/CMPE 2380 and Math 3373.

Text and Materials


Prerequisites

You are expected to have completed CSCI/CMPE 2380. CSCI/CMPE 2380 covers advanced programming techniques, using principles of software engineering. Files, two-dimensional arrays, stacks, queues, linear and circular linked lists, tree data structures, and some sorting and searching method are also taught in CSCI/CMPE 2380.

Course 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 data efficiently in time and space. It covers the following topics:

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

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.

Exam, Assignment and Grading

There will be three exams (two midterm exams and one final) for a total of 75% of the grade, 6 assignments for 20% of the grade, and 5% of attendance. The letter grade will be determined as follows:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>90-100%</td>
</tr>
<tr>
<td>B</td>
<td>80-89%</td>
</tr>
<tr>
<td>C</td>
<td>70-79%</td>
</tr>
<tr>
<td>D</td>
<td>60-69%</td>
</tr>
<tr>
<td>F</td>
<td>0-59%</td>
</tr>
</tbody>
</table>

Note: Bonus projects will be announced during the semester.

Assignment Policies

- All assignments must be in the instructor's hands before class on the due date which will be specified on each assignment. Late assignments will be accepted up to one week with a one-time 20% late penalty.
- Assignments will be graded on the basis of correctness, quality of design, documentation, and style.
- Any assignment submitted without documentation will automatically receive a 20% deduction. Documentation, design, and style guidelines will be discussed as the semester progresses. No programming assignment will be graded which contains syntax errors.

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 quadratic and O(n log n) 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, quadtrees, 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:

(a) An ability to apply knowledge of computing and mathematics appropriate to the discipline

(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

(c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs

(h) Recognition of the need for and an ability to engage in continuing professional development

(i) An ability to use current techniques, skills, and tools necessary for computing practice.

(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

(k) An ability to apply design and development principles in the construction of software systems of varying complexity.

**ABET Student Outcomes for CMPE 3333**

The list of ABET student outcomes related to this class is:
(a) an ability to apply knowledge of mathematics, science, and engineering

(b) an ability to design and conduct experiments, as well as to analyze and interpret data

(c) an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability

(e) an ability to identify, formulate, and solve engineering problems

(i) a recognition of the need for, and an ability to engage in life-long learning

(j) a knowledge of contemporary issues

(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

Calendar of Activities

Include in this section a table or list that provides information for students regarding important dates, assignments or activities. The UTRGV academic calendar can be found at http://my.utrgv.edu at the bottom of the screen, prior to login. Some important dates for Spring 2017 include:

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nov. 1 (Tues.)</td>
<td>Registration Begins – Graduate Students</td>
</tr>
<tr>
<td>Nov. 14 (Mon.)</td>
<td>Registration Begins – Undergraduate Students</td>
</tr>
<tr>
<td>Jan. 11 (Wed.)</td>
<td>Payment Due</td>
</tr>
<tr>
<td>Jan. 13 (Fri.)</td>
<td>Waitlist Ends</td>
</tr>
<tr>
<td></td>
<td>Last day to withdraw (drop all classes) for a 100% refund</td>
</tr>
<tr>
<td>Jan. 16 (Mon.)</td>
<td>Martin Luther King Jr. Holiday. No classes.</td>
</tr>
<tr>
<td>Jan. 17 (Tues.)</td>
<td>Spring classes begin. Official First Class Day.</td>
</tr>
<tr>
<td>Jan. 17 – Jan. 23 (Tues. – Mon.)</td>
<td>Period to withdraw (drop all classes) for an 80% refund</td>
</tr>
<tr>
<td>Jan. 30 (Mon.)</td>
<td>Last day to add a class or register for Spring classes</td>
</tr>
<tr>
<td>Jan. 24 – Jan. 30 (Tues. – Mon.)</td>
<td>Period to withdraw (drop all classes) for a 70% refund</td>
</tr>
<tr>
<td>Jan. 31 – Feb. 6 (Tues. – Mon.)</td>
<td>Period to withdraw (drop all classes) for a 50% refund</td>
</tr>
<tr>
<td>Feb. 1 (Wed.)</td>
<td>Census Date (Last day to drop without it appearing on the transcript)</td>
</tr>
<tr>
<td>Feb. 7 – Feb. 13 (Tues. – Mon.)</td>
<td>Period to withdraw (drop all classes) for a 25% refund</td>
</tr>
<tr>
<td>Mar. 13 – Mar. 18 (Mon. – Sat.)</td>
<td>Spring Break. No classes.</td>
</tr>
<tr>
<td>Apr. 13 (Thurs.)</td>
<td>Last day to drop a class (grade of DR) or withdraw (grade of W)</td>
</tr>
<tr>
<td>Apr. 14 – Apr. 15 (Fri. – Sat.)</td>
<td>Easter Holiday. No classes.</td>
</tr>
<tr>
<td>May 4 (Thurs.)</td>
<td>Study Day. No classes.</td>
</tr>
<tr>
<td>May 5 – 11 (Fri. – Thurs.)</td>
<td>Final Exams</td>
</tr>
<tr>
<td>May 12 – 13 (Fri. – Sat.)</td>
<td>Commencement Exercises</td>
</tr>
<tr>
<td>May 15 (Mon.)</td>
<td>Grades Due</td>
</tr>
</tbody>
</table>
UTRGV Policy Statements

STUDENTS WITH DISABILITIES:
If you have a documented disability (physical, psychological, learning, or other disability which affects your academic performance) and would like to receive academic accommodations, please inform your instructor and contact Student Accessibility Services to schedule an appointment to initiate services. It is recommended that you schedule an appointment with Student Accessibility Services before classes start. However, accommodations can be provided at any time.

Brownsville Campus: Student Accessibility Services is located in Cortez Hall Room 129 and can be contacted by phone at (956) 882-7374 (Voice) or via email at ability@utrgv.edu.

Edinburg Campus: Student Accessibility Services is located in 108 University Center and can be contacted by phone at (956) 665-7005 (Voice), (956) 665-3840 (Fax), or via email at ability@utrgv.edu.

MANDATORY COURSE EVALUATION PERIOD:
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.

ATTENDANCE:
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 INTEGRITY:
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, 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 the Dean of Students.

SEXUAL HARASSMENT, DISCRIMINATION, and VIOLENCE:
In accordance with UT System regulations, your instructor is a “responsible employee” for reporting purposes under Title IX regulations and so must report any instance, occurring during a student’s time in college, of sexual assault, stalking, dating violence, domestic violence, or 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 in an environment free from sexual misconduct and discrimination.

**COURSE DROPS:**
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.