CSE 5311-001: Advanced Algorithms - Spring 2021

MW 4:00 - 5:20

Instructor:	Bob Weems, Associate Professor
Office:	627 ERB (weems@cse.uta.edu, http://ranger.uta.edu/~weems)
Hours:	MW 2:00 - 5:00 p.m. (on Teams)
GTA:	Contact information will be on my personal webpage
Prerequisites:	Algorithms & Data Structures (CSE 3318) Theoretical Computer Science (CSE 3315)
Objectives:	Deeper study of algorithms, data structures, and complexity classes.
Outcomes:	1. Exposure to more sophisticated analysis techniques, e.g. amortized complexity. 2. Exposure to specialized data structures and algorithms. 3. Exposure to models of algorithm design.
Textbook:	Cormen, Leiserson, Rivest, Stein, *Introduction to Algorithms*, *3rd ed.*, MIT Press, 2009. (Henceforth known as CLRS)
References:	S. Baase and A. Van Gelder, *Computer Algorithms, Introduction to Design and Analysis*, *3rd ed.*, Addison-Wesley, 2000.
	M. de Berg et.al., *Computational Geometry: Algorithms and Applications, 3rd ed.*, Springer-Verlag, 2010.
	A. Borodin and R. El-Yaniv, *Online Computation and Competitive Analysis*, Cambridge Univ. Press, 1998.
	E.D. Demaine and J. O'Rourke, *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*, Cambridge Univ. Press, 2007.
	P. Flajolet and R. Sedgewick, *Analytic Combinatorics*, Cambridge Univ. Press, 2009, http://algo.inria.fr/flajolet/Publications/AnaCombi/
	L. Fortnow, *The Golden Ticket: P, NP, and the Search for the Impossible*, Princeton Univ. Press, 2013.
	M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, 1979.
	G. Gonnet and R. Baeza-Yates, *Handbook of Algorithms and Data Structures, 2nd. ed.*, Addison-Wesley, 1991.
	R.L. Graham, D.E. Knuth, and O. Patashnik, *Concrete Mathematics*, Addison-Wesley, 1989.

C.M. Grinstead and J.L. Snell, *Introduction to Probability*,
http://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/book.html

D. Gusfield, *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*, Cambridge Univ. Press, 1997.

D. Gusfield and R. Irving, *The Stable Marriage Problem: Structure and Algorithms*, MIT Press, 1989.

D.S. Hochbaum, ed., *Approximation Algorithms for NP-Hard Problems*, PWS, 1997.

E. Horowitz and S. Sahni, *Fundamentals of Computer Algorithms*, Computer Science Press, 1978.

J. Kleinberg and E. Tardos, *Algorithm Design*, Addison-Wesley, 2006.

D.E. Knuth, *The Art of Computer Programming*, Vols. 1-4, Addison-Wesley.

V. Lifschitz, *Answer Set Programming*, Springer-Verlag,
https://www.cs.utexas.edu/users/vl/teaching/378/ASP.pdf

D. Manlove, Algorithmics of Matching Under Preferences, World Scientific, 2013,
https://ebookcentral-proquest-com.ezproxy.uta.edu/lib/utarl/detail.action?docID=1168176

M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis, 2nd ed.*, Cambridge, 2017.

C. Moore and S. Mertens, *The Nature of Computation*, Oxford, 2011.

R. Motwani and P. Raghavan, *Randomized Algorithms,* Cambridge Univ. Press, 1995.

J. O'Rourke, *Computational Geometry in C, 2nd ed.,* Cambridge Univ. Press, 1998.

C.H. Papadimitriou, *Computational Complexity*, Addison-Wesley, 1994.

R. Sedgewick, *Algorithms in C*, *Parts 1-5, 3rd ed.*, Addison-Wesley, 2003.

R. Sedgewick and P. Flajolet, *An Introduction to the Analysis of Algorithms, 2nd ed.*, Addison-Wesley, 2013.

A. Stepanov and P. McJones, *Elements of Programming*, Addison-Wesley, 2009.

A. Stepanov and D. Rose, *From Mathematics to Generic Programming*, Addison-Wesley, 2014.

Homework: Two assignments - NOT GRADED (solutions are available from the web page)

Grade: Your grade will be based on the following weights:

Exams:	80% (Test 1: 40%; Test 2: 40%, Friday, May 7, 2:00 - 4:30 p.m.)
Labs:	20% (Three labs, equal weight, submitted on Canvas)

Policies:

1. This is a hybrid course. Lectures will be available on Canvas, office hours will be on Teams, but exams will be face-to-face barring pandemic issues. (There is the possibility of a change to Lockdown browser/Respondus monitor exams through Canvas.)

2. Lecture notes, homework, old exams, lab assignment files and sample code for various algorithms are on the course web page `http://ranger.uta.edu/~weems/NOTES5311/cse5311.html`.

3. You are expected to have at least skimmed the new material by the day we start that material in class. The material will be covered in the order given later.

4. CHEATING - YOU ARE EXPECTED TO KNOW UNIVERSITY POLICIES. All cases of plagiarism will be processed through University channels outside the CSE department. `http://www.uta.edu/conduct/`

5. Any request for special consideration must be appropriately documented **in advance**. (Special consideration does not include giving a higher grade than has been earned.)

6. Late programs are penalized according to the following schedule. Labs will be due at 3:45 PM on the due date. After the due time, assistance will be limited.

Degree of lateness	Penalty
Up to 3:45 next day	10 pts
Up to 3:45 two days	20 pts
Up to 3:45 three days	40 pts
Up to 3:45 four days	80 pts

7. Each lab is graded on a 100-point scale as follows:

			Some Issues
a.	Output/Code	60%	If you know that your program has problems, you should let the GTA know what parts are functional. Test cases that demonstrate the limited functionality are useful.
b.	Internal Comments	6%	Beginning of file including `main()` should identify the assignment and who you are, along with giving a high-level description. Each function: identify each argument, describe processing, and each `return`. You may reference notes and text. Excess line-by-line comments are not needed, but the processing for each iteration of a (significant) loop should be explained.
c.	Modularity	6%	Functions are used appropriately. `main()` is kept simple.

d.	Structure	6%	Code is not unnecessarily complicated or long. It is often better to rewrite code rather than patching several times.
e.	Names	6%	Should indicate the purpose of the function, variable/field, or type. Cute or misleading names will be penalized.
f.	Spacing	6%	Indenting, blank lines, placement of {}. Be consistent.
g.	Generality	10%	Program is not unnecessarily limited.

All programs must be written in standard ***C*** to compile and execute on `omega.uta.edu`. Execution on other platforms (e.g. Visual Studio, Code::Blocks) does not assure compliance.

You are responsible for correctly submitting each programming assignment on Canvas.

No points will be awarded for programs that do not compile. *Points for b-g will not be awarded to submissions that are not substantially complete and perform* ***significant*** *processing.* Submissions not reflecting the algorithmic problem-solving techniques discussed in the lab handout will not receive credit.

8. If you require a reasonable accomodation for a disability, please contact me no later than the second week of this semester. Further details are available at `http://www.uta.edu/disability/`

9. Occasional class-wide email messages (e.g. weather situations, clarifications) may be sent to the addresses recorded by MyMav.

Course Outline

Starred (*) topics are not in CLRS

0. Selective review of dynamic programming
1. Mathematical Preliminaries
    - Recurrences - Master Method (4.5-4.6.1)
    - Probability and Randomized Algorithms (5)
2. Binary Search Trees
    - Red-Black Trees - Review (13)
    - AVL Trees*
    - Treaps (problem 13-4)
    - Augmenting Data Structures (14)
    - Optimal Binary Search Trees (15.5)
    - Self-Organizing Linear Search (Computing Surveys*, problem 17-5)
    - Self-Adjusting Binary Search Trees (Splay trees/amortized analysis) (JACM)*
3. Amortized Analysis (17)
4.a. Priority Queues - Review (6.5)
    - Binary Trees, Binary Heaps, d-heaps*, Leftist Heaps*
    - Binomial Heaps (problem 19-2)
    - Pairing Heaps*
5. Hashing
    - Review (11.2-11.4)

- Brent's Rehash*, Cuckoo Hashing*
- Perfect Hashing (11.5)
- Bloom Filters*

6. Medians/Selection (9.3)
7. Disjoint Sets (union-find trees) (21)

TEST 1

8. Minimum Spanning Trees (23)
   - Brief review of Prim
   - Review of Kruskal's Algorithm and extension to detecting non-unique MST
   - Boruvka's Algorithm*
9. Max-Flow/Bipartite Matching (26)
   - Ford-Fulkerson - review, maximum capacity paths*
   - Push-relabel methods
   - Vertex and edge connectivity*
   - Hopcroft-Karp matching (problem 26-6)
10. Matching Under Preferences*
    - Bipartite with Two-Sided Preferences (stable marriages, hospitals/residents)
    - Bipartite with One-Sided Preferences (house allocation)
11. Intractability (34, 35)
    - Sample Intractable Problems
    - Complexity Classes
    - Reductions
    - Polynomial-Time Approximation
12. Matrices
    - Strassen's Matrix Multiplication (4.2)
    - Binary Matrix Multiplication and Four Russians Trick*
13. Computational Geometry (33)
    - Fundamental Predicates
    - Closest Pairs
    - Convex Hulls
    - Sweepline Algorithms
    - Plane Partitions and Point Location*
    - Smallest Enclosing Disk*
    - Euclidean MST/Voronoi Diagram/Delaunay Triangulation*
14. Sequences
    - Pattern Preprocessing Search
      - Rabin-Karp Algorithm (32.2)
      - Knuth-Morris-Pratt Algorithm (32.4)
    - Text Preprocessing - Suffix Arrays*
    - Longest Common Subsequences
      - Dynamic Programming - Review and Linear Space* Version
      - Four Russians for LCS*
      - By Reduction to Longest Strictly Increasing Subsequence*

TEST 2

Calendar - with subject numbers from course content

January				February				March			
18	MLK	20	Syllabus	1		3	2.	1	7.	3	8.
25	0.	27	1.	8	3.	10		8		10	Exam 1
				15	4.a	17		15	SPRING	17	BREAK
				22	5.	24	6.	22	9.	24	
								29	10.	31	

April				May			
5	11.	7		3		7	Exam 2
12		14	12.				
19	13.	21					
26	14.	28					

April 2 is last day to drop; submit requests to major advisor prior to 4:00 p.m.

Messages/disclaimers/fine print from our sponsor:

Mandatory Face Covering Policy
All students and instructional staff are required to wear facial coverings while they are on campus, inside buildings and classrooms. Students that fail to comply with the facial covering requirement will be asked to leave the class session. If students need masks, they may obtain them at the Central Library, the E.H. Hereford University Center's front desk or in their department. Students who refuse to wear a facial covering in class will be asked to leave the session by the instructor, and, if the student refuses to leave, they may be reported to UTA's Office of Student Conduct.

Attendance: At The University of Texas at Arlington, taking attendance is not required but attendance is a critical indicator in student success. Each faculty member is free to develop his or her own methods of evaluating students' academic performance, which includes establishing course-specific policies on attendance. As the instructor of this section, I expect regular attendance. However, while UT Arlington does not require instructors to take attendance in their courses, the U.S. Department of Education requires that the University have a mechanism in place to mark when Federal Student Aid recipients "begin attendance in a course." UT Arlington instructors will report when students begin attendance in a course as part of the final grading process. Specifically, when assigning a student a grade of F, faculty report the last date a student attended their class based on evidence such as a test, participation in a class project or presentation, or an engagement online via Canvas. This date is reported to the Department of Education for federal financial aid recipients.

Emergency Exit Procedures: Should we experience an emergency event that requires us to vacate the building, students should exit the room and move toward the nearest exit. When exiting the building during an emergency, one should never take an elevator but should use the stairwells. Faculty members and instructional staff will assist students in selecting the safest route for evacuation and will make arrangements to assist individuals with disabilities.

Student Success Programs
UT Arlington provides a variety of resources and programs designed to help students develop academic skills, deal with personal situations, and better understand concepts and information related to their courses. Resources include tutoring by appointment, drop-in tutoring, etutoring, supplemental instruction, mentoring (time management, study skills, etc.), success coaching, TRIO Student Support Services, and student success workshops. For additional information, please email resources@uta.edu, or view the Maverick Resources website.

The IDEAS Center (https://www.uta.edu/ideas/) **(**2nd Floor of Central Library) offers **FREE** tutoring and mentoring to all students with a focus on transfer students, sophomores, veterans and others undergoing a transition to UT Arlington. Students can drop in or check the schedule of available peer tutors at www.uta.edu/IDEAS, or call (817) 272-6593.

Institution Information
UTA students are encouraged to review the below institutional policies and informational sections and reach out to the specific office with any questions. To view this institutional information, please visit the Institutional Information page (https://resources.uta.edu/provost/course-related-info/institutional-policies.php) which includes the following policies among others:

- Drop Policy
- Disability Accommodations
- Title IX Policy
- Academic Integrity
- Student Feedback Survey
- Final Exam Schedule