CSE 2320-001: ALGORITHMS & DATA STRUCTURES

Fall 2006: TR 11:00-12:20, Nedderman 229

Instructor: Bob Weems, Associate Professor
Office: 344 Nedderman Hall (weems@uta.edu, http://reptar.uta.edu)
Hours: TR 1:00-2:30 pm

GTA:
Office:
Email: @cse.uta.edu
Hours:

Prerequisites: C++ programming (CSE 1325) - see policy 1.
CSE 2315

Objectives: In future design situations, students will be capable of developing, applying, and evaluating algorithmic solutions.

Outcomes:

1. Understanding of classic approaches to algorithm design - decomposition, dynamic programming, and greedy methods.
2. Understanding of particular algorithms and data structures that have wide applicabilty.
3. Understanding of basic algorithm analysis concepts by applying math skills to worst-case and expected time using recurrences and asymptotic notation.
4. Improved programming skills - especially data structures, recursion, and graphs.

Textbook: Cormen, Leiserson, Rivest, Stein, *Introduction to Algorithms, 2nd ed.,* MIT Press, 2001.

References: S. Baase and A. Van Gelder, Computer Algorithms: *Introduction to Design and Analysis, 3rd ed.*, Addison-Wesley, 2000.

G.H. Gonnet and R. Baeza-Yates, *Handbook of Algorithms and Data Structures, 2nd ed.*, Addison-Wesley, 1991.

R. Sedgewick, *Algorithms in C*, *Parts 1-5, 3rd ed.*, Addison-Wesley, 2001.

Readings: Indicated on calendar later in syllabus.

Homeworks: 3 homeworks, with answers, are available on the course web page.

Grade: Based on the following weights:

Exams: 80% divided evenly among 3 exams. Exam 3: December 12, 11:00-1:30

Programs: 20% divided evenly among four assignments.

Policies:

1. During the class period on Tuesday, September 5, a 25-question multiple choice exam on C++ will be given. Please bring pencils. In addition, you will be allowed to bring reference materials. No sharing. No calculators. BE ON TIME! Students with at least 18 correct answers will be allowed to remain registered. Topics include:

    a. Classes and Objects
    b. Input/Output
    c. Composition
    d. Inheritance
    e. Polymorphism
    f. Exception Handling
    g. Templates, including the "string" data type

2. Regular attendance is expected. You are expected to know lecture contents and announcements. I reserve the right to have surprise quizzes, each quiz being 2% of the semester grade taken from the 80% allocated to exams.

3. Lecture notes and sample code for various algorithms are on the course web page `http://reptar.uta.edu/NOTES2320/cse2320.html`.

4. You are expected to have read the assigned readings by the specified date. Lectures will review and augment the material, but will also consider exercises from the book.

5. CHEATING - YOU ARE EXPECTED TO KNOW UNIVERSITY POLICIES. If you are suspected of cheating, the matter must go through university channels outside of the CSE Department.

    a. **Academic Integrity Policy:** It is the policy of the University of Texas at Arlington to uphold and support standards of personal honesty and integrity for all students consistent with the goals of a community of scholars and students seeking knowledge and truth. Furthermore, it is the policy of the University to enforce these standards through fair and objective procedures governing instances of alleged dishonesty, cheating, and other academic/non-academic misconduct.

       You can assume responsibility in two ways. **First**, if you choose to take the risk associated with scholastic dishonesty and any other violation of the Code of Student Conduct and Discipline, you must assume responsibility for your behaviors and accept the consequences. In an academic community, the standards for integrity are high. **Second**, if you are aware of scholastic dishonesty and any other conduct violations on the part of others, you have the responsibility to report it to the professor or assistant dean of students/director of student judicial affairs. The decision to do so is another moral dilemna to be faced as you define who you are. Students who violate University rules on scholastic dishonesty are subject to disciplinary penalties, including the possibility of failure in the course and dismissal from the University. Since dishonesty harms the individual, all students, and the integrity of the University, policies on scholastic dishonesty will be strictly enforced.

b. **Statement on Ethics, Professionalism, and Conduct of Engineering Students**: The statement is attached. Failure to sign the statement will result in 1) programming assignments to not be accepted, i.e. late penalty and 2) failure on exams.

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

7. Late programs are penalized according to the following schedule. LABS ARE DUE AT 10:45 AM ON THE DUE DATE, NOT MIDNIGHT. After the due time, assistance will not be provided.

Degree of lateness	Penalty
Up to 10:45 next day	10 pts
Up to 10:45 two days	30 pts
Up to 10:45 three days	60 pts

8. Each student will have available *one* 2-day, no-penalty extension that may be applied to *one* of the lab assignments. To use your extension you must send an email to a grader ***before*** the due time. An acknowledgement will be sent.

   RESUBMISSIONS BEFORE THE DUE TIME ARE PENALIZED 10 POINTS EACH. NO RESUBMISSIONS AFTER THE DUE TIME.

9. Each lab is graded 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%	Identification at beginning of program, including the assignment, who you are, how to compile on OMEGA, and 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 compile and execute on OMEGA and be written in C (or C++). You may develop your code on another system, but you must port it to OMEGA. Details for program submission will be included with each assignment.

You are responsible for correctly sending each programming assignment to the GTA as an attachment. (cc: yourself)

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.*

10. Phone calls. I will not answer my phone during office hours if someone else is in my office. After the third ring the call is switched to the CSE office, so please leave a message with the secretary.

11. GTA duties:

    a. Provide first-level of assistance for homeworks and labs.
    b. Grade programs and short-answer test problems.

12. Instructor duties:

    a. Lecture.
    b. Guidance
    c. Tests - preparation and grading long-answer test problems.
    d. Special consideration.
    e. Design homework and programming assignments.

13. Please email the following information to `???@cse.uta.edu` by Thursday, September 7:

    a. Name.
    b. Additional email addresses.
    c. Special circumstances affecting your performance.
    d. (Optional) What do you hope to gain from this course?

Course Content (in chronological order)

1. Algorithmic Concepts (Chap. 1 & 2), Binary Search, Merging
2. Growth of Functions (Chap. 3)
3. Summations (Appendix A)
4. Recurrences (4.1, 4.2)
5. Heapsort/Priority Queues (Chap. 6)
6. Quicksort (7.1, 7.2)
   Lower Bounds (8.1)
   Counting and Radix Sorts (8.2, 8.3)

Test 1: Items 1.-6.

7. Stacks/Queues (10.1)
8. Linked Lists (10.2, 10.3)
9. Rooted Trees (10.4)
   Binary Search Trees (12.1-12.3)
10. Red-Black Trees (13.1-13.4)
11. Hashing (11.1-11.4)

Test 2: Items 7.-11.

12. Graph Representations (22.1)
    Search (22.2, 22.3)
    Search-Based Algorithms (22.4-22.5)
13. Minimum Spanning Trees (23.1, 23.2 - omit Kruskal's algorithm)
14. Shortest Paths (24.3, 25.2)
15. Network Flows and Bipartite Matching (26.1-26.3)
16. Dynamic Programming (Chap. 15)
17. Greedy Algorithms (16.1-16.3)
18. KMP String Search (32.4)

Test 3: Items 12.-18.

Calendar - with subject numbers from course content

	August/September				October		
29	Syllabus	31	1.	3	Exam 1	5	7./8.
5	C++ exam	7	2.	10	8./9.	12	10.
12	3.	14	4.	17	???	19	
19		21	5.	24	11.	26	12.
26	6.	28		31	Exam 2		

	November				December		
		2					
7	13.	9	14.	5	18.	7	
14		16	15.	12	Exam 3		
21		23	HOLIDAY				
28	16.	30	17.				

November 3 is the last day to drop.

Statement of Ethics
Student Confirmation
(CSE 2320-001, Fall 2006)

The following is an excerpt from the College of Engineering's statement on Ethics, Professionalism, and Conduct of Engineering Students. The notes are modifications appropriate for Computer Science and Engineering courses. Read the statement carefully, sign it, and return it to your instructor. Additional copies of this statement can be obtained from your instructor or the Computer Science and Engineering office.

Statement on Ethics, Professionalism, and Conduct of Engineering Students
College of Engineering
The University of Texas at Arlington

The College cannot and will not tolerate any form of academic dishonesty by its students. This includes, but is not limited to 1) cheating on examination, 2) plagiarism, or 3) collusion.

Definitions:

A. **Cheating on an examination** includes:

1. Copying from another's paper, any means of communication with another during an examination, giving aid to or receiving aid from another during an examination;
2. Using any material during an examination that is unauthorized by the proctor;
3. Taking or attempting to take an examination for another student or allowing another student to take or attempt to take an examination for oneself.
4. Using, obtaining, or attempting to obtain by any means the whole or any part of an unadministered examination.

B. **Plagiarism** is the unacknowledged incorporation of another's work into work which the student offers for credit.
C. **Collusion** is the unauthorized collaboration of another in preparing work that a student offers for credit.
D. Other types of **academic dishonesty** include using other student's printouts from the ACS labs or students' disk, etc.

1. The use of the source code of another person's program, even temporarily, is considered **plagiarism**.
2. Allowing another person to use your source code, even temporarily, is considered **collusion**.
3. Use of another person's source code with your modification is considered **plagiarism.**
4. Taking material verbatim (without quoting the source) for reports and/or presentations is considered **plagiarism**
5. For this class, the specific exceptions given below are not considered scholastically dishonest acts:
   Discussion of the algorithm and general programming techniques used to solve a problem

The penalty assessed for cheating on a given assignment will be twice the weight of the assignment and will include notification of the proper authorities as stipulated in the **UTA Handbook of Operating Procedures** and on the web at ***http://www2.uta.edu/discipline***

You may be entitled to know what information UT Arlington (UTA) collects concerning you. You may review and have UTA correct this information according to procedures set forth in UT System BPM #32. The law is found in sections 552.021, 552.023 and 559.004 of the Texas Government Code.

I have read and I understand the above statement.

Student's signature: ____

Student's name (printed): ____

Student's ID number: ____