SI335 Computer Algorithms

Course Policy, Spring AY19

Coordinator: LT M. D. Moeller, x3-6872, moeller@usna.edu

Course Description: Presents techniques for designing and analyzing computer algorithms including divide and conquer, dynamic programming and greedy methods. Introduces classic algorithms for problems such as searching and sorting, graph analysis, file compression and cryptology.

Credits: 3-0-3

Learning Objectives:

1. Demonstrate an understanding of a variety of classic standard algorithms. (supports student outcomes 1, CS-6)
2. Employ a variety of standard techniques to devise efficient algorithms. (supports student outcome CS-6)
3. Compare and analyze the performance of algorithms. (supports student outcomes 1, CS-6)
4. Understand the local and global impact of algorithm design and performance, especially as it relates to cryptography, and the implied ethical issues and responsibilities. (supports student outcome 4)
5. Develop effective written arguments regarding algorithmic performance. (supports student outcome 3)

Student Outcomes:

1. Analyze a complex computing problem and apply principles of computing and other relavent disciplines to identify solutions.
3. Communicate effectively in a variety of professional contexts.
4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.

CS-6. Apply computer science theory and software development fundamentals to produce computing-based solutions.

Textbook(s):
Dasgupta, Papadimitriou, and Vazirani. *Algorithms*. McGraw Hill, 2008 (optional).
Cormen, Leiserson, Rivest, and Stein. *Introduction to Algorithms*, 3rd ed. MIT Press, 2009 (optional).

Extra Instruction: Extra instruction (EI) is strongly encouraged and should be scheduled by email with the instructor. EI is not a substitute lecture; students should come prepared with specific questions or problems.

Collaboration: The guidance in the Honor Concept of the Brigade of Midshipmen and the Computer Science Department Honor Policy must be followed at all times. See `www.usna.edu/CS/resources/honor.php`. Specific instructions for this course: For the purposes of this class, we make a distinction between two types of what would be called collaboration in the Department Honor Policy:

- **Discussion** means talking between classmates about how to tackle a particular (written or programming) problem. Discussion is done away from a computer, and it does not involve looking at each

other's solutions at all. Nothing written or programmed which will be or has been submitted is used as part of a discussion. If anything is written down during the discussion, it is destroyed at the end of the discussion.

- **Collaboration** means actually sitting down and working together on a problem. This is beyond discussion because the written materials are not destroyed. However, each student must still **write up their own solution in their own words**. Note that if you are copying down a classmate's solutions in the few minutes prior to the start of class, this clearly does not meet the definition of "collaboration", and you will be committing a violation of this course policy.

Discussion is allowed on any homework or programming project, as long as it is clearly documented. However, *collaboration* is only allowed on homework. The only help permitted on problem sets is within a student's group. At all times, help from any human not currently enrolled in SI335 is never permitted. This includes other students not in the class, parents, friends, other faculty members, and even online humans (e.g., asking questions in an online forum or discussion board). The only exceptions are the current instructor and SI335 MGSP leader(s).

Any resources other than those on the course website must also be documented clearly and specifically. For example, if a website is used, the full address of the web page in question must be listed, along with what in the assignment that page was used for. Just saying "Wikipedia" is not enough, but something like "en.wikipedia.org/wiki/Big_O_notation example used in problem 2" would be good.

All collaboration and outside sources should always be cited. The same rules apply for giving and receiving assistance. If you are unsure whether a certain kind of assistance or collaboration is permitted, you should assume it is not, work individually, and seek clarification from your instructor.

Classroom Conduct: The section leader will record attendance and bring the class to attention at the beginning of each class. If the instructor is late more than 5 minutes, the section leader will keep the class in place and report to the Computer Science department office. If the instructor is absent, the section leader will direct the class. Drinks are permitted, but they must be in reclosable containers. Food, alcohol, smoking, smokeless tobacco products, and electronic cigarettes are all prohibited. Cell phones must be silent during class.

Late Policy: Penalties for late submission of graded work may vary among courses or from semester to semester, but they will be the same for all sections of a given course. For *this* course:

- Late homeworks, problem sets and programming projects are not accepted.

Grading:
Homework. Homework is due at the start of the lecture after it is assigned, unless specified otherwise. With a planned excused absence, please have a classmate bring the assignment. With an unplanned absence (e.g. injury or illness), submit at the next class attended. Students may collaborate on take-home questions as long as it is clearly documented.
Problem Sets. More extensive non-programming problems, that require significant time, effort, and care, will be assigned in problem sets due every 4 weeks or so. Problem sets may be completed in groups up to a certain size. Solutions will be presented orally, in person with the instructor. Each group is also required to hand in a written outline of their solutions. Each group member is responsible to understand and explain any solution. More details on this process programming projects in this class. At least one will be individual, and at least one will be group. Details on electronic submission will be published along with the

projects and must be followed. Programming projects in this class will require not only coding skills, but also algorithmic development skills, and therefore starting early is extremely important. For individual programming projects, informal discussion between students is allowed, but not collaboration (see definitions below). All discussion and outside resources must be clearly and specifically documented. Group programming assignments, will be combined with Problem Sets, and the same Problem Set rules on collaboration and discussions apply — i.e. only between fellow group members.

Exams. There will be two midterm exams and one final exam. The final exam will be cumulative. Students will be allowed one "crib sheet" for the midterms and two for the final. Obviously, no discussion or collaboration of any sort will be allowed during exams.

	6 weeks	12 weeks	16 weeks	Final
Homework	25%	21%	21%	15%
Problem Sets	22%	22%	28%	20%
Programming Projects	22%	22%	28%	20%
Midterm Exams	31%	35%	23%	15%
Final Exam	0%	0%	0%	30%
Total	100%	100%	100%	100%