Version: September 22, 2008

COURSE SYLLABUS
605.421.71 Foundations of Algorithms
Fall 2008 Johns Hopkins University

Instructor: Mr. John E. Boon, Jr.

NOT AVAILABLE PERIODS FALL 2008

Monday	Tuesday	Wednesday	Thursday	Friday
12:00-3:30pm 4:30-7:10pm	**6:00-9:00pm**	**6:00-9:00pm**	**1:00-3:00pm 6:00-9:00pm**	**11:30am-1:00pm**

Mr. John E. Boon, Jr. is the course instructor for these class meetings. Mr. Boon was previously an Associate Professor of Computer Science and Chairperson, Department of Mathematics and Computer Science, Hood College. He recently re-joined the Hood College faculty. Mr. Boon is an Operations Researcher for RAND. Mr. Boon has an operations research and computer science consulting practice.

You are encouraged at any time to phone my cell number. I encourage you to contact me if you have any questions at all. I may be able to clarify requirements as well as suggest additional resources and strategies for a given problem. Be sure to (1) state your name, (2) state what I may do for you, (3) state your phone number twice, (4) state the hours during which I may return your call if you leave a voice message.

My complete list of contact information follows:

My JHUe-mail addresses: mailto:jboonjr@apl.jhu.edu
WWW Home Pages: http://www.apl.jhu.edu/Notes/Boon/605421/
JHU Course Home Page References:
https://ptesrv.apl.jhu.edu/pages/existing_homepages.php
http://www.apl.jhu.edu/Courses/cs/
http://www.apl.jhu.edu/Classes/Classes.html
Phone: **(301) 606-4115 (cell)**

Course Description:

This follow-on course to data structures (e.g., 605.202) provides a survey of computer algorithms, examines fundamental techniques in algorithm design and analysis, and develops problem-solving skills required in all programs of study involving computer science. Topics include advanced data structures (red-black and 2-3-4 trees, union-find), recursion and mathematical induction, algorithm analysis and computational complexity (recurrence relations, big-O notation, NP-completeness), sorting and searching, design paradigms (divide and conquer, greedy heuristic, dynamic programming, amortized analysis), and graph algorithms (depth-first and breadth-first search, connectivity, minimum spanning trees, network flow). Advanced topics are selected from among the following: randomized algorithms, information retrieval, string and pattern matching, and computational geometry.

Prerequisites:

Working knowledge of data structures and Java, C++, or C. Undergraduate course(s) in algorithms and data structures (for example: JHU courses 605.202, 600.226, 600.363). It is expected that all students are familiar with linked lists, stacks, queues, heaps, binary search trees, and hash tables. Students must also be familiar with standard sorting and searching algorithms. Prior coursework in discrete mathematics (sets, graphs, counting, combinatorics, and mathematical induction) is helpful to reducing your study time at the start of this course.

Computer Requirement:

Students must be proficient in either Java or C++. Students must have an email address and must make that address known to the course instructor as soon as practical after the start of the course. Students may use their own computing resources (hardware/software tools) or may arrange to access the necessary tools through the student services entities of JHU. Class examples will be compiled and processed with Java 2 and the GNU port of gcc for 32-bit Windows. Students must have access to the Internet, have a web browser and the Acrobat Reader application.

Instructional Objectives:

Students completing this course will demonstrate mastery of the mathematical tools used in analyzing the performance and efficiency of computer algorithms.

Students completing this course will be capable of determining the correctness of deterministic and non-deterministic algorithms.

Students completing this course will be able to apply the techniques and methods of algorithm analysis.

Students completing this course will master several important data structures useful in scientific programming and sorting/searching (Red-Black trees, B-Trees, Fibonacci Heaps, Disjoint Sets) and their associated creation and maintenance algorithms.

Students completing this course will master several important data structures useful in operations research, computer science, and telecommunications (graphs and networks) and their associated creation, maintenance, and application algorithms.

Students completing this course will master several important data structures useful in string search and their associated creation, maintenance, and application algorithms.

Required Text:

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Second Edition*. The MIT Press/McGraw-Hill Book Company, 2001.

Optional Resource Texts:

Sara Baase and Allen Van Gelder. *Computer Algorithms: Introduction to Design and Analysis*. Addison-Wesley, third edition, 2000.

Ellis Horowitz and Sartaj Sahni. *Fundamentals of Computer Algorithms*. Computer Science Press, 1983.

Donald E. Knuth. *Fundamental Algorithms*, volume 1 of *The Art of Computer Programming*. Addison-Wesley, 1968. third edition, 1997.

Donald E. Knuth. *Seminumerical Algorithms*, volume 2 of *The Art of Computer Programming*. Addison-Wesley, 1969. third edition, 1998.

Donald E. Knuth. *Sorting and Searching*, volume 3 of *The Art of Computer Programming*. Addison-Wesley, 1973. second edition, 1998.

William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1993.

Robert Sedgewick and Philippe Flajolet. *An Introduction to the Analysis of Algorithms*. Addison-Wesley. 1996.

Robert E. Tarjan. *Data Structures and Network Algorithms (Fourth Printing)*. CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, 1983.

Internet Resources, Library Reserve Texts, Information, Articles, and Videos:

You are encouraged to obtain a JCARD for accessing JHU library resources. You can access library resources from off-campus as well as on-campus computers (http://www.library.jhu.edu/). See instructions at:

http://www.library.jhu.edu/services/computing/remoteaccess.html

It is your responsibility to check the class WWW pages no later than Noon each class day for updates and class information (http://www.apl.jhu.edu/Notes/Boon/605421/). I tend to post extensive class information on my class pages (important news items about the upcoming class, items important to the upcoming class lecture, notes, links, problems, solutions, projects, programming resources). I promise to have the page updated by Noon the day of class for any information you may need to bring with you to class that day.

You are encouraged to consult resources in Dr. Dobb's Journal, http://www.ddj.com for algorithm topical articles on algorithms in C/C++ and Java. If you are interested in general searches for journal articles, please consult http://www.ingenta.com/.

Reading Assignments:

You will be expected to read assigned material before the class at which it will be discussed. You do not serve yourself by being exposed to these ideas for the first time in lecture. Keep notes in the margins or in a notebook to remind you to ask them during lectures later. You will be expected to search out and read additional material on topics during this course.

Homework Assignments:

These assignments will involve research and computation. Research will be done to investigate and study algorithms described in popular and scientific literature. Computation will be done to analyze the time and space complexity of algorithms and to "walk-through" some of the algorithms we won't be translating into source programs. Although the Internet provides more examples of algorithms than you may ever wish to wade through, you are encouraged to think and assimilate, rather than regurgitate. The objective of researching algorithms this semester will be to compare their design and performance and for this you need your mind, not just books and the Internet.

Please note that in all assignments, you must cite all references and sources you use (see links on course WWW page for proper form of references of WWW pages). I will not tolerate the use of quoted or paraphrased information, algorithms, or code segments from published sources or from others without appropriate citation.

Programming Assignments:

You may write programs in Java or C++. You may use your own compiler (e.g., Sun, GNU, Borland, Microsoft) or resources on JHU computers.

I will provide you with a code header that you will complete for each program. You will be required to develop test cases for each program you submit. When you deliver your work, submit:

1. CD formatted for Window/Intel computer:
    a. program source code, including completed code header documentation http://www.apl.jhu.edu/Notes/Boon/Common/Java_Links.shtml or http://www.apl.jhu.edu/Notes/Boon/Common/CPP_Links.shtml)
    b. data file(s) required by your program;
2. printed listing(s) of your program(s);
3. printed test case statements and printed test case results.

Failure to submit all required materials will result in a reduction of grade on the assignment.

Exams:

Two exams are planned. The final exam will contain questions that concentrate on the last half of the course but will assume mastery of material in previous half. These exams are intended to assess your mastery of the key concepts and relationships investigated each half of the course. All exams are equally weighted. Exams will be take-home, open-book, and due the following week no later than a time specified on the exam. Late exams will be penalized one letter grade.

Graded Assignments:

Distribution of weights in grading:

Homework	(50%) – each week
Programs	(15%) – a few during the semester
Exams	(35%) – mid-term and final exams

Homework and programming assignments turned after the due date will not be accepted. Up to one homework or programming assignments may be marked as excused for extraordinary situations; no points will be assigned and the assignment will not be used in computing your final grade. Each homework assignment is equally important to your grade (that is, some homework assignments contain more problems than others but all are scored on a percent of possible basis resulting in a % earned out of 100% available score).

Grading Policy:

I will award partial credit for work done even if the result is incorrect, but this implies that you show all your intermediate work and clearly label your answer. I will deduct points for answers that do not make any sense at all; you should always check your work, even work done using the computer. The explanation of your work is as important as the work itself -- do not concentrate on the programs or mathematics and ignore the importance of clear descriptions of what you did and what it means.

The following grading scale will be used 100-90 % = A, 89-80 %=B, 79-70 %=C, 69 % - Below=F. Letter grades approximately imply:

- A - complete understanding of the topic plus additional insight, creative, or other indicators of advancement beyond complete understanding of the topic;
- B - mastery of the basic material;
- C - attempt to complete the basic material but significant gaps in understanding and mastery exist;
- F - failure to complete basic material successfully.

I do not grade on a curve.

Attendance:

Class attendance is essential. Significant material is covered at each of our class meetings this semester. I may excuse absences if I am notified.

Academic Standards:

Students are reminded of sections in the current catalog: Academic Ethics and Violations of Academic Integrity[1]. All assignments this semester are individual effort assignments. It is a violation of the rules of academic conduct in this class for individuals to collaborate with other individuals, whether or not they are members of this class, on assignments, unless specifically directed that such collaboration is allowed for a specific assignment by the professor.

During class, you are expected to concentrate on and contribute to class presentations, lecture, and group discussion. Small group discussions that distract from the ability of others in class to adequately concentrate upon class presentations, lecture, and group discussion will not be tolerated. Use of laptop computers during class will be restricted to note taking and computer-based activities as assigned by the instructor. Internet surfing, writing of other papers, or use of programs not specifically related to the lecture will not be tolerated.

1 http://catalog.epp.jhu.edu/content.php?catoid=12&navoid=248#acad_ethi

TENTATIVE COURSE SCHEDULE
605.421.71 Foundations of Algorithms, Fall 2007, JHU

Class meets Monday 4:30pm-7:10pm, A&R 211, Montgomery County Center
Available for questions Monday TBD, A&R 2nd Floor Lounge, Montgomery County Center

Class	Date	Objectives	Text
1	09/08	What is Algorithm Analysis?	Chapters 1 & 2
2	09/15	What is Algorithm Analysis?	Chapters 1 & 2
3	09/22	[No class lecture]	
4	09/29	Mathematics for Algorithm Analysis	Chapter 3;
5	10/06	Recurrences	Chapter 4
6	10/13	Recurrences	Chapter 4
7	10/20	NP-Completeness & Approximation Algorithms	Chapters 34-35
		Mid-term exam to be distributed 10/20 Due 10/27 at 4:30 pm (take-home mid-term exam)	
8	10/27	Probabilistic Analysis, Randomized Algorithms & Sorting	Chapter 5-8
9	11/03	Medians and Order Statistics	Chapter 9
10	11/10	Hash Tables, Binary Trees, Red-Black Trees, and Augmenting Data Structures	Chapters 10-14
11	11/17	Dynamic Programming & Greedy Algorithms	Chapters 15-16
12	11/24	Amortized Analysis; Basic Graph Algorithms; MST Algorithms	Chapters 17; Chapters 22-23
13	12/01	Shortest Path Algorithms	Chapters 24 -25
		Final exam to be distributed 12/03 Due 12/10 at 6:00 pm (take-home final exam)	
14	12/08	Linear Programming	Chapter 29