COP 5536 Advanced Data Structures

Summer 2015

Instructor: Dr. Sartaj K Sahni (sahni@cise.ufl.edu)
Office Hours: By Appointment
Phone: 352-450-1490
Office: E536 Computer Science and Engineering Building

TA (Office Hours):
Adeesha Malavi(adeeshaw@ufl.edu): Tuesday 12.50 to 1.50 pm and Thursday: 3 to 4 pm at E309

Tentative Class Schedule:

Week 1(May 11 - May 15) : Lecture 1 - Lecture 4	
Week 2 (May 18 – May 22) : Lecture 5 - Lecture 8	
Week 3 (May 25 – May 29) : Lecture 9 - Lecture 12	
Week 4 (Jun 1 – Jun 5) : Lecture 13 - Lecture 16	
Week 5 (Jun 8 – Jun 12) : Lecture 17 - Lecture 20	**Exam 1: Date :June 11 (Lec 1-13)**
Week 6 (Jun 15 - Jun 19) : Lecture 21 - Lecture 24	
Week 7 (Jun 22 - Jun 26) : Summer Break – No Classes	
Week 8 (Jun 29 – July 3) : Lecture 25 - Lecture 28	
Week 9 (July 6 - July 10) : Lecture 29 - Lecture 32	**Exam 2: Date :July 9 (Lec 14-28)**
Week 10 (July 13 – July 17) : Lecture 33 - Lecture 35	
Week 11 (July 20 – July 24) : Lecture 36 - Lecture 39	
Week 12 (July 27 – July 31) : Lecture 40	**Exam 3: Date :August 6 (Lec 29-40)**

Note:- The exam dates are tentative subject to room availability. These exams will most probably be held in the evening.

Pre-requisites:

You should know the following:
1. C, C++, or Java. Since the text is in C++, you should at least be able to read C++.
2. Algorithm analysis methods (in particular asymptotic complexity).
3. Basic data structures such as stacks, queues, linked lists, trees, and graphs. Basic sorting methods such as insertion sort, heap sort, merge sort, and quick sort.

Course Requirements:

There will be two assignments and three exams. The exams will be closed book exams. The programming assignment(s) may be done in any high level language. Some possibilities are C, C++, and Java. Please have the use of any other language approved by the instructor or the TA. C++ is the preferred language.

Grading:

Exam 1: 25% (TBD)
Exam 2: 25% (TBD)
Exam 3: 25% (TBD)
Assignment 1: 20%
Assignment 2: 5%

Course Outline

The specific topics are:

1. Amortized complexity
2. External sorting & tournament trees
3. Buffering
4. Run generation & optimal merge patterns
5. Priority queues and merging
6. Leftist trees, Binomial heaps and Fibonacci heaps
7. Pairing heaps
8. Double ended priority queues
9. Static and dynamic weighted binary search trees
10. AVL-trees
11. Red-black trees
12. Splay trees
13. B-, B+- and B*-trees
14. Tries and digital search trees
15. Tries and packet forwarding
16. Suffix trees
17. Bloom filters
18. Segment trees
19. Interval trees
20. Priority search trees
21. k-d trees
22. Quad and oct trees
23. BSP trees
24. R-trees

Lecture	Content
1	Amortized complexity.
2	Amortized Complexity.

3	Introduction to external sorting.
4	Introduction to external sorting.
5	Selection trees & k-way merging.
6	Run generation.
7	Optimal merging of runs.
8	Buffering.
9	Double-ended priority queues. General methods.
10	Double-ended priority queues. Interval heaps.
11	Leftist trees.
12	Binomial heaps.
13	Binomial heaps.
14	Fibonacci heaps.
15	Pairing heaps.
16	Dictionaries.
17	Optimal binary search trees.
18	AVL trees.
19	AVL trees
20	Red-black trees.
21	Red-black trees.
22	B-Trees.
23	B-trees.

24	B+ and B*-trees.
25	Splay Trees.
26	Splay Trees.
27	Binary Tries.
28	Compressed Binary Tries.
29	High-order Tries.
30	Tries and Packet Forwarding.
31	Suffix Trees.
32	Bloom Filters.
33	Segment Trees.
34	Interval Trees.
35	Priority Search Trees.
36	Priority Search Trees.
37	Multidimensional Search Trees.
38	Quad Trees.
39	BSP Trees.
40	R-trees.

Course Policies:

1. Every student is expected to follow the University of Florida Honor Code. (See http://www.dso.ufl.edu/STG/default.html)
2. Handouts, assignments, solutions, and others will be posted on Sakai. Students should check Sakai regularly, at least once per week.
3. When submitting homework for grading, your answers should be written neatly and contain an explanation that is clear and reasonably concise.
4. All requests for re-grading must be made to a teaching assistant within 2 weeks of the date that scores are posted.