COP 5536 Advanced Data Structures

Fall 2011

Instructor: Dr. Sartaj K Sahni (sahni@cise.ufl.edu)
Office Hours: 2:00 – 3:00 PM Monday and Wednesday
Office: CSE 536
Phone: 352-450-1490
Email: sahni@cise.ufl.edu

TA:
Jianmin Chen (jichen@cise.ufl.edu): Period 5 and 1:50-4:50pm on Friday
Yian Zhou (yian@cise.ufl.edu): Period 9 and 10 on Monday and Wednesday
Tania Mishra (tmishra@cise.ufl.edu): Period 3 and 4 on Tuesday and Thursday
Office: E309 Computer Science and Engineering Building

Class Schedule:
Week 1(Aug 22 - Aug 26) : Lecture 1 - Lecture 3
Week 2 (Aug 29 – Sep 2) : Lecture 4 - Lecture 6
Week 3 (Sep 5 – Sep 9) : Lecture 7 - Lecture 8 Holiday: Sep 5
Week 4 (Sep 12 - Sep 16) : Lecture 9 - Lecture 11
Week 5 (Sep 19 - Sep 23) : Lecture 12 - Lecture 14
Week 6 (Sep 26 - Sep 30) : Lecture 15 - Lecture 16 Exam 1: Sep 30
Week 7 (Oct 3 - Oct 7) : Lecture 17 - Lecture 19
Week 8 (Oct 10 - Oct 14) : Lecture 20 - Lecture 22
Week 9 (Oct 17 - Oct 21) : Lecture 23 - Lecture 25
Week 10 (Oct 24 - Oct 28) : Lecture 26 - Lecture 28
Week 11 (Oct 31 – Nov 4) : Lecture 29 - Lecture 29 Exam 2: Nov 2, Holiday: Nov 4
Week 11 (Nov 7 – Nov 11) : Lecture 30 - Lecture 31 Holiday: Nov 11
Week 11 (Nov 14 – Nov 18) : Lecture 32 - Lecture 34
Week 11 (Nov 21 – Nov 25) : Lecture 35 - Lecture 36 Holiday: Nov 25
Week 11 (Nov 28 – Dec 2) : Lecture 37 - Lecture 39
Week 11 (Dec 5 – Dec 9) : Lecture 40 - Lecture 40

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 three 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 and Tentative Dates:

Exam 1: Friday, Sep 30	25%
Exam 2: Wednesday, Nov 2	25%
Exam 3: Thursday, Dec 15	25%
Three Assignments: TBA	25%

Course Outline

The specific topics and associated readings are:

1. Amortized complexity (Web)
2. External sorting & tournament trees (Sections 7.10.1, 7.10.2, and 5.8)
3. Buffering (Section 7.10.3)
4. Run generation & optimal merge patterns (Huffman trees) (Sections 7.10.4 and 7.10.5)
5. Priority queues and merging (Section 5.6)
6. Leftist trees, Binomial heaps and Fibonacci heaps (Sections 9.2, 9.3, and 9.4)
7. Pairing heaps (Section 9.5)
8. Double ended priority queues (Sections 9.6 and 9.7, Web)
9. Static and dynamic weighted binary search trees (Section 10.1)
10. AVL-trees (Section 10.2)
11. Red-black trees (Section 10.3)
12. Splay trees (Section 10.4)
13. B-, B+- and B*-trees (Sections 11.1-11.3)
14. Tries and digital search trees (Sections 12.1-12.3)
15. Tries and packet forwarding (Section 12.5)
16. Suffix trees (Section 12.4)
17. Bloom filters (Section 8.4)
18. Segment trees (readings)
19. Interval trees
20. Priority search trees (readings)
21. k-d trees (readings)
22. Quad and oct trees (readings)
23. BSP trees
24. R-trees

Lecture	Content	Reading

1	Amortized complexity.	Web resource.
2	Amortized Complexity.	Web resource.
3	Introduction to external sorting.	Section 7.10.1.
4	Introduction to external sorting.	Section 7.10.1.
5	Selection trees & k-way merging.	Sections 5.8 and 7.10.2.
6	Run generation.	Section 7.10.4.
7	Optimal merging of runs.	Section 7.10.5.
8	Buffering.	Sections 7.10.3.
9	Double-ended priority queues. General methods.	Sections 9.6, 9.7, and Web resource.
10	Double-ended priority queues. Interval heaps.	Sections 9.7.
11	Leftist trees.	Section 9.2.
12	Binomial heaps.	Section 9.3.
13	Binomial heaps.	Section 9.3.
14	Fibonacci heaps.	Section 9.4.
15	Pairing heaps.	Section 9.5.
16	Dictionaries.	Section 5.7.
17	Optimal binary search trees.	Section 10.1.
18	AVL trees.	Section 10.2.
19	AVL trees	Section 10.2.
20	Red-black trees.	Section 10.3.
21	Red-black trees.	Section 10.3.
22	B-Trees.	Sections 11.1 and 11.2
23	B-trees.	Sections 11.1 and 11.2.
24	B+ and B*-trees.	Section 11.3.
25	Splay Trees.	Section 10.4.
26	Splay Trees.	Section 10.4.
27	Binary Tries.	Section 12.1.
28	Compressed Binary Tries.	Section 12.2.
29	High-order Tries.	Sections 12.3 and Web Resource.
30	Tries and Packet Forwarding.	Section 12.5.
31	Suffix Trees.	Section 12.4.
32	Bloom Filters.	Section 8.4.

33	Segment Trees.	
34	Interval Trees.	
35	Priority Search Trees.	References.
36	Priority Search Trees.	References.
37	Multidimensional Search Trees.	References.
38	Quad Trees.	References.
39	BSP Trees.	
40	R-trees.	

Course Policies:

1. Every student is expected to follow the University of Florida Honor Code. (See 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. For distance students, you will have a 7 day window following the in-class exam date to schedule and complete exam with their proctor.