

CS136: Data Structures & Advanced Programming

Spring 2011 Schedule

Revised May 9, 2011 by Prof. McGuire.

Lecture:	MWF	11 – 11:50 am	Hollander 040	Exam dates, no-class days, and labs are guaranteed. Lectures and reading will adjust by a few days throughout the semester. Hyperlinks are underlined.
Lab:	Wed.	1 – 4 pm	TCL 217	
Office Hrs: (136) (General)	Sun.	3 – 5 pm	TCL 217	
	Mon.	1:15 – 2:15 pm	TCL 308	

Chapter reading assignments are in Bailey's free online textbook Bailey, *Java Structures*. Labs are available Monday or Tuesday morning and are due 10 pm the following Monday. All exams are 90-minute take-home. I do not grant extensions or accept late work.

MONDAY	WEDNESDAY	FRIDAY
<div style="border: 1px solid black; padding: 2px; display: inline-block;">Jan 31st</div>	<div style="border: 1px solid black; padding: 2px; display: inline-block;">Feb 2nd</div> <i>No class—Thursday schedule</i>	4th 1 <i>First day of CS136</i> INTRODUCTION <ul style="list-style-type: none"> • Data Structures • Interface vs. Implementation • Programming • Honor Code
7th 2 JAVA SYNTAX <ul style="list-style-type: none"> • Declarations • Statements (control flow) • Expressions • Some keywords • <u>2010 Sample Lab 2</u> <u>Sum.java, CardInterface.java, Card.java</u> Reading: Ch 1	9th 3 Lab: Silver Dollar FIXED-SIZE ARRAYS <ul style="list-style-type: none"> • Pointer aliasing • Array syntax • Iteration strategies • 2D Arrays • Shuffling example <u>Card2.java, PokerHand.java</u>	11th 4 CONTRACTS <ul style="list-style-type: none"> • Interfaces • Preconditions • Postconditions • Assertions • A dynamic array interface Reading: Ch 2
14th 5 DYNAMIC ARRAYS <ul style="list-style-type: none"> • Growing an Array • Shrinking an Array • Asymptotic analysis • Amortized analysis Reading: Ch 3	16th 6 Lab: Mazes RECURSION <ul style="list-style-type: none"> • L-Systems • Iterating without loops • The Towers of Hanoi • Mathematical Induction <u>Recursion.java</u> Reading: Ch 4	18th <i>No lecture – Winter Carnival.</i>
21st <i>No lecture – Prof. McGuire out of town.</i>	23rd <i>No lecture – Prof. McGuire out of town. Lab with April.</i> Lab: Recursion Reading: Ch 5	25th 7 LINKED LISTS <ul style="list-style-type: none"> • Node interface • Iterative implementation • Recursive implementation • Redundant state for performance <u>List.java, ListRecursive.java, FastList.java</u> Reading: Ch 9
28th 8 SORTING I <ul style="list-style-type: none"> • Comparable interface • Insertion sort • Scientific publications Reading: Ch 6	<div style="border: 1px solid black; padding: 2px; display: inline-block;">Mar 2nd</div> 9 Lab: Sorting SORTING II <ul style="list-style-type: none"> • Stable sorting • Merge sort • Quick sort • Conclusions <u>List.java</u> (now with merge sort!) Reading: Ch 7	4th 10 ORDERED STRUCTURES <ul style="list-style-type: none"> • Natural comparators • Ordered list <u>OrderedList.java</u> Reading: Ch 11

MONDAY	WEDNESDAY	FRIDAY	
7th 11 SEARCHING <ul style="list-style-type: none"> Ordered vector Linear Search Binary Search Stacks Rewriting Recursion <u>OrderedVector.java</u> , <u>StackExample.java</u>	9th 12 <u>Lab: P.S. It's Just a Stack</u> LINEAR STRUCTURES I <ul style="list-style-type: none"> Array stack List stack PostScript language <i>Continued in Lab...</i> Reading: Ch 10	11th 13 LINEAR STRUCTURES II <ul style="list-style-type: none"> Infix to postfix Queues Circular buffers 	
14th 14 BACKTRACKING <ul style="list-style-type: none"> Review Session (<u>Sample Exam</u>) Mazes and labyrinths Backtracking 	16th 15 Exam 1: Wed 12 pm – Fri 11 am SOLVING THE MAZE <u>maze.zip</u> <i>No lab</i>	18th 16 ITERATORS, TREES <ul style="list-style-type: none"> Iterator Reverse Iterator Efficient search on a list? Binary search tree Other hierarchies: File system, genealogy, expressions, decision trees <u>Vector.java</u> , <u>ReverseVectorIterator.java</u> , <u>AutoReverseIterator.java</u> , <u>OrderedList.java</u> , <u>Tree.java</u> Reading: Ch 8	
21st <i>Spring Break</i>	23rd <i>Spring Break</i>	25th <i>Spring Break</i>	
28th <i>Spring Break</i>	30th <i>Spring Break</i>	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>Apr 1st</td></tr></table> <i>Spring Break</i>	Apr 1st
Apr 1st			
4th 17 BINARY TREES <ul style="list-style-type: none"> Tree terminology Maintaining consistency Compute depth Compute height Displaying <u>BinaryTree.java</u> Reading: Ch 12	6th 18 <u>Lab: The Two Towers</u> TREE TRAVERSAL <ul style="list-style-type: none"> in-order iterator pre-order iterator post-order iterator level-order iterator <u>BinaryTree.java</u> , <u>Inorder.java</u> , <u>Preorder.java</u> , <u>Postorder.java</u> , <u>Levelorder.java</u>	8th 19 TREE SEARCH <ul style="list-style-type: none"> <code>getDepth</code>, <code>getHeight</code> <code>getDegree</code> <code>isFull</code> <code>contains</code>: Depth-first Breadth-first Naïve pathfinding: revisiting the labyrinth 	
11th 20 ARTIFICIAL INTELLIGENCE <ul style="list-style-type: none"> Tree quiz Adversarial mazes Payoff example: <ul style="list-style-type: none"> move backward vs. turn around Learning 	13th 21 <u>Lab: Darwin's Maze</u> SHARING INFORMATION <ul style="list-style-type: none"> Tree quiz <code>static</code> members Complex static initializers Counting instances Communal memory 	15th 22 PRIORITY QUEUES <ul style="list-style-type: none"> Tree quiz Motivation: <code>removeMin</code> An inefficient priority queue Binary heap Vector heap <code>percolateUp</code> and <code>add</code> Reading: Ch 13	
18th 23 HEAPS <ul style="list-style-type: none"> <code>removeMin</code> <code>pushDownRoot</code> Exam review <u>Heap.java</u>	20th 24 Exam 2: Wed 12 pm – Fri 11 am HEAP SORT <ul style="list-style-type: none"> Skew heap Heapsort Exam review <i>No lab (exam)</i>	22nd 25 BINARY SEARCH TREES: INSERT <ul style="list-style-type: none"> Binary search Concealing the nodes <code>BST.contains()</code> <code>BST.insert()</code> <u>BST.java</u> Reading: Ch 14	

MONDAY		WEDNESDAY		FRIDAY	
25th	26	27th	27	29th	28
BINARY SEARCH TREES: REMOVE <ul style="list-style-type: none"> Remove <u>BST.java</u> (revised)		Lab: <u>Lexicon</u> GRAPH CONCEPTS <ul style="list-style-type: none"> Vertices Edges Paths Reachability Flow Cycles 		GRAPH IMPLEMENTATIONS <ul style="list-style-type: none"> Matrix List Implicit Reading: Ch 16	
May 2nd	29	4th	30	6th	31
PATH-FINDING <ul style="list-style-type: none"> DFS Dijkstra's Algorithm Fibonacci Heap A* 		Lab: <u>Natural Selection</u> <i>(with partners!)</i> ARRAY MAP <ul style="list-style-type: none"> Association Array Map Ordered Array Map <u>ArrayMap.java</u>		HASH MAP I <ul style="list-style-type: none"> Speculative allocation Hash functions Modular arithmetic Collisions <u>SimpleHashMap.java</u>	
9th	32	11th	33	13th	34
HASH MAP II <ul style="list-style-type: none"> Open addressing Probing Double hashing Chaining Good hash functions <u>HashMap.java</u> Reading: Ch 15		FINAL EXAM REVIEW <i>No lab</i>		<i>Last Day of Classes</i> BEYOND "ADVANCED" PROGRAMMING <ul style="list-style-type: none"> Machine organization Concurrent processing Distributed computing Type inference Dynamic types Macros Structures for many dimensions 	