Computer Science 228
Introduction to Data Structures
Fall 2016
Tentative Schedule

Note: Dates are subject to change. Remember to refresh your browser to see updates to this document.

Updated on 16 August 2016

Week 1: August 22–26

- *Aug. 22:* Introduction: course staff; course policies; overview of the course. Algorithms and abstract data types.
- *Aug. 24:* Modularity and abstraction. Objects and classes; overview of encapsulation, inheritance, and polymorphism. Java interfaces.
- *Aug. 26:* Inheritance: interface implementation and class extension; overriding. Polymorphism: compile-time types, runtime types, and dynamic binding.

Week 2: August 29 – September 2

- *Aug. 29:* More on polymorphism and dynamic binding. Abstract classes.
- *Aug. 31:* Access modifiers. The `Object` class. Primitive types versus object types. Object equality and the `equals()` method: deep versus shallow comparison.
- *Sep. 2:* Copying and cloning, copy constructors. Shallow copying vs. Deep copying.

Week 3: September 5–9

- *Sep. 5:* University holiday; *no class.*
- *Sep. 7:* Review of exception handling. Analysis of algorithms and big-O notation.
- *Sep. 9:* More on big-O. Array equality.

Week 4: September 12–16

- *Sep. 12:* Binary search; further examples of algorithm analysis.
- *Sep. 14:* Sorting; selection sort and insertion sort; merge sort.
- *Sep. 16:* Analysis of partition and quicksort. Stability of sorting.

Week 5: September 19–23

- *Sep. 19:* Generic programming. The `Comparable` interface.
- *Sep. 21:* The `Comparator` interface. Wild cards.
- *Sep. 23:* Sorting with generics.

Week 6: September 26 – September 30

- *Sep. 26:* Review for exam 1.
  - **Exam 1:** 6:45pm-8:00pm,
- *Sep. 28: No class.*
- *Sep. 30:* More on generics: raw types and erasure. The Java `Collections` framework and `Iterator` interface.

Week 7: October 3–7

- *Oct. 3:* Array implementation of collections.
- *Oct. 5:* Introduction to linked lists. Singly- and doubly-linked lists.
- *Oct. 7:* Linked list implementation of collections.

Week 8: October 10–14

- *Oct. 10:* The `List` interface and the `ListIterator` interface.
- *Oct. 12:* Doubly-linked list implementation of the List interface.
- *Oct. 14:* Implementation of the List interface, iterators. Array-based implementation.

Week 9: October 17–21

- *Oct. 17:* Stacks and applications.
- *Oct. 19:* Postfix and infix notation. Infix-to-postfix conversion.
- *Oct. 21:* Convex hulls. Graham's scan.

Week 10: October 24–28

- *Oct. 24:* Queues. Introduction to trees. Child-sibling trees.
- *Oct. 26:* Tree traversals and arity.
- *Oct. 28:* Binary trees. Expression tree.

Week 11: October 31– November 4

- *Oct. 31:* Review for exam 2
- *Nov. 1:*
  - **Exam 2:** 6:45pm-8:00pm
- *Nov. 2: No class.*
- *Nov. 4:* Sets. Binary search trees. Implementation of BSTs

Week 12: November 7–11

- *Nov. 7:* Continuing BST implementation; successor.
- *Nov. 9:* Splay trees.
- *Nov. 11* The `Map` interface.

Week 13: November 14–18

- *Nov. 14:* Hash table. Hash functions; `hashcode()`; `equals()` revisited.
- *Nov. 16:* Priority queues. Introduction to heaps — pseudocode and big-O analysis.
- *Nov. 18:* Heap implementation.

Week 14: November 21–25

Thanksgiving break; no class.

Week 15: November 28 – December 2

- *Nov. 28:* Graphs.
- *Nov. 30:* Graph representations: adjacency matrix and adjacency list. Breadth-first search.
- *Dec. 2:* Depth-first search.

Week 16: December 5 – 9 (Dead week)

- *Dec. 5:* Topological sort.
- *Dec. 7:* Shortest path. Dijkstra's algorithm.
- *Dec. 9:* Review for final exam.

Week 17: December 12–16

Final Exam week; no class.