

CS367 Announcements

Wed, Aug 7th, 2013

- H8 due Fri Aug, 9th 6:00pm
- Final Tomorrow in class

Last Time

- Sorting (cont.)

Today

- Finish Sort
- Final Review
- Course Evaluations

Course Overview

Data Structures (DS) and Abstract Data Types (ADTs)

Algorithms

Complexity

Course Overview (cont.)

Data Structures (DS) and Abstract Data Types (ADTs)

- linear
 - 1 predecessor (except first) and 1 successor (except last)
 - DS: array, chain of list nodes, circular data structures
 - ADTs: list, linked list, stack, queue, deque, hashtable?
- hierarchical (non-linear possible)
 - 1 predecessor (except root) and 1 or more successors (except leaves)
 - DS: tree nodes, heap
 - ADTs: trees (general, binary, search, etc.), priority queue
- graphical (non-linear/non-hierarchical possible)
 - 0 or more predecessor and successors, specify source/start
 - DS: vertexes/nodes stored in list/map/set, edges stored in adjacency matrix/lists
 - ADT: graph (undirected/directed, weighted, etc.)
 - ADTs: trees (general, binary, search, etc.), priority queue
- position oriented
 - list
 - stack
 - queue
 - general tree
 - binary tree
- key-value oriented
 - stored list
 - binary search tree
 - balanced search trees (red black)
 - map, set
 - graph
- hybrid
 - priority queue
 - hashtable

Course Overview (cont.)

Algorithms

- operations on ADTs/data structures
- recursion
- searching
- hashing
- sorting

Complexity

- time complexity, space complexity
- worst-case, average case, best case
- determining complexity for methods/algorithms
 - non-recursive
 - recursive

Course Overview (cont.)

Java Concepts

- reference types
- command-line arguments
- “generic” programming
- exceptions
- iterators
- interfaces
- Java Collections framework