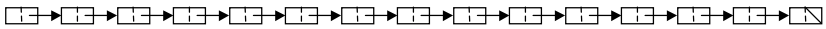
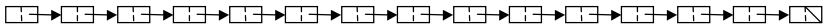


CSE 373: Wrap-up

Brad Chamberlain
University of Washington
Spring 1999



Observation

- Data is an attribute common to all programs
  - programs *process, manipulate, store, display, gather*
  - data may be *information*, *numbers*, *images*, *sound*
- Each program must decide how to store data
- Choice influences program at every level:
  - execution speed
  - memory requirements
  - maintenance (debugging, extending, etc.)

ADT Tensions

Ideal: a fast, elegant ADT that uses little memory

Generates tensions:

- time *vs.* space
- performance *vs.* elegance
- generality *vs.* simplicity
- one operation's performance *vs.* another's

The Myth of ADTs

Not a perfect black box:

- knowing how an ADT will be used can lead to a good choice of implementation
- also, knowledge of an ADT's implementation may change how a client uses it

But... ADTs are still a useful concept

Use motivates design

Course Goals

- To introduce several standard data structures
- To teach how data structures are evaluated
- To determine when each structure is useful

- To give you the ability to design, build, and evaluate your own data structures

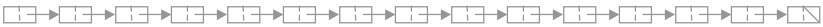
Course Topics Overview

Data Structures

- Storage: × *operations*
  - List/Sorted List
  - Stack
  - Queue
  - Binary Search Tree
  - Hash Table
  - Heap

  `Insert()`
  `Find()`
  `Delete()`
  `FindMin()/FindMax()`
  `DeleteMin()/DeleteMax()`
  traversals (sorted vs. complete)
- Structural
  - Tree — *traversals*
  - Graph — *traversals, shortest paths, MSTs*
- Sparse Arrays — *storage, traversals*

Course Topics Overview



Algorithms

- categorization of algorithms:
  - recursive vs. iterative
  - primary vs. secondary effects
  - greedy vs. divide-and-conquer vs. other
  - NP problems
  - intractable problems
- types of algorithms
  - searching (linear, binary, using data structures)
  - selection (using data structures, quickselect)
  - sorting (insertion, shellsort, mergesort, quicksort, bucketsort)
  - statistical (median, mean, mode)