CSCI 136
Data Structures & Advanced Programming

Jeannie Albrecht
Lecture 18
March 21, 2014

Administrative Details

- Lab 6 due Tuesday after break
  - I think some TAs will be around on Sunday and Monday evening (look for an email with hours)

Last Time

- Finished up stacks
  - LIFO linear data structure

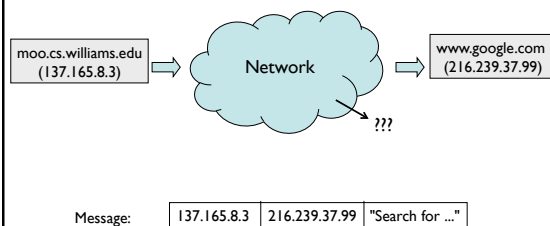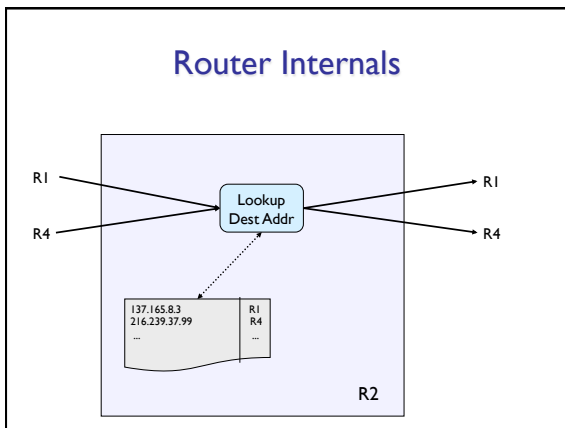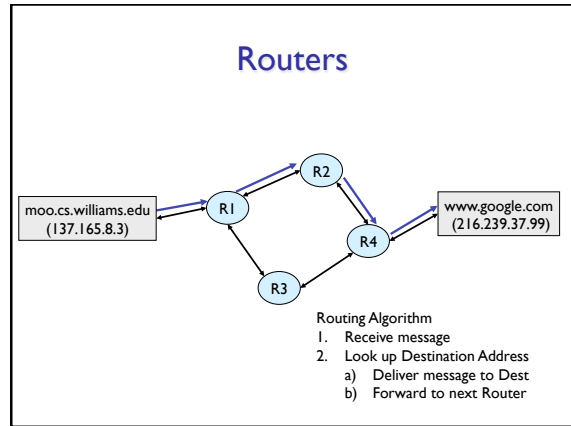- Learned about queues
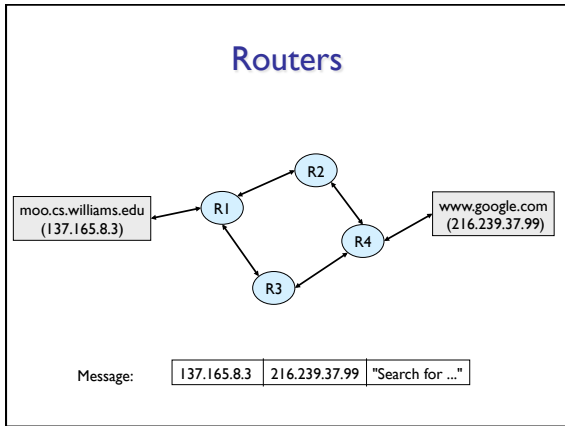 - FIFO linear data structure

---

Today's Outline

- Finish up queues
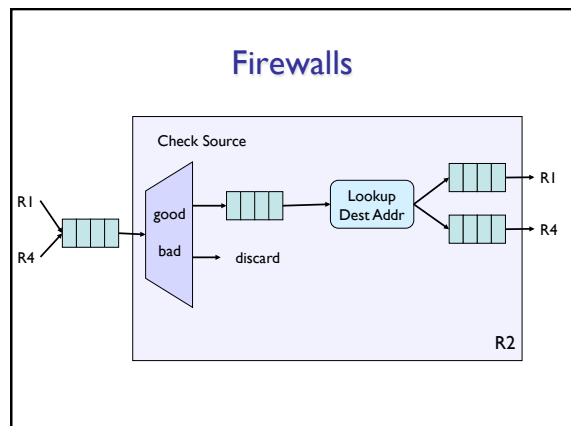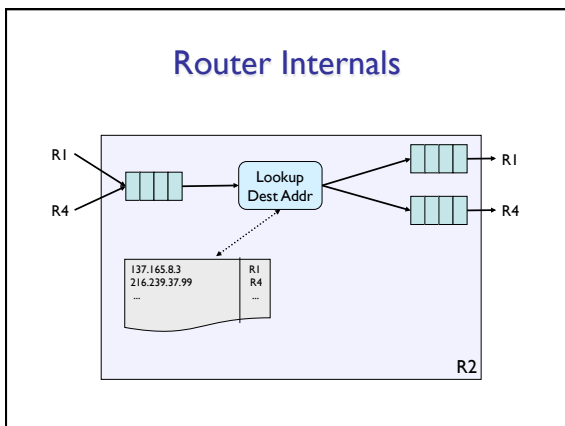- Maybe start talking about iterators

---

Routing With Queues

(Slides initially created by Stephen Freund)

The Network

| moo.cs.williams.edu (137.165.8.3) | → | Network | → | www.google.com (216.239.37.99) |

???

Message: | 137.165.8.3 | 216.239.37.99 | "Search for ..." |

Routers

moo.cs.williams.edu
(137.165.8.3)

R1

R2

R3

R4

www.google.com
(216.239.37.99)

Message: | 137.165.8.3 | 216.239.37.99 | "Search for ..." |

Routers

moo.cs.williams.edu
(137.165.8.3)

R1

R2

R3

R4

www.google.com
(216.239.37.99)

Routing Algorithm
1. Receive message
2. Look up Destination Address
   a) Deliver message to Dest
   b) Forward to next Router

Router Internals

R1

R4

Lookup
Dest Addr

R1

R4

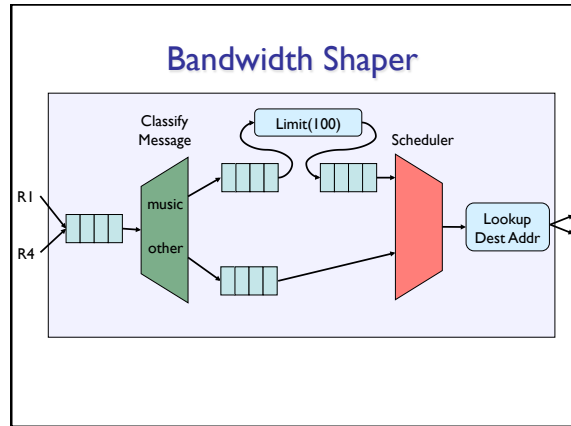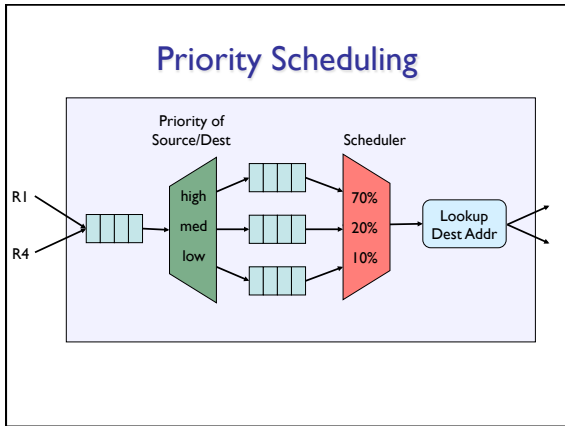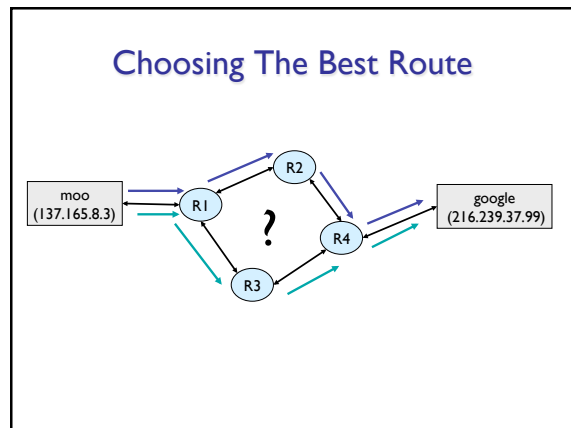| 137.165.8.3 | R1 |
| 216.239.37.99 | R4 |
| ... | ... |

R2

Buffering Messages

- There may be routing delays
  - Router receives messages faster than they can process and send -> causes *congestion*
  - Some links are slower than others
    - Common speeds: 10Mbs, 100Mbs, 1Gbs, 10Gbs
      - Wireless, satellite, optical fiber, cable, telephone line, ...
  - Hardware problems
- Want to be able to gracefully handle short-term congestion problems

Router Internals

R1

R4

Lookup
Dest Addr

R1

R4

137.165.8.3	R1
216.239.37.99	R4
...	...

R2

Firewalls

Check Source

R1

R4

good

bad

discard

Lookup
Dest Addr

R1

R4

R2

4/7/14

Priority Scheduling

Priority of Source/Dest — Scheduler

R1
R4

high
med
low

70%
20%
10%

Lookup Dest Addr

Bandwidth Shaper

Classify Message — Limit(100) — Scheduler

R1
R4

music
other

Lookup Dest Addr

More On Modular Routers

"The Click Modular Router", Eddie Koller and Robert Morris, Jr.

Choosing The Best Route

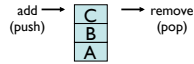moo (137.165.8.3)

R1
R2
R3
R4
?

google (216.239.37.99)

Choosing Routes

- Routers exchange information periodically
  - Attempt to route messages via "best" path to destination
  - Not easy to determine:
    - Network congestion varies (evening vs. morning)
    - Hardware added/removed or failures
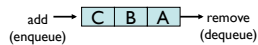- Dijkstra's algorithm (later)

Moving on...

3

Summary

- Linear Structures: stacks and queues
  - One method for adding items
  - One method for removing items
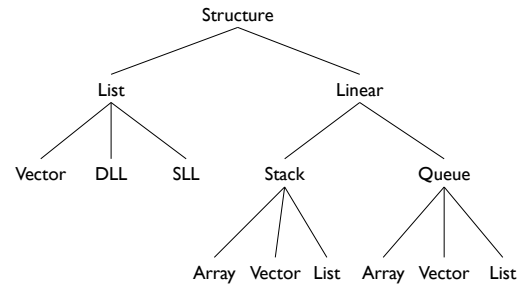 - Restricted interface
- Stack

add ⟶ (push) [C/B/A] ⟶ remove (pop)

- Queue

add ⟶ (enqueue) [C|B|A] ⟶ remove (dequeue)

Summary

Structure
- List
  - Vector
  - DLL
  - SLL
- Linear
  - Stack
    - Array
    - Vector
    - List
  - Queue
    - Array
    - Vector
    - List

Common Structure Operations

- `size()`
- `isEmpty()`
- `add()`
- `remove()`
- `clear()`
- `contains()`

- What's missing? (Recall our unsolved SLL "problem"…)

Visiting Data from Structure

- Write a method (numOccurs) that counts the number of times a particular Object appears in a structure

- Does this work on all structures?

Problems

- get(i) not defined on Linear structures
- get(i) may by slow on some structures
  - O(n) on SLL
  - So numOccurs = $O(n^2)$
- How do we process data in structures in a general, efficient way?
  - Must be data structure-specific for efficiency
  - Must always use some interface to make general

Iterators

- **Iterators** provide us with a way to cycle through elements of a structure in an efficient way
- An Iterator is an object that:
  - Provides generic methods to traverse elements
  - Abstracts away details of how to access structure
  - Has different implementations for different structures

4