

# Computer Networks

Instructor: Niklas Carlsson

Email: [niklas.carlsson@liu.se](mailto:niklas.carlsson@liu.se)

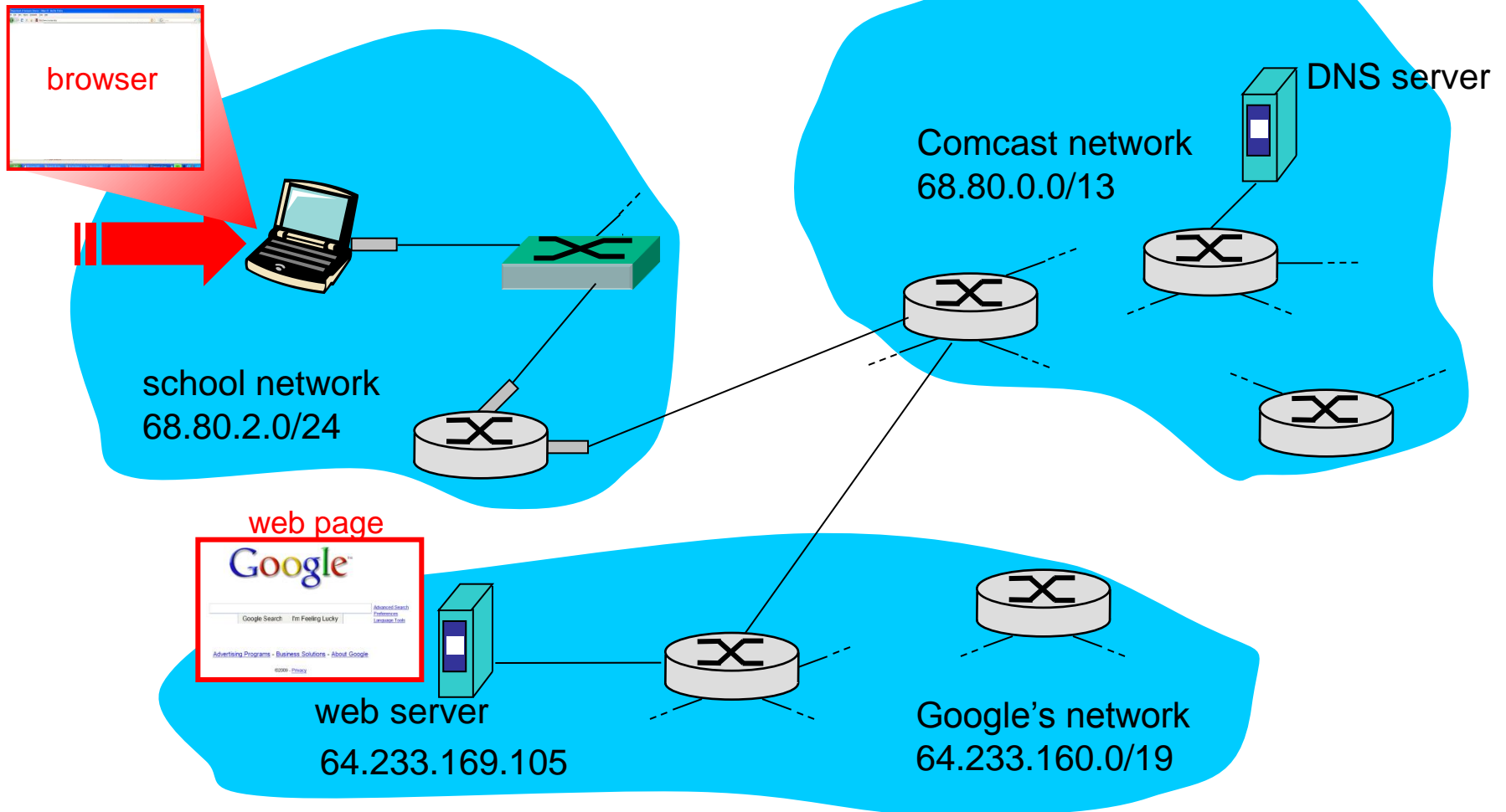
Notes derived from "*Computer Networking: A Top Down Approach*", by Jim Kurose and Keith Ross, Addison-Wesley.

The slides are adapted and modified based on slides from the book's companion Web site, as well as modified slides by Anirban Mahanti and Carey Williamson.

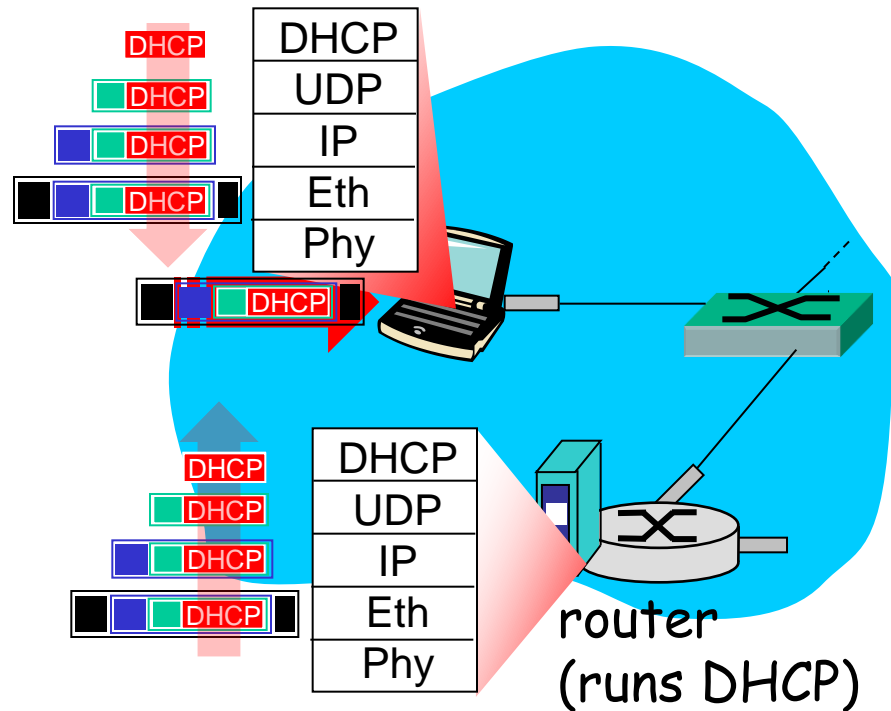
## Synthesis: a day in the life of a web request

- ❖ journey down protocol stack complete!
  - application, transport, network, link
- ❖ putting-it-all-together: synthesis!
  - *goal*: identify, review, understand protocols (at all layers) involved in seemingly simple scenario: requesting www page
  - *scenario*: student attaches laptop to campus network, requests/receives `www.google.com`

# A day in the life: scenario

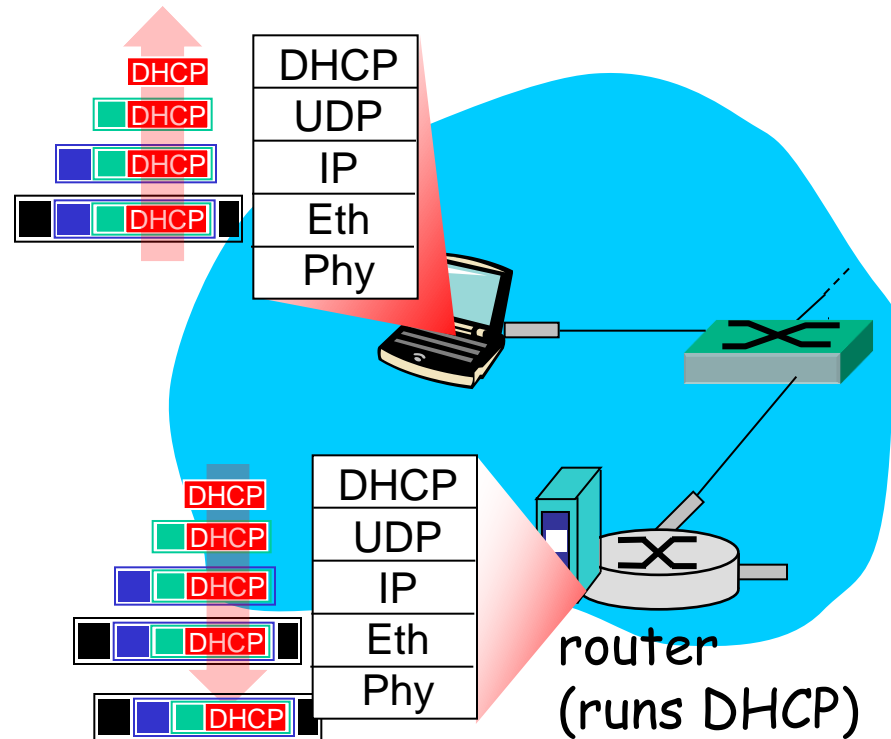


# A day in the life... connecting to the Internet



- ❖ connecting laptop needs to get its own IP address, addr of first-hop router, addr of DNS server: use **DHCP**
- ❖ DHCP request **encapsulated** in **UDP**, encapsulated in **IP**, encapsulated in **802.3** Ethernet
- ❖ Ethernet frame **broadcast** (dest: FFFFFFFFFFFFFFFF) on LAN, received at router running **DHCP** server
- ❖ Ethernet **demuxed** to IP demuxed, UDP demuxed to DHCP

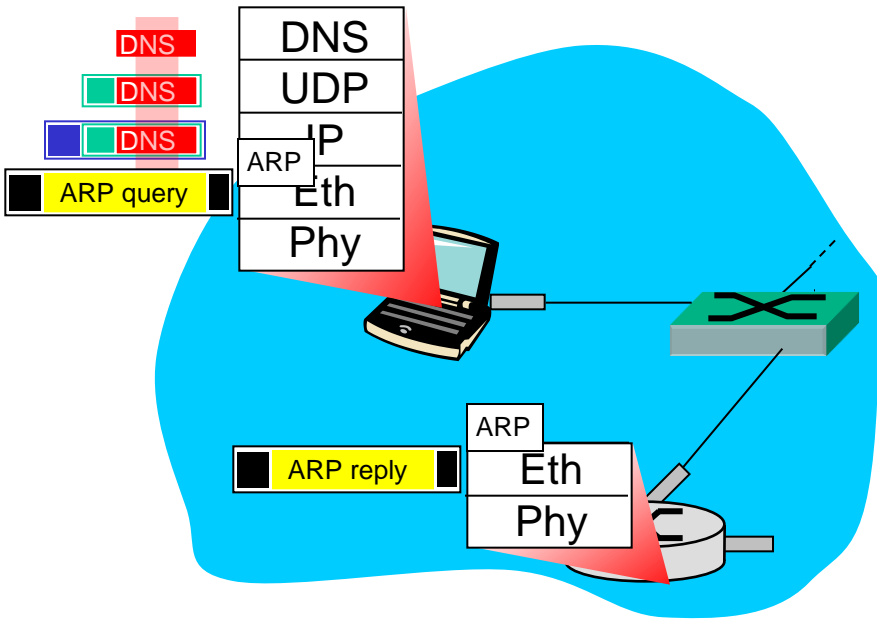
# A day in the life... connecting to the Internet



- ❖ DHCP server formulates **DHCP ACK** containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server
- ❖ encapsulation at DHCP server, frame forwarded (**switch learning**) through LAN, demultiplexing at client
- ❖ DHCP client receives DHCP ACK reply

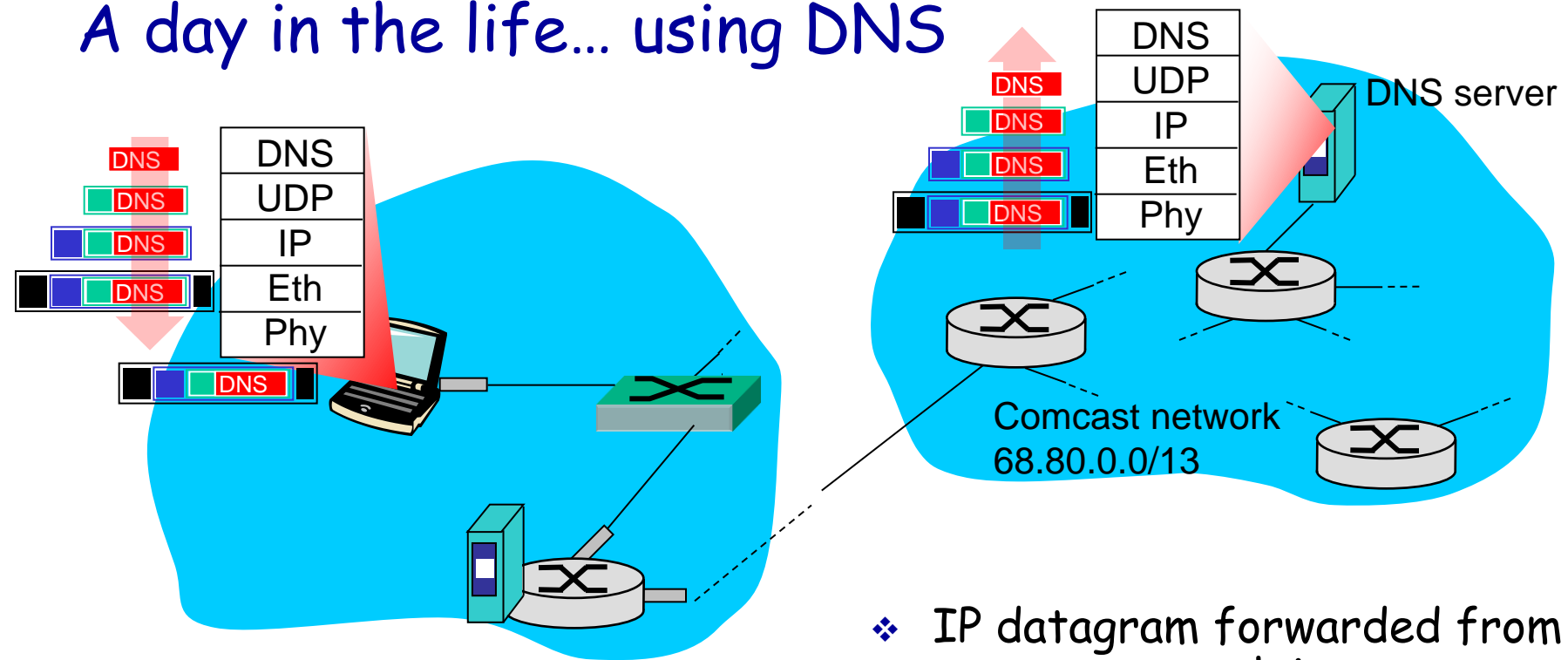
Client now has IP address, knows name & addr of DNS server, IP address of its first-hop router

# A day in the life... ARP (before DNS, before HTTP)



- ❖ before sending *HTTP* request, need IP address of `www.google.com`: **DNS**
- ❖ DNS query created, encapsulated in UDP, encapsulated in IP, encapsulated in Eth. In order to send frame to router, need MAC address of router interface: **ARP**
- ❖ **ARP query** broadcast, received by router, which replies with **ARP reply** giving MAC address of router interface
- ❖ client now knows MAC address of first hop router, so can now send frame containing DNS query

# A day in the life... using DNS

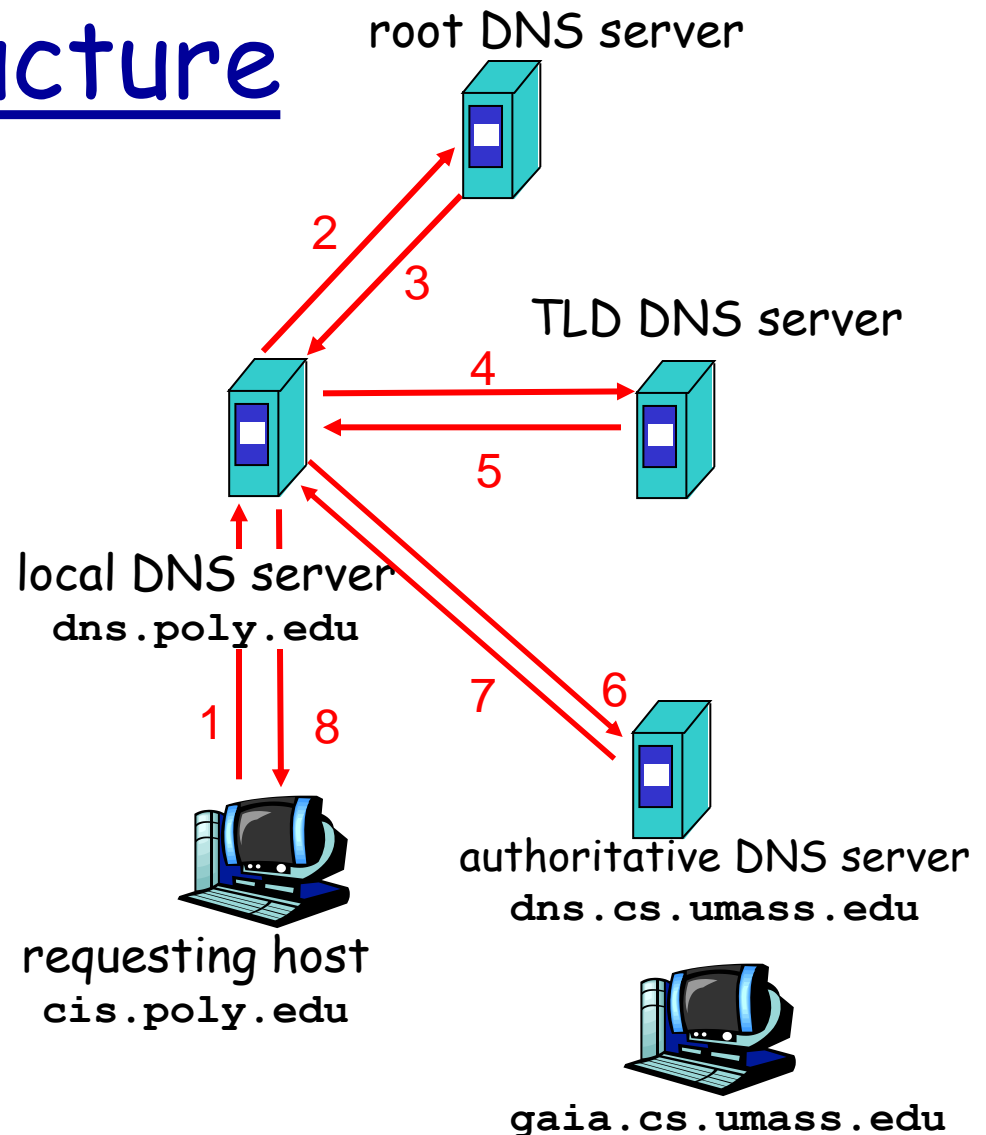


- ❖ IP datagram containing DNS query forwarded via LAN switch from client to 1<sup>st</sup> hop router

- ❖ IP datagram forwarded from campus network into comcast network, routed (tables created by **RIP**, **OSPF**, **IS-IS** and/or **BGP** routing protocols) to DNS server
- ❖ demuxed to DNS server
- ❖ DNS server replies to client with IP address of [www.google.com](http://www.google.com)

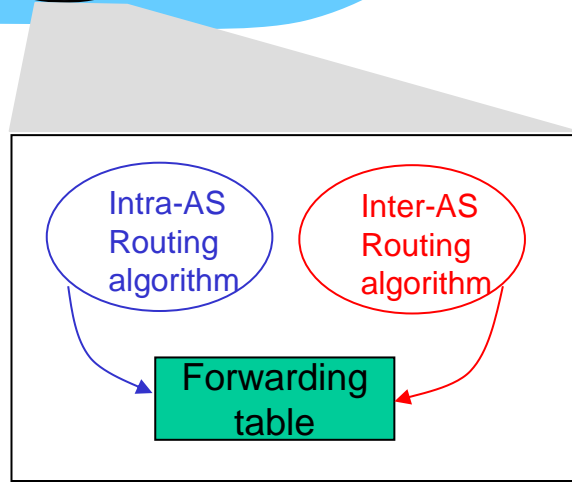
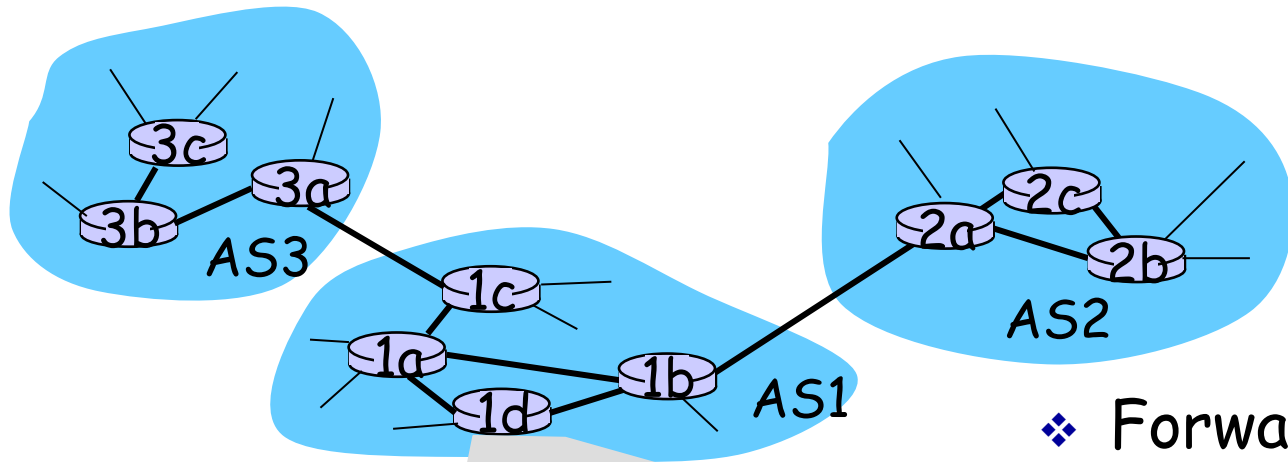
# DNS Infrastructure

- ❖ Host at cis.poly.edu wants IP address for gaia.cs.umass.edu
- ❖ Infrastructure:
  - Client resolver
  - Local DNS server
  - Authoritative DNS Server
  - Root DNS Server
  - Top-Level Domain DNS Server



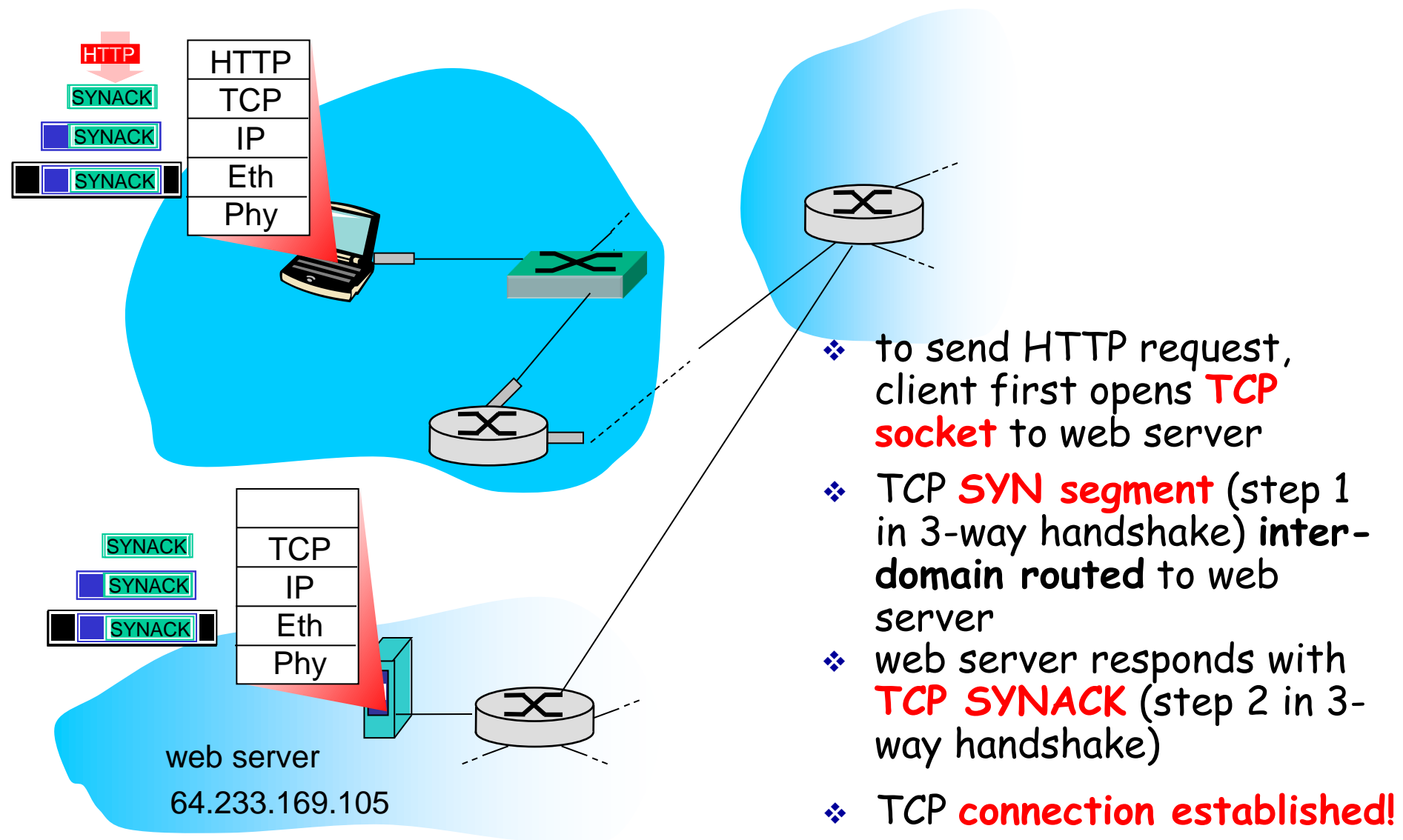


# Routing and Forwarding



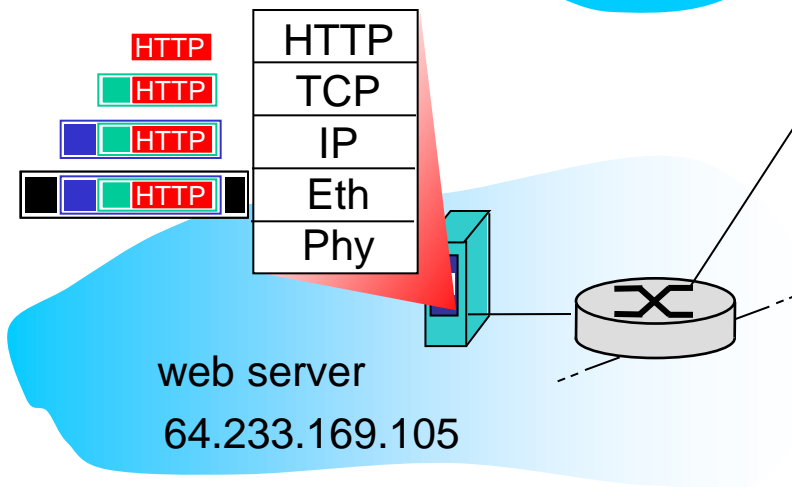
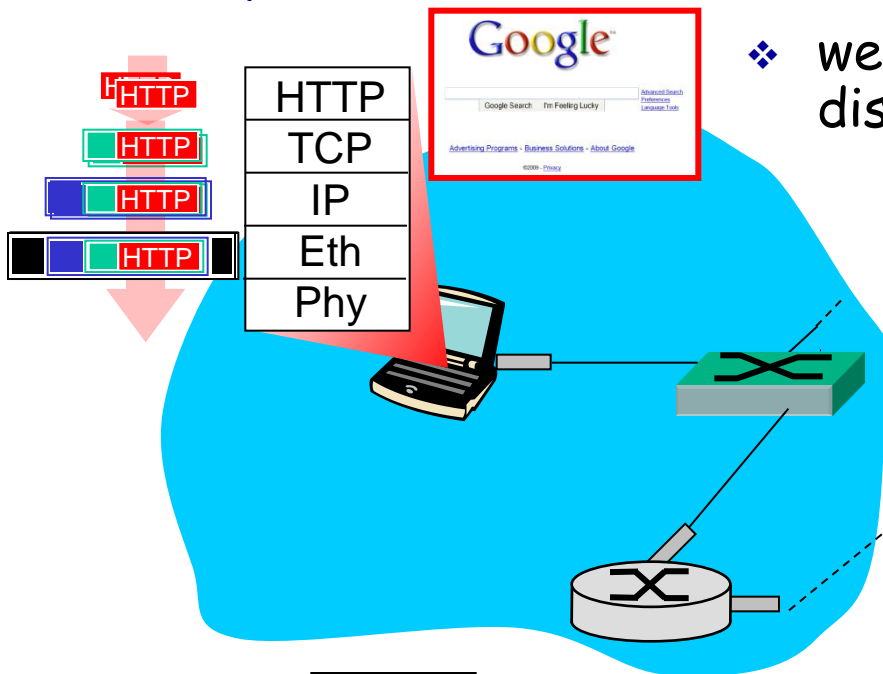
- ❖ Forwarding table is configured by both intra- and inter-AS routing algorithm
  - Intra-AS sets entries for internal dests
  - Inter-AS & Intra-As sets entries for external dests

# A day in the life... TCP connection carrying HTTP



# A day in the life... HTTP request/reply

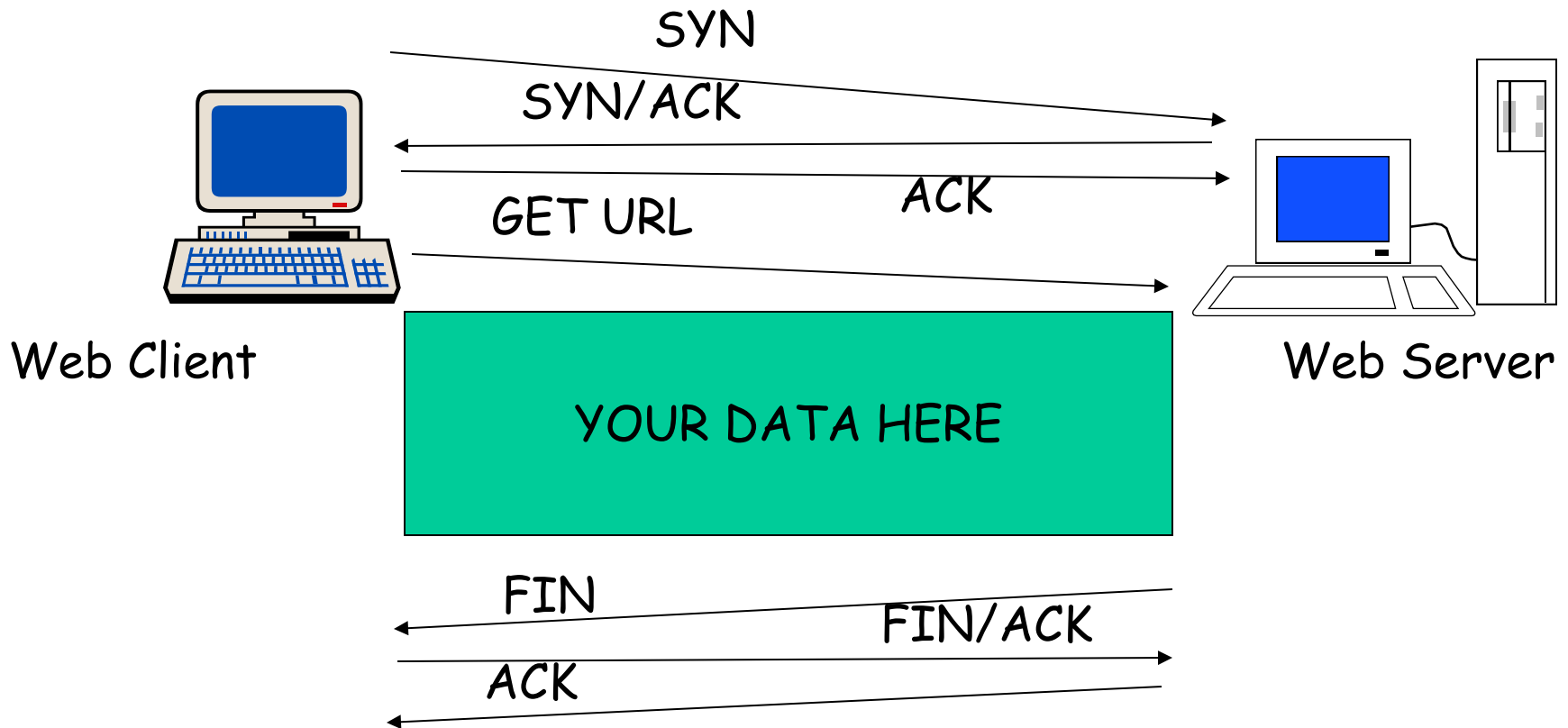
❖ web page **finally (!!!)** displayed



- ❖ **HTTP request** sent into TCP socket
- ❖ IP datagram containing HTTP request routed to `www.google.com`
- ❖ web server responds with **HTTP reply** (containing web page)
- ❖ IP datagram containing HTTP reply routed back to client

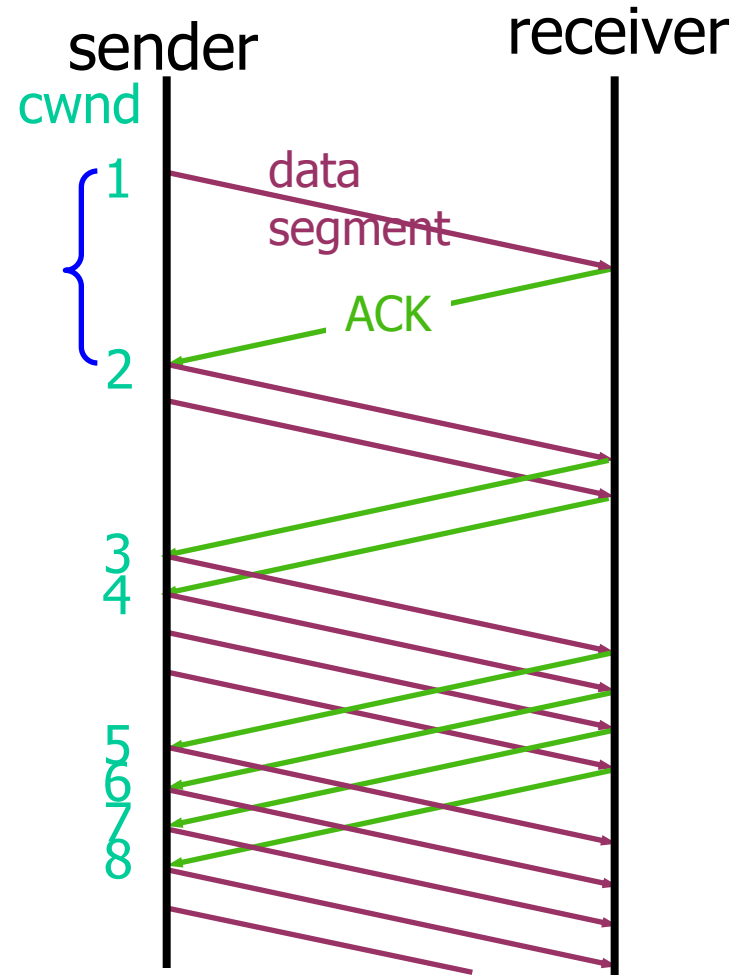
# Network View: HTTP and TCP

- ❖ TCP is a connection-oriented protocol



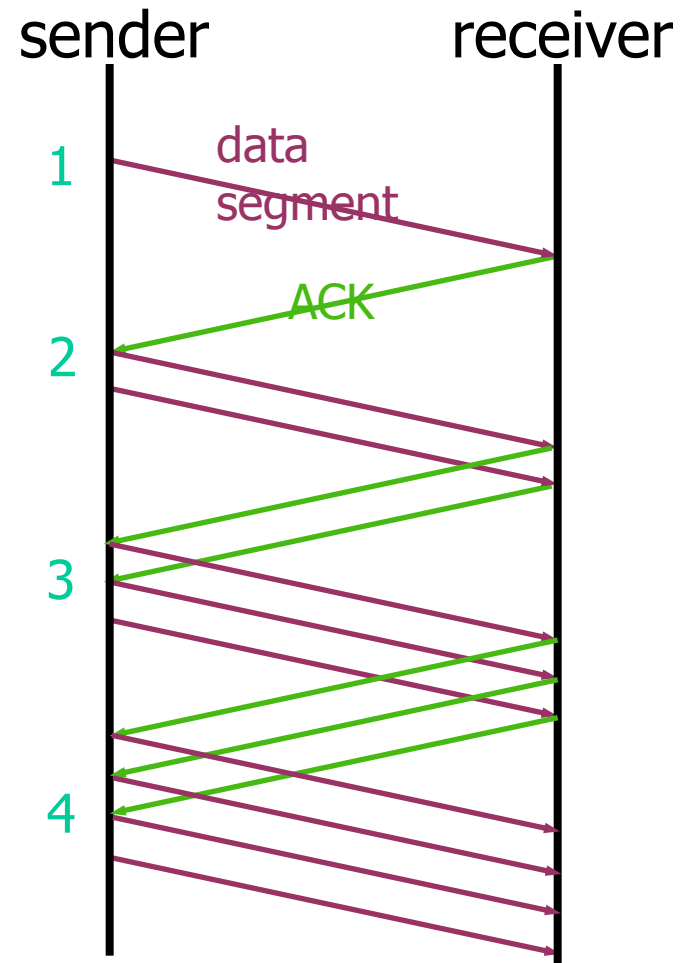
# Slow Start

- ❖ “Slow Start” is used to reach the equilibrium state
- ❖ Initially:  $W = 1$  (slow start)
- ❖ On each successful ACK:  
$$W \leftarrow W + 1$$
- ❖ Exponential growth of  $W$   
each RTT:  $W \leftarrow 2 \times W$
- ❖ Enter CA when  
$$W \geq ssthresh$$
- ❖  $ssthresh$ : window size after which TCP cautiously probes for bandwidth



# Congestion Avoidance

- ❖ Starts when
$$W \geq ssthresh$$
- ❖ On each successful ACK:
$$W \leftarrow W + 1/W$$
- ❖ Linear growth of  $W$  each RTT:
$$W \leftarrow W + 1$$





# Exam

- ❖ Read all instructions carefully
- ❖ Possible bonus marks from bonus question
- ❖ Please explain how you derived your answers. Your final answers should be clearly stated.
- ❖ Write answers legibly; no marks will be given for answers that cannot be read easily.
- ❖ Where a discourse or discussion is called for, be concise and precise.
- ❖ No assistance: closed book, closed notes, and no electronics ...



# Exam (cnt.) ...

- ❖ If necessary, state any assumptions you made in answering a question. However, remember to read the instructions for each question carefully and answer the questions as precisely as possible. Solving the *wrong question* may result in deductions! It is better to solve the *right question incorrectly*, than the *wrong question correctly*.
- ❖ Please use English. (If needed, feel free to bring a dictionary from an official publisher. Hardcopy, not electronic!! Also, your dictionary is not allowed to contain any notes; only the printed text by the publisher.)

# Some example questions

- ❖ Note: Expect to see that you understand the material ... questions will therefore be given that test your understanding.
- ❖ Also, please see the old exams (that have been given by me). At a minimum, I expect that you by the exam understand the material well enough that you can solve these and similar questions ...
  - Including providing convincing arguments and explanatory figures



## Question: TCP (6) [Web]

- ❖ Consider two machines A and B which are located 75ms apart. Assume that A is delivering a large file to B. Draw a figure that captures the send and receive events of the first 16 packets. You should consider three scenarios. (a) Assume no packet losses. (b) Assume that packets 8, 9, and 10 are lost. (c) Assume that packets 5, 6, and 7 are lost. You can assume that the TCP version is implementing fast retransmit and the timeout period is 250ms. For simplicity, you can also assume that the transmission rate is infinite.

# Question: TCP fairness (3)

## [Web]

- ❖ Please explain what it means for an UDP flow to be TCP friendly. Why is this reasonable etiquette?
- ❖ Estimate TCP throughput for two clients with the same bottleneck link.
  - May be given list of TCP connections each user has, including information about each connections RTTs ...
  - How does window size vary for the two?

# Question: Dijkstra's algorithm

## (4) [Web]

- ❖ Consider the network in the graph below. Calculate the shortest path between A and D. (Please insert your own random graph topology here.)

## Question: CDN (3) [Web]

- ❖ Both Content Distribution Networks (CDNs) and Internet Service Providers (ISPs) use caching to improve end user performance. Please explain how service models of CDNs (such as Akamai) differ from that of ISPs?

## Question: Chord (5) [Web]

- ❖ Assume a fully populated chord circle (in which there is one node per value), with a total of 128 values. Further assume that each node keeps track of its successor, predecessor, and full finger table. Please provide the routing tables of nodes 0, 1, and 5. Be sure to indicate which entries correspond to the successor, predecessor, and full finger table.