

to make the router's job easier, including using a datagram network layer rather than a virtual-circuit network layer, using a streamlined and fixed-sized header (as in IPv6), eliminating fragmentation (also done in IPv6), and providing the one and only best-effort service. Perhaps the most important trick here is *not* to keep track of individual flows, but instead base routing decisions solely on hierarchically structured destination addresses in the datagrams. It is interesting to note that the postal service has been using this approach for many years.

In this chapter, we also looked at the underlying principles of routing algorithms. We learned how routing algorithms abstract the computer network to a graph with nodes and links. With this abstraction, we can exploit the rich theory of shortest-path routing in graphs, which has been developed over the past 40 years in the operations research and algorithms communities. We saw that there are two broad approaches: a centralized (global) approach, in which each node obtains a complete map of the network and independently applies a shortest-path routing algorithm; and a decentralized approach, in which individual nodes have only a partial picture of the entire network, yet the nodes work together to deliver packets along the shortest routes. We also studied how hierarchy is used to deal with the problem of scale by partitioning large networks into independent administrative domains called autonomous systems (ASs). Each AS independently routes its datagrams through the AS, just as each country independently routes its postal mail through the country. We learned how centralized, decentralized, and hierarchical approaches are embodied in the principal routing protocols in the Internet: RIP, OSPF, and BGP. We concluded our study of routing algorithms by considering broadcast and multicast routing.

Having completed our study of the network layer, our journey now takes us one step further down the protocol stack, namely, to the link layer. Like the network layer, the link layer is also part of the network core. But we will see in the next chapter that the link layer has the much more localized task of moving packets between nodes on the same link or LAN. Although this task may appear on the surface to be trivial compared with that of the network layer's tasks, we will see that the link layer involves a number of important and fascinating issues that can keep us busy for a long time.



## Homework Problems and Questions

---

### Chapter 4 Review Questions

#### SECTIONS 4.1–4.2

- R1. Let's review some of the terminology used in this textbook. Recall that the name of a transport-layer packet is *segment* and that the name of a link-layer packet is *frame*. What is the name of a network-layer packet? Recall that both routers and link-layer switches are called *packet switches*. What is the fundamental difference between a router and link-layer switch? Recall that we use the term *routers* for both datagram networks and VC networks.

- R2. What are the two most important network-layer functions in a datagram network? What are the three most important network-layer functions in a virtual-circuit network?
- R3. What is the difference between routing and forwarding?
- R4. Do the routers in both datagram networks and virtual-circuit networks use forwarding tables? If so, describe the forwarding tables for both classes of networks.
- R5. Describe some hypothetical services that the network layer can provide to a single packet. Do the same for a flow of packets. Are any of your hypothetical services provided by the Internet's network layer? Are any provided by ATM's CBR service model? Are any provided by ATM's ABR service model?
- R6. List some applications that would benefit from ATM's CBR service model.

### SECTION 4.3

- R7. Discuss why each input port in a high-speed router stores a shadow copy of the forwarding table.
- R8. Three types of switching fabrics are discussed in Section 4.3. List and briefly describe each type. Which, if any, can send multiple packets across the fabric in parallel?
- R9. Describe how packet loss can occur at input ports. Describe how packet loss at input ports can be eliminated (without using infinite buffers).
- R10. Describe how packet loss can occur at output ports. Can this loss be prevented by increasing the switch fabric speed?
- R11. What is HOL blocking? Does it occur in input ports or output ports?

### SECTION 4.4

- R12. Do routers have IP addresses? If so, how many?
- R13. What is the 32-bit binary equivalent of the IP address 223.1.3.27?
- R14. Visit a host that uses DHCP to obtain its IP address, network mask, default router, and IP address of its local DNS server. List these values.
- R15. Suppose there are three routers between a source host and a destination host. Ignoring fragmentation, an IP datagram sent from the source host to the destination host will travel over how many interfaces? How many forwarding tables will be indexed to move the datagram from the source to the destination?
- R16. Suppose an application generates chunks of 40 bytes of data every 20 msec, and each chunk gets encapsulated in a TCP segment and then an IP datagram. What percentage of each datagram will be overhead, and what percentage will be application data?
- R17. Suppose Host A sends Host B a TCP segment encapsulated in an IP datagram. When Host B receives the datagram, how does the network layer in Host B

know it should pass the segment (that is, the payload of the datagram) to TCP rather than to UDP or to something else?

- R18. Suppose you purchase a wireless router and connect it to your cable modem. Also suppose that your ISP dynamically assigns your connected device (that is, your wireless router) one IP address. Also suppose that you have five PCs at home that use 802.11 to wirelessly connect to your wireless router. How are IP addresses assigned to the five PCs? Does the wireless router use NAT? Why or why not?
- R19. Compare and contrast the IPv4 and the IPv6 header fields. Do they have any fields in common?
- R20. It has been said that when IPv6 tunnels through IPv4 routers, IPv6 treats the IPv4 tunnels as link-layer protocols. Do you agree with this statement? Why or why not?

#### SECTION 4.5

- R21. Compare and contrast link-state and distance-vector routing algorithms.
- R22. Discuss how a hierarchical organization of the Internet has made it possible to scale to millions of users.
- R23. Is it necessary that every autonomous system use the same intra-AS routing algorithm? Why or why not?

#### SECTION 4.6

- R24. Consider Figure 4.37. Starting with the original table in *D*, suppose that *D* receives from *A* the following advertisement:

Destination Subnet	Next Router	Number of Hops to Destination
z	C	10
w	—	1
x	—	1
....	....	....

Will the table in *D* change? If so how?

- R25. Compare and contrast the advertisements used by RIP and OSPF.
- R26. Fill in the blank: RIP advertisements typically announce the number of hops to various destinations. BGP updates, on the other hand, announce the \_\_\_\_\_ to the various destinations.
- R27. Why are different inter-AS and intra-AS protocols used in the Internet?
- R28. Why are policy considerations as important for intra-AS protocols, such as OSPF and RIP, as they are for an inter-AS routing protocol like BGP?

- R29. Define and contrast the following terms: *subnet*, *prefix*, and *BGP route*.
- R30. How does BGP use the NEXT-HOP attribute? How does it use the AS-PATH attribute?
- R31. Describe how a network administrator of an upper-tier ISP can implement policy when configuring BGP.

#### SECTION 4.7

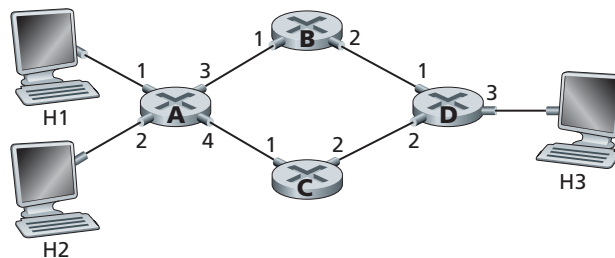
- R32. What is an important difference between implementing the broadcast abstraction via multiple unicasts, and a single network- (router-) supported broadcast?
- R33. For each of the three general approaches we studied for broadcast communication (uncontrolled flooding, controlled flooding, and spanning-tree broadcast), are the following statements true or false? You may assume that no packets are lost due to buffer overflow and all packets are delivered on a link in the order in which they were sent.
  - a. A node may receive multiple copies of the same packet.
  - b. A node may forward multiple copies of a packet over the same outgoing link.
- R34. When a host joins a multicast group, must it change its IP address to that of the multicast group it is joining?
- R35. What are the roles played by the IGMP protocol and a wide-area multicast routing protocol?
- R36. What is the difference between a group-shared tree and a source-based tree in the context of multicast routing?



## Problems

- P1. In this question, we consider some of the pros and cons of virtual-circuit and datagram networks.
  - a. Suppose that routers were subjected to conditions that might cause them to fail fairly often. Would this argue in favor of a VC or datagram architecture? Why?
  - b. Suppose that a source node and a destination require that a fixed amount of capacity always be available at all routers on the path between the source and destination node, for the exclusive use of traffic flowing between this source and destination node. Would this argue in favor of a VC or datagram architecture? Why?
  - c. Suppose that the links and routers in the network never fail and that routing paths used between all source/destination pairs remains constant. In this scenario, does a VC or datagram architecture have more control traffic overhead? Why?

- P2. Consider a virtual-circuit network. Suppose the VC number is an 8-bit field.
- What is the maximum number of virtual circuits that can be carried over a link?
  - Suppose a central node determines paths and VC numbers at connection setup. Suppose the same VC number is used on each link along the VC's path. Describe how the central node might determine the VC number at connection setup. Is it possible that there are fewer VCs in progress than the maximum as determined in part (a) yet there is no common free VC number?
  - Suppose that different VC numbers are permitted in each link along a VC's path. During connection setup, after an end-to-end path is determined, describe how the links can choose their VC numbers and configure their forwarding tables in a decentralized manner, without reliance on a central node.
- P3. A bare-bones forwarding table in a VC network has four columns. What is the meaning of the values in each of these columns? A bare-bones forwarding table in a datagram network has two columns. What is the meaning of the values in each of these columns?
- P4. Consider the network below.
- Suppose that this network is a datagram network. Show the forwarding table in router A, such that all traffic destined to host H3 is forwarded through interface 3.
  - Suppose that this network is a datagram network. Can you write down a forwarding table in router A, such that all traffic from H1 destined to host H3 is forwarded through interface 3, while all traffic from H2 destined to host H3 is forwarded through interface 4? (Hint: this is a trick question.)
  - Now suppose that this network is a virtual circuit network and that there is one ongoing call between H1 and H3, and another ongoing call between H2 and H3. Write down a forwarding table in router A, such that all traffic from H1 destined to host H3 is forwarded through interface 3, while all traffic from H2 destined to host H3 is forwarded through interface 4.
  - Assuming the same scenario as (c), write down the forwarding tables in nodes B, C, and D.



- P5. Consider a VC network with a 2-bit field for the VC number. Suppose that the network wants to set up a virtual circuit over four links: link A, link B,

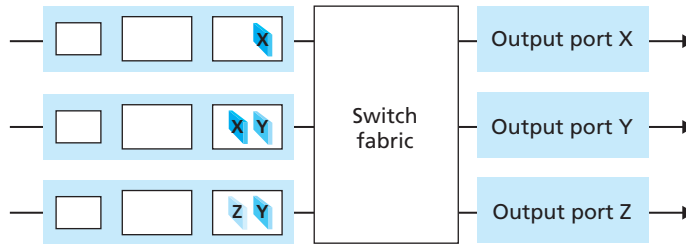
link C, and link D. Suppose that each of these links is currently carrying two other virtual circuits, and the VC numbers of these other VCs are as follows:

Link A	Link B	Link C	Link D
00	01	10	11
01	10	11	00

In answering the following questions, keep in mind that each of the existing VCs may only be traversing one of the four links.

- a. If each VC is required to use the same VC number on all links along its path, what VC number could be assigned to the new VC?
  - b. If each VC is permitted to have different VC numbers in the different links along its path (so that forwarding tables must perform VC number translation), how many different combinations of four VC numbers (one for each of the four links) could be used?
- P6. In the text we have used the term *connection-oriented service* to describe a transport-layer service and *connection service* for a network-layer service. Why the subtle shades in terminology?
- P7. Suppose two packets arrive to two different input ports of a router at exactly the same time. Also suppose there are no other packets anywhere in the router.
- a. Suppose the two packets are to be forwarded to two *different* output ports. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a *shared bus*?
  - b. Suppose the two packets are to be forwarded to two *different* output ports. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a *crossbar*?
  - c. Suppose the two packets are to be forwarded to the *same* output port. Is it possible to forward the two packets through the switch fabric at the same time when the fabric uses a *crossbar*?
- P8. In Section 4.3, we noted that the maximum queuing delay is  $(n-1)D$  if the switching fabric is  $n$  times faster than the input line rates. Suppose that all packets are of the same length,  $n$  packets arrive at the same time to the  $n$  input ports, and all  $n$  packets want to be forwarded to *different* output ports. What is the maximum delay for a packet for the (a) memory, (b) bus, and (c) crossbar switching fabrics?
- P9. Consider the switch shown below. Suppose that all datagrams have the same fixed length, that the switch operates in a slotted, synchronous manner, and that in one time slot a datagram can be transferred from an input port to an output port. The switch fabric is a crossbar so that at most one datagram can

be transferred to a given output port in a time slot, but different output ports can receive datagrams from different input ports in a single time slot. What is the minimal number of time slots needed to transfer the packets shown from input ports to their output ports, assuming any input queue scheduling order you want (i.e., it need not have HOL blocking)? What is the largest number of slots needed, assuming the worst-case scheduling order you can devise, assuming that a non-empty input queue is never idle?



- P10. Consider a datagram network using 32-bit host addresses. Suppose a router has four links, numbered 0 through 3, and packets are to be forwarded to the link interfaces as follows:

Destination Address Range	Link Interface
11100000 00000000 00000000 00000000 through 11100000 00111111 11111111 11111111	0
11100000 01000000 00000000 00000000 through 11100000 01000000 11111111 11111111	1
11100000 01000001 00000000 00000000 through 11100001 01111111 11111111 11111111	2
otherwise	3

- Provide a forwarding table that has five entries, uses longest prefix matching, and forwards packets to the correct link interfaces.
- Describe how your forwarding table determines the appropriate link interface for datagrams with destination addresses:

```

11001000 10010001 01010001 01010101
11100001 01000000 11000011 00111100
11100001 10000000 00010001 01110111

```

- P11. Consider a datagram network using 8-bit host addresses. Suppose a router uses longest prefix matching and has the following forwarding table:

Prefix Match	Interface
00	0
010	1
011	2
10	2
11	3

For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range.

- P12. Consider a datagram network using 8-bit host addresses. Suppose a router uses longest prefix matching and has the following forwarding table:

Prefix Match	Interface
1	0
10	1
111	2
otherwise	3

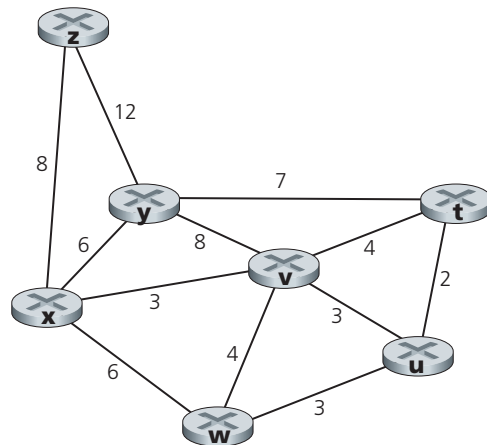
For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range.

- P13. Consider a router that interconnects three subnets: Subnet 1, Subnet 2, and Subnet 3. Suppose all of the interfaces in each of these three subnets are required to have the prefix 223.1.17/24. Also suppose that Subnet 1 is required to support at least 60 interfaces, Subnet 2 is to support at least 90 interfaces, and Subnet 3 is to support at least 12 interfaces. Provide three network addresses (of the form a.b.c.d/x) that satisfy these constraints.
- P14. In Section 4.2.2 an example forwarding table (using longest prefix matching) is given. Rewrite this forwarding table using the a.b.c.d/x notation instead of the binary string notation.
- P15. In Problem P10 you are asked to provide a forwarding table (using longest prefix matching). Rewrite this forwarding table using the a.b.c.d/x notation instead of the binary string notation.



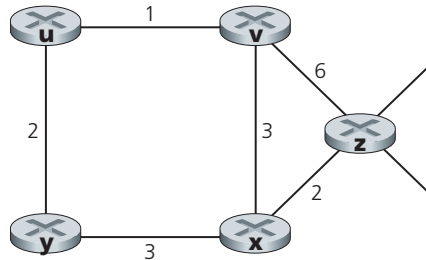
- P16. Consider a subnet with prefix 128.119.40.128/26. Give an example of one IP address (of form xxx.xxx.xxx.xxx) that can be assigned to this network. Suppose an ISP owns the block of addresses of the form 128.119.40.64/26. Suppose it wants to create four subnets from this block, with each block having the same number of IP addresses. What are the prefixes (of form a.b.c.d/x) for the four subnets?
- P17. Consider the topology shown in Figure 4.17. Denote the three subnets with hosts (starting clockwise at 12:00) as Networks A, B, and C. Denote the subnets without hosts as Networks D, E, and F.
- Assign network addresses to each of these six subnets, with the following constraints: All addresses must be allocated from 214.97.254/23; Subnet A should have enough addresses to support 250 interfaces; Subnet B should have enough addresses to support 120 interfaces; and Subnet C should have enough addresses to support 120 interfaces. Of course, subnets D, E and F should each be able to support two interfaces. For each subnet, the assignment should take the form a.b.c.d/x or a.b.c.d/x – e.f.g.h/y.
  - Using your answer to part (a), provide the forwarding tables (using longest prefix matching) for each of the three routers.
- P18. Use the whois service at the American Registry for Internet Numbers (<http://www.arin.net/whois>) to determine the IP address blocks for three universities. Can the whois services be used to determine with certainty the geographical location of a specific IP address? Use [www.maxmind.com](http://www.maxmind.com) to determine the locations of the Web servers at each of these universities.
- P19. Consider sending a 2400-byte datagram into a link that has an MTU of 700 bytes. Suppose the original datagram is stamped with the identification number 422. How many fragments are generated? What are the values in the various fields in the IP datagram(s) generated related to fragmentation?
- P20. Suppose datagrams are limited to 1,500 bytes (including header) between source Host A and destination Host B. Assuming a 20-byte IP header, how many datagrams would be required to send an MP3 consisting of 5 million bytes? Explain how you computed your answer.
- P21. Consider the network setup in Figure 4.22. Suppose that the ISP instead assigns the router the address 24.34.112.235 and that the network address of the home network is 192.168.1/24.
- Assign addresses to all interfaces in the home network.
  - Suppose each host has two ongoing TCP connections, all to port 80 at host 128.119.40.86. Provide the six corresponding entries in the NAT translation table.

- P22. Suppose you are interested in detecting the number of hosts behind a NAT. You observe that the IP layer stamps an identification number sequentially on each IP packet. The identification number of the first IP packet generated by a host is a random number, and the identification numbers of the subsequent IP packets are sequentially assigned. Assume all IP packets generated by hosts behind the NAT are sent to the outside world.
- Based on this observation, and assuming you can sniff all packets sent by the NAT to the outside, can you outline a simple technique that detects the number of unique hosts behind a NAT? Justify your answer.
  - If the identification numbers are not sequentially assigned but randomly assigned, would your technique work? Justify your answer.
- P23. In this problem we'll explore the impact of NATs on P2P applications. Suppose a peer with username Arnold discovers through querying that a peer with username Bernard has a file it wants to download. Also suppose that Bernard and Arnold are both behind a NAT. Try to devise a technique that will allow Arnold to establish a TCP connection with Bernard without application-specific NAT configuration. If you have difficulty devising such a technique, discuss why.
- P24. Looking at Figure 4.27, enumerate the paths from  $y$  to  $u$  that do not contain any loops.
- P25. Repeat Problem P24 for paths from  $x$  to  $z$ ,  $z$  to  $u$ , and  $z$  to  $w$ .
- P26. Consider the following network. With the indicated link costs, use Dijkstra's shortest-path algorithm to compute the shortest path from  $x$  to all network nodes. Show how the algorithm works by computing a table similar to Table 4.3.

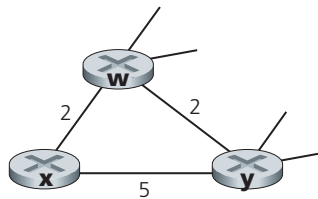


VideoNote  
Dijkstra's algorithm:  
discussion and example

- P27. Consider the network shown in Problem P26. Using Dijkstra's algorithm, and showing your work using a table similar to Table 4.3, do the following:
- Compute the shortest path from  $t$  to all network nodes.
  - Compute the shortest path from  $u$  to all network nodes.
  - Compute the shortest path from  $v$  to all network nodes.
  - Compute the shortest path from  $w$  to all network nodes.
  - Compute the shortest path from  $y$  to all network nodes.
  - Compute the shortest path from  $z$  to all network nodes.
- P28. Consider the network shown below, and assume that each node initially knows the costs to each of its neighbors. Consider the distance-vector algorithm and show the distance table entries at node  $z$ .

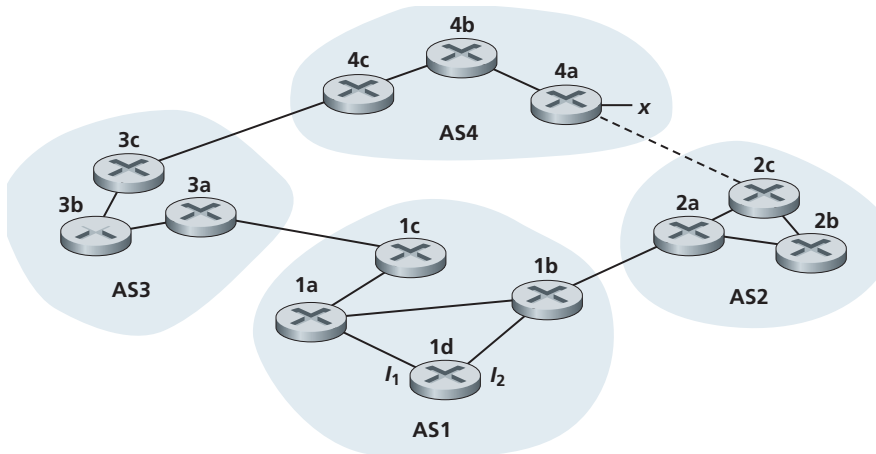


- P29. Consider a general topology (that is, not the specific network shown above) and a synchronous version of the distance-vector algorithm. Suppose that at each iteration, a node exchanges its distance vectors with its neighbors and receives their distance vectors. Assuming that the algorithm begins with each node knowing only the costs to its immediate neighbors, what is the maximum number of iterations required before the distributed algorithm converges? Justify your answer.
- P30. Consider the network fragment shown below.  $x$  has only two attached neighbors,  $w$  and  $y$ .  $w$  has a minimum-cost path to destination  $u$  (not shown) of 5, and  $y$  has a minimum-cost path to  $u$  of 6. The complete paths from  $w$  and  $y$  to  $u$  (and between  $w$  and  $y$ ) are not shown. All link costs in the network have strictly positive integer values.



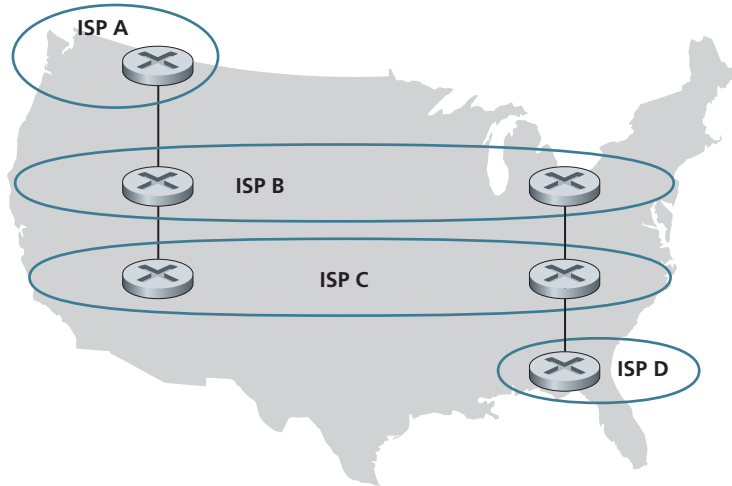
- a. Give  $x$ 's distance vector for destinations  $w$ ,  $y$ , and  $u$ .
  - b. Give a link-cost change for either  $c(x,w)$  or  $c(x,y)$  such that  $x$  will inform its neighbors of a new minimum-cost path to  $u$  as a result of executing the distance-vector algorithm.
  - c. Give a link-cost change for either  $c(x,w)$  or  $c(x,y)$  such that  $x$  will *not* inform its neighbors of a new minimum-cost path to  $u$  as a result of executing the distance-vector algorithm.
- P31. Consider the three-node topology shown in Figure 4.30. Rather than having the link costs shown in Figure 4.30, the link costs are  $c(x,y) = 3$ ,  $c(y,z) = 6$ ,  $c(z,x) = 4$ . Compute the distance tables after the initialization step and after each iteration of a synchronous version of the distance-vector algorithm (as we did in our earlier discussion of Figure 4.30).
- P32. Consider the count-to-infinity problem in the distance vector routing. Will the count-to-infinity problem occur if we decrease the cost of a link? Why? How about if we connect two nodes which do not have a link?
- P33. Argue that for the distance-vector algorithm in Figure 4.30, each value in the distance vector  $D(x)$  is non-increasing and will eventually stabilize in a finite number of steps.
- P34. Consider Figure 4.31. Suppose there is another router  $w$ , connected to router  $y$  and  $z$ . The costs of all links are given as follows:  $c(x,y) = 4$ ,  $c(x,z) = 50$ ,  $c(y,w) = 1$ ,  $c(z,w) = 1$ ,  $c(y,z) = 3$ . Suppose that poisoned reverse is used in the distance-vector routing algorithm.
- a. When the distance vector routing is stabilized, router  $w$ ,  $y$ , and  $z$  inform their distances to  $x$  to each other. What distance values do they tell each other?
  - b. Now suppose that the link cost between  $x$  and  $y$  increases to 60. Will there be a count-to-infinity problem even if poisoned reverse is used? Why or why not? If there is a count-to-infinity problem, then how many iterations are needed for the distance-vector routing to reach a stable state again? Justify your answer.
  - c. How do you modify  $c(y,z)$  such that there is no count-to-infinity problem at all if  $c(y,x)$  changes from 4 to 60?
- P35. Describe how loops in paths can be detected in BGP.
- P36. Will a BGP router always choose the loop-free route with the shortest AS-path length? Justify your answer.
- P37. Consider the network shown below. Suppose AS3 and AS2 are running OSPF for their intra-AS routing protocol. Suppose AS1 and AS4 are running RIP for their intra-AS routing protocol. Suppose eBGP and iBGP are used for the inter-AS routing protocol. Initially suppose there is *no* physical link between AS2 and AS4.

- Router 3c learns about prefix  $x$  from which routing protocol: OSPF, RIP, eBGP, or iBGP?
- Router 3a learns about  $x$  from which routing protocol?
- Router 1c learns about  $x$  from which routing protocol?
- Router 1d learns about  $x$  from which routing protocol?

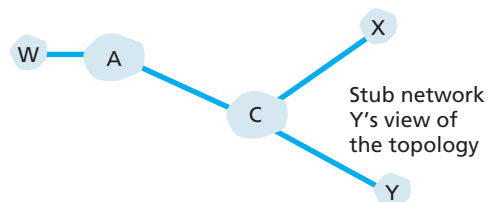


- P38. Referring to the previous problem, once router 1d learns about  $x$  it will put an entry  $(x, I)$  in its forwarding table.
- Will  $I$  be equal to  $I_1$  or  $I_2$  for this entry? Explain why in one sentence.
  - Now suppose that there is a physical link between AS2 and AS4, shown by the dotted line. Suppose router 1d learns that  $x$  is accessible via AS2 as well as via AS3. Will  $I$  be set to  $I_1$  or  $I_2$ ? Explain why in one sentence.
  - Now suppose there is another AS, called AS5, which lies on the path between AS2 and AS4 (not shown in diagram). Suppose router 1d learns that  $x$  is accessible via AS2 AS5 AS4 as well as via AS3 AS4. Will  $I$  be set to  $I_1$  or  $I_2$ ? Explain why in one sentence.
- P39. Consider the following network. ISP B provides national backbone service to regional ISP A. ISP C provides national backbone service to regional ISP D. Each ISP consists of one AS. B and C peer with each other in two places using BGP. Consider traffic going from A to D. B would prefer to hand that traffic over to C on the West Coast (so that C would have to absorb the cost of carrying the traffic cross-country), while C would prefer to get the traffic via its East Coast peering point with B (so that B would have carried the traffic across the country). What BGP mechanism might C use, so that B would hand over A-to-D traffic at its East Coast

peering point? To answer this question, you will need to dig into the BGP specification.



- P40. In Figure 4.42, consider the path information that reaches stub networks W, X, and Y. Based on the information available at W and X, what are their respective views of the network topology? Justify your answer. The topology view at Y is shown below.



- P41. Consider Figure 4.42. B would never forward traffic destined to Y via X based on BGP routing. But there are some very popular applications for which data packets go to X first and then flow to Y. Identify one such application, and describe how data packets follow a path not given by BGP routing.
- P42. In Figure 4.42, suppose that there is another stub network V that is a customer of ISP A. Suppose that B and C have a peering relationship, and A is a customer of both B and C. Suppose that A would like to have the traffic destined to W to come from B only, and the traffic destined to V from either B or C. How should A advertise its routes to B and C? What AS routes does C receive?
- P43. Suppose ASs X and Z are not directly connected but instead are connected by AS Y. Further suppose that X has a peering agreement with Y, and that Y has

a peering agreement with Z. Finally, suppose that Z wants to transit all of Y's traffic but does not want to transit X's traffic. Does BGP allow Z to implement this policy?

- P44. Consider the seven-node network (with nodes labeled  $t$  to  $z$ ) in Problem P26. Show the minimal-cost tree rooted at  $z$  that includes (as end hosts) nodes  $u$ ,  $v$ ,  $w$ , and  $y$ . Informally argue why your tree is a minimal-cost tree.
- P45. Consider the two basic approaches identified for achieving broadcast, unicast emulation and network-layer (i.e., router-assisted) broadcast, and suppose spanning-tree broadcast is used to achieve network-layer broadcast. Consider a single sender and 32 receivers. Suppose the sender is connected to the receivers by a binary tree of routers. What is the cost of sending a broadcast packet, in the cases of unicast emulation and network-layer broadcast, for this topology? Here, each time a packet (or copy of a packet) is sent over a single link, it incurs a unit of cost. What topology for interconnecting the sender, receivers, and routers will bring the cost of unicast emulation and true network-layer broadcast as far apart as possible? You can choose as many routers as you'd like.
- P46. Consider the operation of the reverse path forwarding (RPF) algorithm in Figure 4.44. Using the same topology, find a set of paths from all nodes to the source node  $A$  (and indicate these paths in a graph using thicker-shaded lines as in Figure 4.44) such that if these paths were the least-cost paths, then node  $B$  would receive a copy of  $A$ 's broadcast message from nodes  $A$ ,  $C$ , and  $D$  under RPF.
- P47. Consider the topology shown in Figure 4.44. Suppose that all links have unit cost and that node  $E$  is the broadcast source. Using arrows like those shown in Figure 4.44 indicate links over which packets will be forwarded using RPF, and links over which packets will not be forwarded, given that node  $E$  is the source.
- P48. Repeat Problem P47 using the graph from Problem P26. Assume that  $z$  is the broadcast source, and that the link costs are as shown in Problem P26.
- P49. Consider the topology shown in Figure 4.46, and suppose that each link has unit cost. Suppose node  $C$  is chosen as the center in a center-based multicast routing algorithm. Assuming that each attached router uses its least-cost path to node  $C$  to send join messages to  $C$ , draw the resulting center-based routing tree. Is the resulting tree a minimum-cost tree? Justify your answer.
- P50. Repeat Problem P49, using the graph from Problem P26. Assume that the center node is  $v$ .
- P51. In Section 4.5.1 we studied Dijkstra's link-state routing algorithm for computing the unicast paths that are individually the least-cost paths from the source to all destinations. The union of these paths might be thought of as forming a **least-unicast-cost path tree** (or a shortest unicast path tree, if all link costs are identical). By constructing a counterexample, show that the least-cost path tree is *not* always the same as a minimum spanning tree.

- P52. Consider a network in which all nodes are connected to three other nodes. In a single time step, a node can receive all transmitted broadcast packets from its neighbors, duplicate the packets, and send them to all of its neighbors (except to the node that sent a given packet). At the next time step, neighboring nodes can receive, duplicate, and forward these packets, and so on. Suppose that uncontrolled flooding is used to provide broadcast in such a network. At time step  $t$ , how many copies of the broadcast packet will be transmitted, assuming that during time step 1, a single broadcast packet is transmitted by the source node to its three neighbors.
- P53. We saw in Section 4.7 that there is no network-layer protocol that can be used to identify the hosts participating in a multicast group. Given this, how can multicast applications learn the identities of the hosts that are participating in a multicast group?
- P54. Design (give a pseudocode description of) an application-level protocol that maintains the host addresses of all hosts participating in a multicast group. Specifically identify the network service (unicast or multicast) that is used by your protocol, and indicate whether your protocol is sending messages in-band or out-of-band (with respect to the application data flow among the multicast group participants) and why.
- P55. What is the size of the multicast address space? Suppose now that two multicast groups randomly choose a multicast address. What is the probability that they choose the same address? Suppose now that 1,000 multicast groups are ongoing at the same time and choose their multicast group addresses at random. What is the probability that they interfere with each other?



## Socket Programming Assignment

At the end of Chapter 2, there are four socket programming assignments. Below, you will find a fifth assignment which employs ICMP, a protocol discussed in this chapter.

### Assignment 5: ICMP Ping

Ping is a popular networking application used to test from a remote location whether a particular host is up and reachable. It is also often used to measure latency between the client host and the target host. It works by sending ICMP “echo request” packets (i.e., ping packets) to the target host and listening for ICMP “echo response” replies (i.e., pong packets). Ping measures the RRT, records packet loss, and calculates a statistical summary of multiple ping-pong exchanges (the minimum, mean, max, and standard deviation of the round-trip times).