CS 168 — Introduction to the Internet

Spring 2025 — Midterm

PRINT Your Name: ______________________________

PRINT Your Student ID: ______________________

You have 110 minutes. There are 7 questions of varying credit. (100 points total)

Question:	1	2	3	4	5	6	7	Total
Points:	12	14	12	10	10	16	26	100

For questions with **circular bubbles**, you may select only one choice.

- ○ Unselected option (Completely unfilled)
- ◎ Don't do this (it will be graded as incorrect)
- ● Only one selected option (completely filled)

For questions with **square checkboxes**, you may select one or more choices.

- ■ You can select
- ■ multiple squares
- ☑ Don't do this (it will be graded as incorrect)

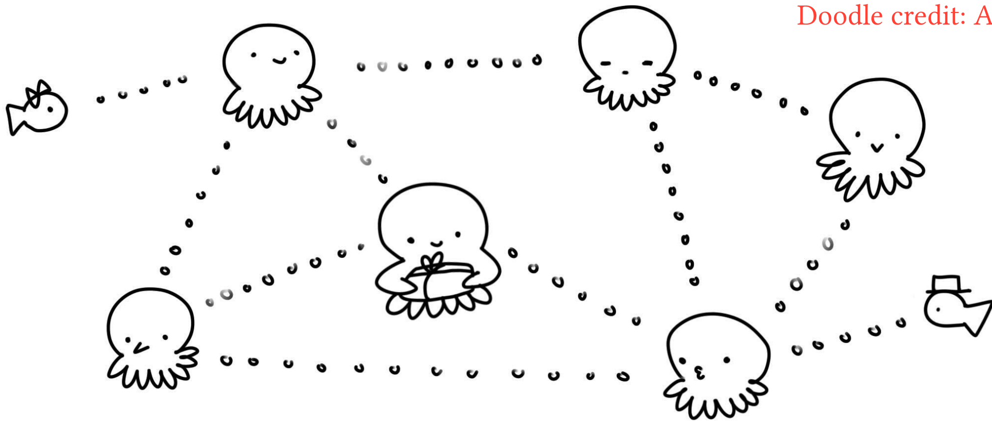
Anything you write outside the answer boxes or you ~~cross out~~ will not be graded. If you write multiple answers, your answer is ambiguous, or the bubble/checkbox is not entirely filled in, we will grade the worst interpretation.

Honor Code: Read the honor code below and sign your name.

> I understand that I may not collaborate with anyone else on this exam, or cheat in any way. I am aware of the Berkeley Campus Code of Student Conduct and acknowledge that academic misconduct will be reported to the Center for Student Conduct and may further result in, at minimum, negative points on the exam.

SIGN your name: ______________________

Doodle credit: Andrea Lou



Q1 *Potpourri* **(12 points)**

Q1.1 (2 points) When reading a UDP-over-IP packet, which of these describes the byte immediately after the end of the Layer 3 header? Select all that apply.

- ☐ Layer 4 header
- ☐ Layer 7 header
- ☐ Layer 3 payload
- ☐ Layer 4 payload
- ☐ Layer 7 payload
- ◯ None of the above

Q1.2 (1 point) If only end hosts implement reliability, it is possible to guarantee reliability.

◯ True ◯ False

Q1.3 (1 point) If only routers implement reliability, it is possible to guarantee reliability.

◯ True ◯ False

Q1.4 (1 point) If both routers and end hosts implement reliability, it is possible to guarantee reliability.

◯ True ◯ False

Q1.5 (1 point) In a circuit-switched network, routers can always use destination-based forwarding to forward packets.

◯ True ◯ False

Q1.6 (1 point) Circuit switching guarantees that packets are reliably delivered.

◯ True ◯ False

Q1.7 (1 point) If a router's packet queue is not empty, the router always processes packets in its queue before processing packets arriving from the incoming links.

◯ True ◯ False

Q1.8 (2 points) In Project 2 (routing), you implemented incremental updates in Stage 10 by supporting the `force=False` option in the `send_routes` function.

If we instead always called `send_routes` with `force=True`, would the code result in valid routing state?

- ◯ Yes, but with more advertisements sent.
- ◯ No, too many advertisements get sent.
- ◯ Yes, but with fewer advertisements sent.
- ◯ No, not enough advertisements get sent.

Q1.9 (1 point) In BGP, the size of a router's forwarding table scales with the number of hosts in the router's own AS, plus the number of other ASes.

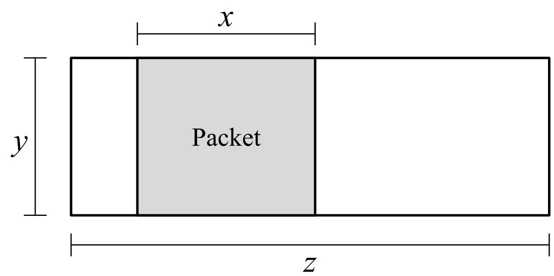
◯ True ◯ False

Q1.10 (1 point) The IPv4 checksum is used to detect payload corruption.

◯ True ◯ False

Q2 *Pipes* **(14 points)**

Consider the pipe diagram below, with a single packet in the pipe:



Q2.1 (1 point) What is the size of this packet?

- ◯ xy
- ◯ $x + y$
- ◯ $xy + z$
- ◯ xz
- ◯ $x + z$
- ◯ $x + yz$
- ◯ yz
- ◯ $y + z$
- ◯ Not enough information

Q2.2 (1 point) How long does it take to send a packet of the same size as the packet shown?

(Count from the time the first byte is sent, to the time the last byte is received.)

- ◯ xy
- ◯ $x + y$
- ◯ $xy + z$
- ◯ xz
- ◯ $x + z$
- ◯ $x + yz$
- ◯ yz
- ◯ $y + z$
- ◯ Not enough information

Q2.3 (2 points) How long does it take to send 7 packets, all of the same size as the packet shown?

(Count from the time the first byte of the first packet is sent, to the time the last byte of the last packet is received.)

- ◯ $7xy$
- ◯ $7x + y$
- ◯ $x + 7y$
- ◯ $7xz$
- ◯ $7x + z$
- ◯ $y + 7z$
- ◯ $7yz$
- ◯ $7y + z$
- ◯ Not enough information

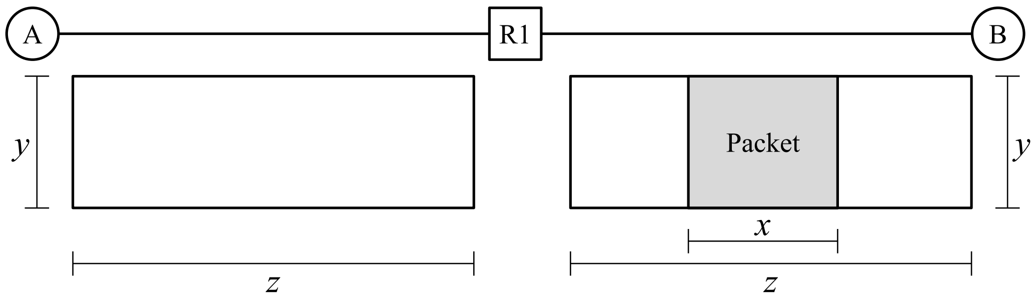
Q2.4 (2 points) What is the maximum number of packets that could be in the process of being sent along this link, at any given moment?

Don't worry about fractions and rounding (e.g. assume x, y, and z are defined such that all answer choices are integers).

- ◯ $\frac{y}{x}$
- ◯ $\frac{xy}{z}$
- ◯ $\frac{x+y}{z}$
- ◯ $\frac{z}{y}$
- ◯ $\frac{xz}{y}$
- ◯ $\frac{x+z}{y}$
- ◯ $\frac{z}{x}$
- ◯ $\frac{yz}{x}$
- ◯ Not enough information

Now, consider the topology below. R1 has an infinite-size queue, and R1 continually processes packets in its queue in FIFO order. R1 can only start transmitting a packet once it has received the entire packet.

All packets in the question are the **same size** as the packet shown in the diagram below.



A wants to send a packet to B. No other packets are being sent along the links (i.e. suppose the packet in the diagram is not there).

In the next two subparts, select how long it takes to send the packet from A to B. Count from the time the first byte of the packet is sent at A, to the time the last byte of the packet is received at B.

Q2.5 (2 points) In this subpart, suppose R1 has **1 other packet** in its queue when A sends the packet.

How long does it take to send a single packet from A to B?

- ◯ $x + z$
- ◯ $y + 2z$
- ◯ $2x + z$
- ◯ $x + 2z$
- ◯ $2y + 2z$
- ◯ $2x + 2z$
- ◯ $2x + y + 2z$
- ◯ $2x + 2y + 2z$
- ◯ Not enough information

Q2.6 (3 points) In this subpart, suppose R1 has **20 other packets** in its queue when A sends the packet.

How long does it take to send a single packet from A to B? Assume that the queue at R1 is not empty when this packet arrives at R1.

- ◯ $20x + z$
- ◯ $21x + z$
- ◯ $21x + 21z$
- ◯ $20x + 2z$
- ◯ $21x + 2z$
- ◯ $20x + 20z$
- ◯ $20x + 20y + 20z$
- ◯ $21x + 21y + 21z$
- ◯ Not enough information

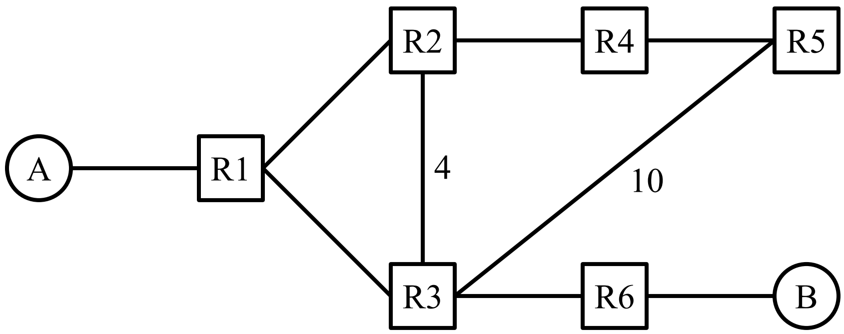
Q2.7 (3 points) What is the maximum number of packets that can be queued at R1, such that when the last byte of the A-to-B packet reaches R1, there is no queue at R1?

Don't worry about fractions and rounding (e.g. assume x, y, and z are defined such that all answer choices are integers).

- ◯ $\frac{x+y}{x}$
- ◯ $\frac{xz}{y}$
- ◯ $\frac{x+z}{x}$
- ◯ $\frac{x+z}{y}$
- ◯ $\frac{xy}{z+x}$
- ◯ $\frac{x+y+z}{x+z}$
- ◯ xz
- ◯ $x + z$
- ◯ Not enough information

Q3 ***Distance-Vector*** **(12 points)**

Consider running the distance-vector protocol from lecture on the topology below. All unlabeled links cost 1.



At time $t = 0$, all forwarding tables are empty. At time $t = 1$, static routes are installed. At each subsequent time step, every router advertises all of its routes to all of its neighbors.

Assume that advertisements are sent, received, and processed on the same time step. For example, if R3 sends an advertisement at $t = 10$, then R1 receives the advertisement and updates its table entry at $t = 10$. R1 can then advertise this updated entry at $t = 11$.

Each subpart is independent unless otherwise stated.

Q3.1 (1 point) What is the first time step t where R5 will have a table entry for destination B?

$t =$

Q3.2 (1 point) What is the first time step t where R5 will have a table entry with the least-cost route for destination B?

$t =$

Q3.3 (2 points) For this subpart only, split horizon and poison reverse are disabled.

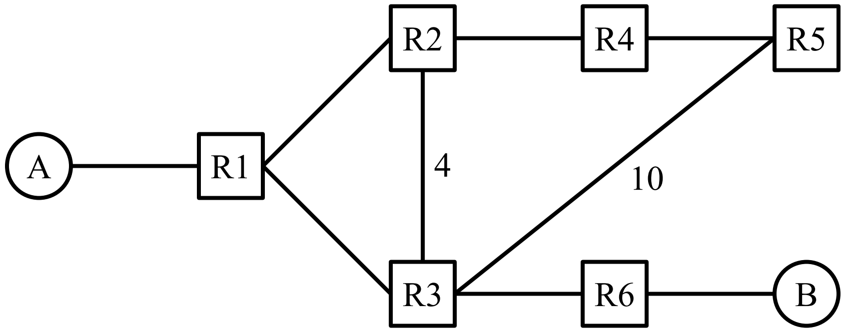
A long time later, the network has converged. At this time, the R3-to-R6 link goes down.

R3 learns about this change and updates its table entry for B.

Which routers will R3 advertise this change to?

- ◯ Only R1, because the change should only be advertised to routers using R3 as a next-hop.
- ◯ Only R1, because R1 will propagate the change to the other routers eventually.
- ◯ R1, R2, and R5, because the change should only be advertised to routers using R3 as a next-hop.
- ◯ R1, R2, and R5, because the change should be advertised to all neighbors.

The topology, reprinted for your convenience:



In the rest of the question, consider this scenario:

- Split horizon is enabled.
- No poison is ever sent.
- A long time later, the network has converged. After convergence, host A leaves the network.
- At time $t = 100$, R2 still has its entry for destination A (the same entry that it had at convergence).
- At time $t = 100$, all other routers have had their entries for destination A expired and deleted.

Q3.4 (6 points) At time $t = 103$, what table entry does each router have for destination A?

Note: $t = 103$ means you should consider three rounds of sending advertisements, receiving advertisements, and updating tables. The first round ($t = 101$) starts with R2 advertising its entry for A.

Some entries are filled in for you. For example, the bottom-right row says that R6′s forwarding table has the entry "I can reach A with cost 7, via R3."

In the Cost column, write an integer, or ∞, or "N/A" (if the table has no entry for destination A).

In the Next-Hop column, write a router (e.g. "R1"), or "N/A" (if the entry has cost infinity, or if the table has no entry for destination A).

Router	Cost to A	Next-Hop to A
R1		
R2		
R3		

Router	Cost to A	Next-Hop to A
R4	3	R2
R5		
R6	7	R3

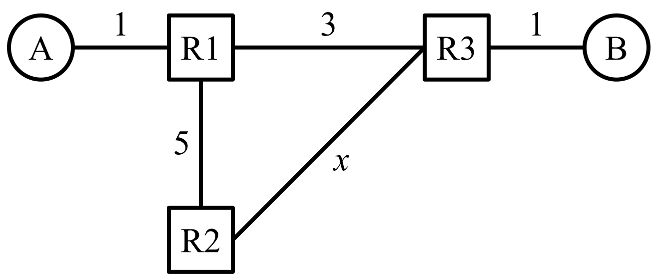
Q3.5 (2 points) Many time steps later, what happens to the routers' table entries for destination A?

(Reminder: All costs 16 or greater are infinite, and all costs 15 or less are finite.)

- ◯ Every router's entry has infinite cost.
- ◯ Only R2 has an entry with infinite cost, and all other routers have an entry with finite cost.
- ◯ R1, R2, R3 have an entry with infinite cost, and R4, R5, R6 have an entry with finite cost.
- ◯ All routers have an entry with finite cost.

Q4 *Link-State* **(10 points)**

Consider the following network topology:



All routers are running the link-state protocol from lecture.

Each subpart is independent unless otherwise stated.

Q4.1 (2 points) In total, how many "hello" advertisements get sent between the routers? Do not count periodically re-sent advertisements.

In the next two subparts, fill in the blank with a strict inequality (e.g. $x > 20$ or $x < 9$). Don't worry about cases where the result (loop or no loop) depends on tiebreaking between equal-cost routes.

Q4.2 (2 points) Assume **R1** doesn't know about the R1-to-R3 link. All other routers know the full topology.

If _____, then a packet sent from **A to B** would get stuck in a routing loop.

Q4.3 (2 points) Assume **R3** doesn't know about the R1-to-R3 link. All other routers know the full topology.

If _____, then a packet sent from **B to A** would get stuck in a routing loop.

Q4.4 (1 point) What feature of the link-state protocol will help update the forwarding table to remove these routing loops?

- ◯ Routers send poison advertisements.
- ◯ Packets have a TTL field, and the packet is dropped when the TTL reaches 0.
- ◯ Advertisements are periodically re-sent.
- ◯ None of the above. The routing loops in the above scenarios will stay forever.

Q4.5 (1 point) What is a possible cause of the scenario where R1 doesn't know about the R1-to-R3 link?

- ◯ Advertisement(s) get dropped.
- ◯ User packets get dropped.
- ◯ R1 is not running a correct shortest-path algorithm.
- ◯ The scenario where R1 doesn't know about a link will never happen in the link-state protocol.

Q4.6 (2 points) Suppose a routing loop has formed between R2 and R3.

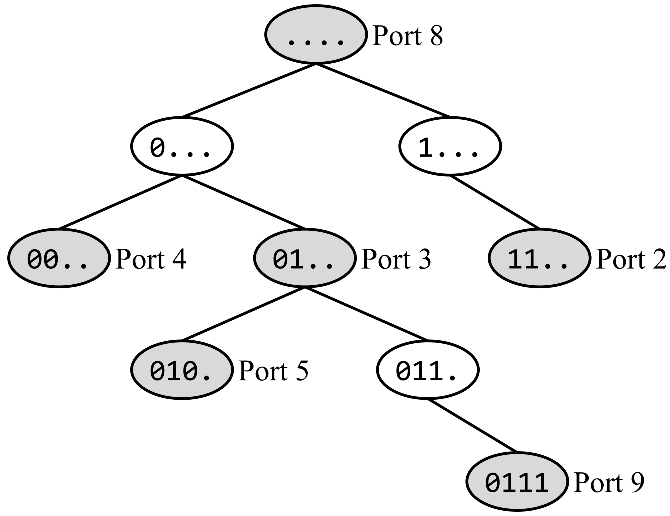
R2 forwards a user packet to R3, and then the same packet is sent back to R2.

According to the link-state protocol from lecture, what will R2 do as a result of receiving this packet?

- ◯ Forward the packet to R3, and remove an entry from its forwarding table.
- ◯ Forward the packet to R3, and add an entry to its forwarding table.
- ◯ Forward the packet to a different router (not R3), and leave the forwarding table unchanged.
- ◯ Forward the packet to R3, and leave the forwarding table unchanged.

Q5 *Longest Prefix Matching* **(10 points)**

In this question, consider the longest-prefix-matching trie below. All subparts are independent.



Q5.1 (1 point) What part of the router is responsible for building this trie?

○ Data plane

○ Control plane

○ Management plane

Q5.2 (1 point) True or false: The port numbers in this trie can be used for demultiplexing between different applications at the end host.

○ True ○ False

Q5.3 (2 points) How many of the 16 possible 4-bit IP prefixes are forwarded along the default route?

(Your answer should be a number between 0 and 16, inclusive.)

Q5.4 (2 points) In this subpart only, suppose we delete the node labeled `010.` (Port 5).

How many of the 16 possible 4-bit IP prefixes are forwarded along a different port using the modified trie (compared to the original trie)?

(Your answer should be a number between 0 and 16, inclusive.)

Q5.5 (2 points) Which of these IP prefixes represents exactly the set of IPv4 addresses that will get forwarded along Port 2?

○ `0.0.0.192/2` ○ `0.0.0.192/30` ○ `12.0.0.0/30`

○ `192.0.0.0/2` ○ `192.0.0.0/14` ○ `12.255.255.255/28`

○ `192.0.0.0/30` ○ `12.0.0.0/2` ○ `12.255.255.255/30`

Q5.6 (2 points) How many 32-bit IPv4 addresses are forwarded along Port 3?

Reminder: $2^k - 2^{k-1} = 2^{k-1}$

○ 2^2 ○ 2^{12} ○ 2^{24} ○ 2^{29} ○ 2^{31}

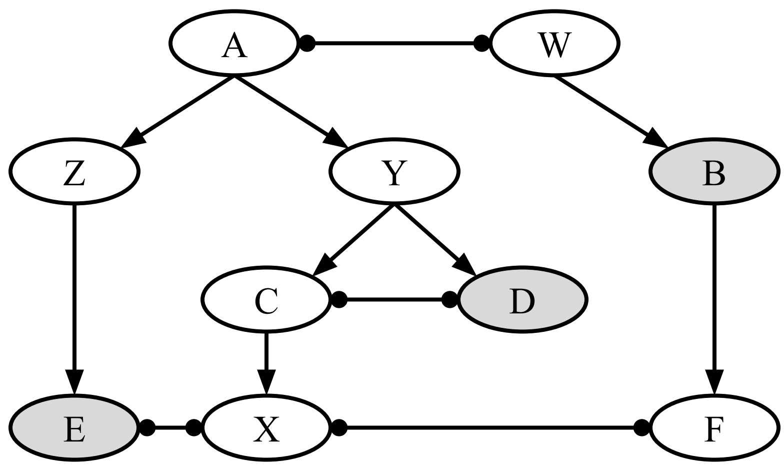
○ 2^4 ○ 2^{16} ○ 2^{28} ○ 2^{30} ○ 2^{32}

Q6 *BGP: Shady ASes with Shady Policies* **(16 points)**

In the graph below, the shaded ASes (B, D, and E) use a special policy (instead of the Gao-Rexford policies):

- **Import policy**: Instead of Gao-Rexford, prefer the advertisement from the AS that comes first alphabetically. All subsequent tiebreaking rules remain unchanged.
- **Export policy**: When you receive a path, advertise it to all neighbors.

The non-shaded ASes use the standard Gao-Rexford import and export rules.



In the next three subparts, select whether the given AS path is valid (even if it is not necessarily the path used to forward packets). In other words, select whether all ASes along the path will advertise the path.

Q6.1 (1 point) X — E — Z — A — Y

◯ Valid ◯ Invalid

Q6.2 (1 point) X — C — D — Y

◯ Valid ◯ Invalid

Q6.3 (1 point) Z — E — X — F

◯ Valid ◯ Invalid

Q6.4 (2 points) What path will packets take from X to B?

◯ X — F — B
◯ X — C — Y — A — W — B
◯ X — E — Z — A — W — B
◯ X — C — D — Y — A — W — B

Q6.5 (2 points) For this subpart only, suppose X is a shaded AS. What path will packets take from X to B?

Assume that if an AS receives two paths, it will import the one preferred path, and only export that one preferred path.

◯ X — F — B
◯ X — C — Y — A — W — B
◯ X — E — Z — A — W — B
◯ X — C — D — Y — A — W — B

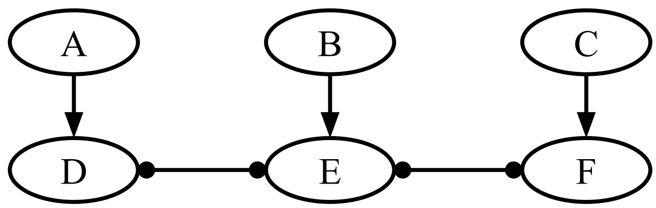
Q6.6 (2 points) Are these special policies always profitable for any AS graph (not necessarily the one above)? In other words, do these special policies always help the ASes make money and save money?

- ◯ Only the import policy is profitable.
- ◯ Both policies are profitable.
- ◯ Only the export policy is profitable.
- ◯ Neither policy is profitable.

Q6.7 (1 point) For any AS graph with one or more shaded ASes (not necessarily the one above), are paths valley-free if the shaded ASes follow the special policies?

- ◯ Always
- ◯ Sometimes
- ◯ Never

Q6.8 (3 points) In this subpart, consider this AS graph:



Select the minimum set of ASes that must be shaded in order for every pair of ASes to have a valid path between them.

(Reminder: A "valid" path is one where all ASes along the path will advertise that path.)

- ☐ A
- ☐ B
- ☐ C
- ☐ D
- ☐ E
- ☐ F

In the next three subparts, complete the following statement:

To implement the special import policy, ______ (1) routers must set the ______ (2) route attribute when sending ______ (3) messages.

Q6.9 (1 point) Blank (1):

- ◯ internal
- ◯ border
- ◯ both internal and border

Q6.10 (1 point) Blank (2):

- ◯ LOCAL PREF
- ◯ ASPATH
- ◯ MED

Q6.11 (1 point) Blank (3):

- ◯ iBGP
- ◯ eBGP
- ◯ both iBGP and eBGP

Q7 *Evil Traceroute* **(26 points)**

You are the operator of the Soda Hall router. (This means that all packets to and from Soda Hall are forwarded through your router.)

In this question, your goal is to trick any Soda Hall users who are running traceroute. If a Soda Hall end host is trying to run traceroute on the destination `4.4.4.4`, you should cause their implementation to return this fake path of 20 routers, followed by the destination:

```
["10.0.0.1", "10.0.0.2", "10.0.0.3", ... , "10.0.0.20", "4.4.4.4"]
```

Otherwise, you should process the packet as a router normally would.

Fill in `evil_traceroute` to implement this behavior. You can assume that all packets arriving at the Soda Hall router will be processed using this function. You can also assume that the traceroute users always probe using port 33434, and non-traceroute users never use port 33434.

```
# The fake path to 4.4.4.4 to show the user.
path = ["10.0.0.1", "10.0.0.2", "10.0.0.3", ... , "10.0.0.20"]

def evil_traceroute(raw_bytes):

    # Parse the packet.
    ip = IPv4(raw_bytes)
    udp = UDP(raw_bytes[20:])

    # If user is running traceroute, do evil things.
    if (Q7.1 and Q7.2 and Q7.3):

        if (Q7.4):
            reply_packet = make_icmp_packet(src_ip    = Q7.5,
                                            dst_ip    = Q7.6,
                                            error_msg = Q7.7)
        else:
            reply_packet = make_icmp_packet(src_ip    = Q7.8,
                                            dst_ip    = Q7.9,
                                            error_msg = Q7.10)

        send(reply_packet, next_hop = Q7.11)

    # Otherwise, do what a router normally would.
    else:
        processed_bytes = process_packet(raw_bytes) # Q7.12
        send(processed_bytes, next_hop = Q7.13)
```

Q7.1 (1 point) Which of these conditions helps you check if the packet is from a Soda Hall user running traceroute?

- ◯ `ip.src == "4.4.4.4"`
- ◯ `ip.src == "10.0.0.1"`
- ◯ `ip.dst == "4.4.4.4"`
- ◯ `ip.dst == "10.0.0.1"`

Q7.2 (1 point) Which of these conditions helps you check if the packet is from a Soda Hall user running traceroute?

- ◯ `udp.src_port == 33434`
- ◯ `udp.src_port == udp.dst_port`
- ◯ `udp.dst_port == 33434`
- ◯ `udp.src_port != udp.dst_port`

Q7.3 (1 point) Which of these conditions helps you check if the packet is from a Soda Hall user running traceroute?

- ◯ Check if **`ip.src`** is in the range of addresses allocated to Soda Hall.
- ◯ Check if **`ip.dst`** is in the range of addresses allocated to Soda Hall.
- ◯ Check if the router's IP address is in the range of addresses allocated to Soda Hall.
- ◯ Check if **`4.4.4.4`** is in the range of addresses allocated to Soda Hall.

Q7.4 (2 points) What condition goes in the blank?

- ◯ `if (ip.ttl > 0)`
- ◯ `if (ip.ttl > 20)`
- ◯ `if (ip.ttl > 4)`
- ◯ `if (ip.ttl > 30)`

In the next three subparts, you'll fill in the fields of **`reply_packet`** for the first case. Your answers can be a hard-coded string/integer or a Python expression. Your answers can use any of the variables in the answer choices of the previous subparts, e.g. **`ip.src`** or **`ip.ttl`** or **`udp.src_port`**.

Q7.5 (2 points) Source IP:

Q7.6 (1 point) Destination IP:

Q7.7 (1 point) ICMP Error Message:

- ◯ ICMP Port Unreachable
- ◯ ICMP Time Exceeded

In the next three subparts, you'll fill in the fields of `reply_packet` for the second case. Your answers can be a hard-coded string/integer or a Python expression.

Q7.8 (4 points) Source IP:

Q7.9 (1 point) Destination IP:

Q7.10 (1 point) ICMP Error Message:

- ◯ ICMP Port Unreachable
- ◯ ICMP Time Exceeded

In the next three subparts, `table` is the router's forwarding table (similar to the table in Project 2). You can ignore IP address aggregation in these subparts.

Q7.11 (1 point) What next-hop should we forward `reply_packet` to?

- ◯ `table[ip.src]`
- ◯ `table[udp.src_port]`
- ◯ `table[ip.dst]`
- ◯ `table[udp.dst_port]`

Q7.12 (3 points) What processing does the router need to do on a non-traceroute packet before sending it out? Select all that apply.

- ☐ Increase the TTL by 1.
- ☐ Decrease the TTL by 1.
- ☐ Recompute the IPv4 checksum.
- ☐ Recompute the UDP checksum.
- ☐ Fragment the packet if it's too big.
- ☐ Unwrap the IP header to process the UDP packet.
- ◯ None of the above

Q7.13 (1 point) What next-hop should we forward non-traceroute packets to?

- ◯ `table[ip.src]`
- ◯ `table[udp.src_port]`
- ◯ `table[ip.dst]`
- ◯ `table[udp.dst_port]`

Q7.14 (1 point) Where would the `evil_traceroute` function most likely be implemented?

- ◯ Data plane
- ◯ Control plane
- ◯ Management plane

In the next two subparts, complete the following statement:

A router implementing **evil_traceroute** would most likely handle traceroute packets as ______ (1), and handle other packets as ______ (2).

Q7.15 (1 point) Blank (1):

◯ user packets ◯ punt traffic ◯ control traffic

Q7.16 (1 point) Blank (2):

◯ user packets ◯ punt traffic ◯ control traffic

Q7.17 (1 point) Would the **evil_traceroute** function work as intended if some IPv4 packets had additional Options in their headers?

◯ Yes, **evil_traceroute** would just ignore the Options.

◯ Yes, **evil_traceroute** would correctly process the Options.

◯ No, because of Line 8: `udp = UDP(raw_bytes[20:])`

◯ No, because of Line 27: `send(processed_bytes, ...)`

In the last two subparts, consider a user in Soda Hall running traceroute. The user always sends traceroute packets from Port 29130, and always probes using Port 33434.

The user sends out a traceroute packet, and receives a response packet from the router running **evil_traceroute**.

Q7.18 (1 point) If the response packet contains a UDP Source Port header field, what is the value of this header field?

◯ 29130

◯ 33434

◯ The response does not contain a UDP Source Port field.

Q7.19 (1 point) If the response packet contains a UDP Destination Port header field, what is the value of this header field?

◯ 29130

◯ 33434

◯ The response does not contain a UDP Destination Port field.

Comment Box

Congrats for making it to the end of the exam! Leave any thoughts, comments, feedback, or doodles here.

Nothing in the comment box will affect your grade.

Ambiguities

If you feel like there was an ambiguity on the exam, you can put it in the box below.

For ambiguities, you must qualify your answer and provide an answer for both interpretations. For example, “if the question is asking about A, then my answer is X, but if the question is asking about B, then my answer is Y.” You will only receive credit if it is a genuine ambiguity and both of your answers are correct. We will only look at this box if you request a regrade.