

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Advanced Computer Networking

Exam: IN2097 / Endterm **Date:** Thursday 4th March, 2021
Examiner: Prof. Dr.-Ing. Georg Carle **Time:** 11:00 – 12:15

Working instructions

- This exam consists of **14 pages** with a total of **5 problems**.
Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 75 credits.
- Removing pages from the exam is prohibited.
- Allowed resources:
 - one **analog dictionary** English ↔ native language
 - the provided cheat sheet without any annotations
- Subproblems marked by * can be solved without results of previous subproblems.
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.

Code of conduct

- I participate without the help of others and only use the allowed resources.
- I do not share, discuss, or exchange any information related to the exam with anybody.
- I feel in good health and I am able to participate in the exam.
- I understood the examination policy, agree to the video supervision, and adhere to this process.

Left room from _____ to _____ / Early submission at _____

Problem 1 P4 (16.5 credits)

This problem investigates a Software-Defined Network (SDN) powered by P4. The source code of a P4 switch program is given in Listing 1.

```
1 header eth_t { bit<48> dstAddr;
                bit<48> srcAddr;
3                 bit<16> etherType; }

5 header ip6_t { bit<4>  version;
                bit<8>  trafficclass;
7                 bit<20> flowlabel;
                bit<16> payloadlength;
9                 bit<8>  nextheader;
                bit<8>  hoplimit;
11                bit<128> srcAddr;
                bit<128> dstAddr; }

13 header udp_t { bit<16> srcPort;
15                bit<16> dstPort;
                bit<16> length;
17                bit<16> checksum; }

19 struct metadata { /* unused */ }

21 struct headers { eth_t  eth;
                  ip6_t  ipv6;
23                  udp_t  udp; }

25 parser ParserImpl(packet_in packet, out headers hdr, inout metadata meta, inout standard_metadata_t
    standard_metadata) {

27     state parse_udp { packet.extract(hdr.udp);
                       transition accept; }

29     state parse_ip6 { packet.extract(hdr.ipv6);
                       transition select(hdr.ipv6.nextheader) { 0x11:  parse_udp;
                                                                default: accept; }}

31     state parse_eth { packet.extract(hdr.eth);
                       transition select(hdr.eth.etherType) { 0x86dd:  parse_ip6;
                                                                default: accept; }}

33     state start    { transition parse_eth; }

37 }

39 control DeparserImpl(packet_out packet, in headers hdr) {
41     apply { packet.emit(hdr.eth);
             packet.emit(hdr.ipv6);
43             packet.emit(hdr.udp); }
45 }

47 control Pipeline(inout headers hdr, inout metadata meta, inout standard_metadata_t standard_metadata) {
49     action my_drop()          { mark_to_drop(standard_metadata); }
51     action set_egress(bit<9> port) { standard_metadata.egress_spec = port; }
53     action set_default_egress()  { standard_metadata.egress_spec = 2; }

55     table filter { actions = { set_egress; my_drop; set_default_egress; }
                    key =     { hdr.udp.dstPort: exact; }
                    default_action = set_default_egress(); }

57     table forward { actions = { set_egress; my_drop; set_default_egress; }
                      key =     { standard_metadata.ingress_port: exact; }
                      default_action = set_default_egress(); }

59     apply { if (hdr.udp.isValid()) {
61             filter.apply();
63             } else if (hdr.eth.isValid()) {
65                 forward.apply();
67             }
69 }

69 V1Switch(ParserImpl(), Pipeline(), DeparserImpl()) main;
```

Listing 1: Simple P4 program

For the following problems use the network topology given in Figure 1.1. Switch S is a P4 switch running the P4 program of Listing 1.

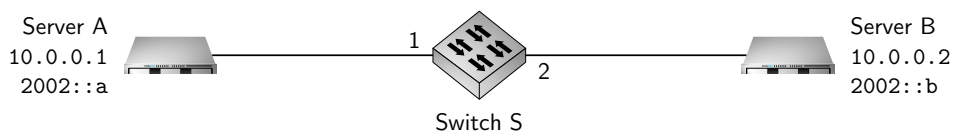


Figure 1.1: Network topology

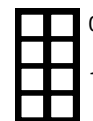
Match field(s)	Key	Action	Action data
standard_metadata.ingress_port	2	set_egress	1

Table 1.1: Rule entered into the table forward

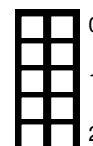
For the following subproblems you can assume that Servers A and B know the MAC address of each other, i.e., you do not need to describe or consider address resolution in your answers.

The administrator of the network in Figure 1.1 wants test the connectivity in his network using ping6. Therefore, he executes the command `ping6 2002::b` on Server A.

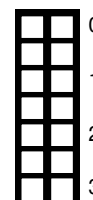
a)* What are the protocols used for the packet created by mentioned ping6 command?




b) Explain the path of the previously parsed packet through the `ParserImpl` of the P4 program in Listing 1, mentioning all passed states and the decisions taken in these states.




c) Explain the path of the packet through the `Pipeline` of the P4 program in Listing 1, mentioning all passed decisions, tables, table entries, actions, and what happens to the packet.




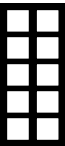
0  d) Explain the differences of the packet's path through the P4 program if the administrator executes ping6 2002::a on Server B.

The administrator wants to test the connectivity of a webserver running on Server A from Server B. Therefore, he uses the QUIC protocol (on IPv6).

0  e)* Create an entry in Table filter so the QUIC packets sent by Server B can reach Server A. **Hint:** The table below may contain more rows than the number of rules actually required to perform the described task.

Match field(s)	Key	Action	Action data

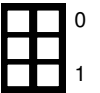
0  f) Explain what happens to the reply message of Server A to the QUIC request message of Server B on Switch S?

0  g)* The administrator of the network in Figure 1.1 wants to block ssh. Reason why he can or cannot specifically block ssh (and no other protocols) using only table entries on Switch S?

Problem 2 Quiz (17 credits)

The following questions cover multiple topics and can be solved independently of each other.

a)* Modern network interface cards (NICs) use standardized switchable transceivers, such as the SFP module. What is a benefit of such a system compared to a NIC with a fixed transceiver?



b)* Given the IP address 10.0.21.16 and subnet mask 255.255.255.192, determine the corresponding network and broadcast addresses.

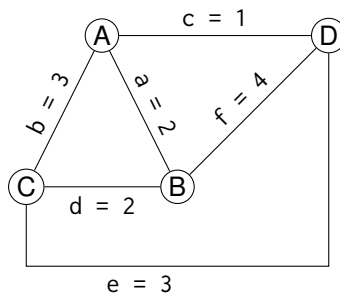
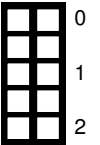
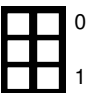
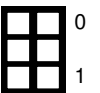


Figure 2.1: Network topology

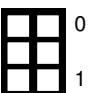
c)* Figure 2.1 shows a network topology with redundant connections and their corresponding costs. List the edges that are part of a minimum spanning tree on this topology.



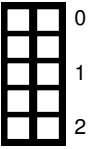
d)* List the edges that would be part of the resulting graph if the spanning tree protocol (STP) is performed on the topology in Figure 2.1 (with A as its root bridge).



e)* How does the memory consumption change for a growing number of entries in a routing table using the DIF-24-8 algorithm?



j)* Consider the following command outputs: One is executed from a host in Munich while the other is executed from a host in Frankfurt.



```
Munich ~$ dig +short example.com
52.208.128.101
52.18.15.9
52.19.40.147
```

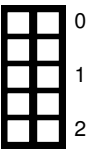
```
Frankfurt ~$ dig +short example.com
54.76.83.27
54.77.81.254
54.77.186.213
```

Listing 2: Command execution from Munich

Listing 3: Command execution from Frankfurt

Describe the DNS-based load balancing mechanisms deployed by `example.com` that can be directly derived from the commands and their outputs.

k)* Describe two effects (besides the simple traffic increase) of the COVID-19 pandemic on the Internet, which were presented during the lecture.



Problem 3 Wireshark (15.5 credits)

According to the OSI model network protocols are distributed to seven different layers each containing several protocols. In this problem a frame is analyzed referring to the involved protocols.

```

0x0000  00 90 1a 41 65 41 00 16      cf 41 9c 20 08 00 45 00

0x0010  00 50 93 5a 00 00 80 29      ff ee 46 37 d5 d3 c0 58

0x0020  63 01 60 00 00 00 00 14      06 80 20 02 46 37 d5 d3

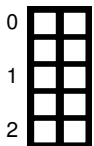
0x0030  00 00 00 00 00 00 46 37      d5 d3 20 01 00 00 0a 00

0x0040  00 00 a9 00 00 00 00 00      00 1a a5 07 00 50 22 ec
                                     TCP header start
0x0050  58 2e 3a c0 18 c5 50 10      42 48 b8 b3 00 00 00 c0

0x0060  ff ee
  
```

Figure 3.1: Hexdump of a complete Ethernet frame including FCS

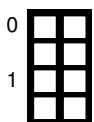
For this problem you must indicate the location of the bytes in the hexdump in Figure 3.1. You can **either** mark the corresponding bytes directly in the figure **or** list the locations of the corresponding bytes using []. Example: the bytes from position 0 to 2 can be written as [0, 2] = 0x00901a.



a)* Mark and name all parts of the protocol specific information for layer 2 for the hexdump in Figure 3.1.

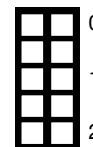
In the next six subproblems you are asked to identify which protocols were used for each layer. For each question do the following:

- **either** mark the corresponding bytes in the hexdump **or** list the bytes using the [] notation,
- **and** write the corresponding bytes in the solutionbox.

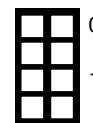


b)* Name the L3 protocol.

c) What is the length (in bytes) of the header of the protocol identified in Subproblem b).



d) Name the protocol of the payload of the protocol identified in Subproblem b).



e) Write down the destination address of the payload of the protocol identified in Subproblem d).



If you could not solve the previous subproblems, you can start with the evaluation of the TCP header at the marked position in Figure 3.1.

f)* Identify the application layer protocol.

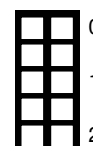


g)* Was the hexdump taken during the connection setup or teardown of the connection?



For the following questions you do **NOT** need to mark/list the corresponding bytes in the hexdump.

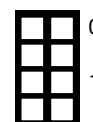
h) Note the address of Subproblem e) in the **shortest** way possible.



i) Explain a situation where a combination of protocols identified in Subproblems b) and d) may be necessary.



j) Name all the checksums contained in the hexdump in Figure 3.1.



Problem 4 AS Relations and BGP (14.5 credits)

This problem investigates the autonomous system (AS) relationships in a given network and their impact on routing and traffic. All ASes apply standard routing behavior. Furthermore, the following policies are applied:

- If routes with the same prefix exist, an AS selects the cost-efficient route.
- If routes with the same prefix exist, and the cost to route traffic is equal for all routes, the shorter route is selected.

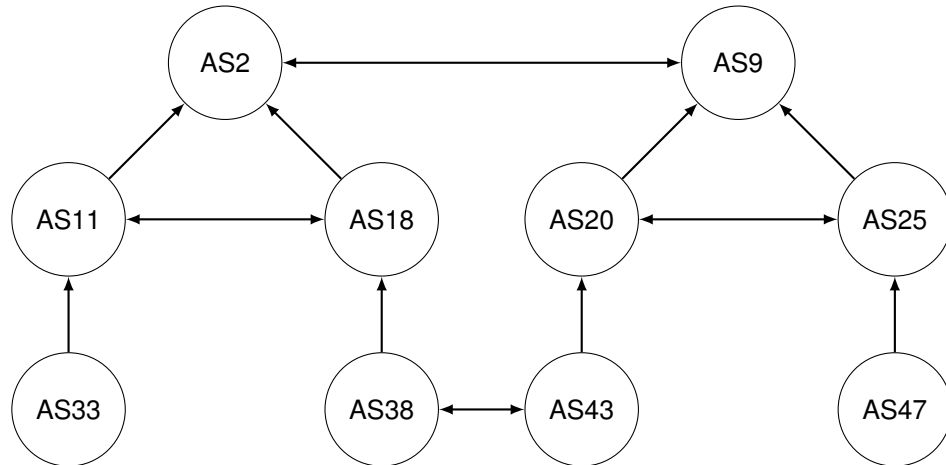
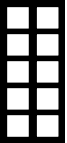




Figure 4.1: AS Network

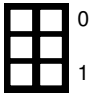
0  a)* Calculate the k-core of the given graph. What is the degree of the k-core? List all removed nodes and briefly explain your steps.

0  b)* Briefly explain the term **Tier-1 provider** based on the lecture and name all Tier-1 providers in the given network.

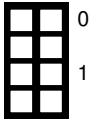
0  c)* Briefly explain the term **stub network** based on the lecture and name all stub networks in the given network.

AS47 owns the prefix 10.0.0.0/22 and announces the prefix to the network.

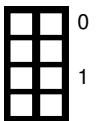
d)* How is traffic from AS20 routed towards the announced prefix? Briefly explain your answers.



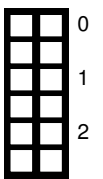
e)* How is traffic from AS38 routed towards the announced prefix? Briefly explain your answers.



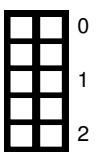
f)* How is traffic from AS33 routed towards the announced prefix? Briefly explain your answers.



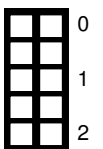
g)* AS18 wants to *eavesdrop* traffic from AS33 towards prefix 10.0.0.0/22 owned by AS47 without **hijacking** the prefix. Is it possible for AS18 to *eavesdrop* the traffic. If yes, explain how.



h)* AS18 wants to hijack the prefix from AS47 and announces 10.0.0.0/23. Which other ASes are impacted by this announcement? Is this a successful hijack of 10.0.0.0/22? Briefly explain your answer.



i) Announcements from AS18 propagate through the network and also reach AS47. AS47 realizes this is a hijack and wants to protect itself. Which BGP announcements can AS47 send to reclaim the hijacked prefix throughout the complete network. Briefly explain your answer.



Problem 5 Network Calculus (11.5 credits)

This problem investigates Network Calculus and its applications to determine delay bounds in networks. Always document your approach and simplify terms as much as possible, unless specified otherwise.

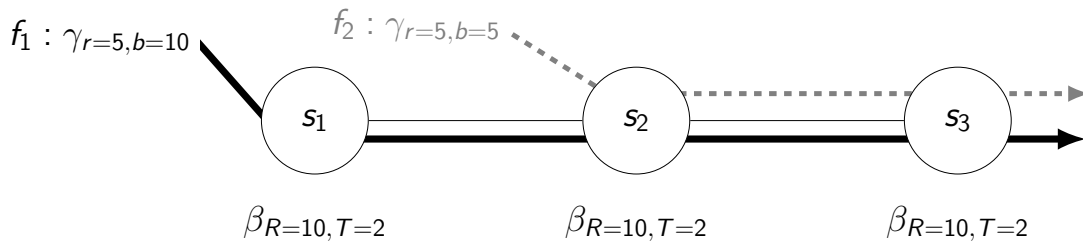


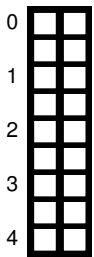
Figure 5.1: Network topology and flow description

Let the output envelope of a token-bucket constrained flow $\gamma_{r,b}$ traversing a rate-latency server $\beta_{R,T}$ be defined as $(\gamma_{r,b} \otimes \beta_{R,T})(t) = \gamma_{r,b+r \cdot T}(t)$

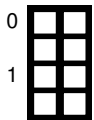
Let the left-over service curve for a token-bucket constrained flow and a rate-latency server be defined as $\beta^{l.o.} = [\beta_{R,T} - \gamma_{r,b}]^+ = \beta_{R-r, \frac{b+r \cdot T}{R-r}}$

Consider the network shown in Figure 5.1. Assume preemptive Strict Priority Queuing at each server. Flow f_1 has a low priority and Flow f_2 has a high priority. We are interested in calculating the delay bound for Flow f_1 using the Separate Flow Analysis.

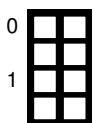
Note: You are not required to use special symbols (e.g. $\beta_{R,T}$ and \otimes). But make sure your notation is consistent and understandable (e.g., *beta_R,T* and *convolution*).



a)* Perform the first step of the Separate Flow Analysis.



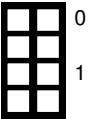
b) Perform the second step of the Separate Flow Analysis.



c) Perform the third step of the Separate Flow Analysis.

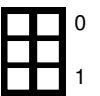
d) Assume two separate scenarios:

1. The service curve of only s_2 is changed to $\beta_{R=10, T=4}$
2. The service curve of only s_3 is changed to $\beta_{R=10, T=4}$

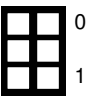


Briefly argue which scenario produces the larger delay bound for Flow f_1 under the Separate Flow Analysis.

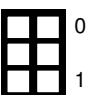
e)* Why is it important to distinguish between preemptive and non-preemptive scheduling when performing the Separate Flow Analysis?



f)* What is given by the maximum vertical distance between service curve and arrival curve?



g)* Assume a token-bucket constrained flow $\gamma_{r=10, b=50}$ traversing a single rate-latency server $\beta_{R=12, T=8}$. Assume arbitrary multiplexing. Argue whether or not a finite delay bound for the flow can be computed.



Additional space for solutions—clearly mark the (sub)problem your answers are related to and strike out invalid solutions.

