



UNIVERSITETET I BERGEN

KANDIDAT

**115**

PRØVE

**INF142 0 Datanett**

Emnekode	INF142
Vurderingsform	Skriftlig eksamen
Starttid	31.05.2021 07:00
Sluttid	31.05.2021 10:00
Sensurfrist	--
PDF opprettet	02.10.2022 11:21

**Information**

Oppgave	Oppgavetype
<b>i</b>	Informasjon eller ressurser
<b>i</b>	Informasjon eller ressurser
<b>i</b>	Informasjon eller ressurser

**True/False**

Oppgave	Oppgavetype
1	Sant/usant
2	Sant/usant
3	Sant/usant
4	Sant/usant
5	Sant/usant
6	Sant/usant
7	Sant/usant
8	Sant/usant
9	Sant/usant
10	Sant/usant
11	Sant/usant
12	Sant/usant

**Multiple choice**

Oppgave	Oppgavetype
---------	-------------

13	Flervalg
----	----------

14	Flervalg
----	----------

15	Flervalg
----	----------

### Delay in a videocall

Oppgave	Oppgavetype
---------	-------------

16	Fyll inn tall
----	---------------

### Throughput

Oppgave	Oppgavetype
---------	-------------

17	Flervalg
----	----------

### HTTP

Oppgave	Oppgavetype
---------	-------------

18	Langsvar
----	----------

### E-mail

Oppgave	Oppgavetype
---------	-------------

19	Langsvar
----	----------

### Tiny-UDP

Oppgave	Oppgavetype
---------	-------------

20	Langsvar
----	----------

### Subnet

Oppgave	Oppgavetype
---------	-------------

21

Langsvar

**Overhead percentage****Oppgave****Oppgavetype**

22

Fyll inn tall

**Dijkstra****Oppgave****Oppgavetype**

23

Langsvar

**Socket Programming****Oppgave****Oppgavetype**

24

Langsvar

**Wireshark****Oppgave****Oppgavetype**

25

Langsvar

**Workaround****Oppgave****Oppgavetype**

26

Tekstfelt

- 1 The UDP/TCP checksum is used to determine whether the destination-port exists or not.

**Select one alternative:**

☐ True

☒ False

---

Maks poeng: 1

- 2 A TCP connection provides a full-duplex service.

**Select one alternative:**

☐ False

☒ True

---

Maks poeng: 1

- 3 IPv6 has a Header checksum field.

**Select one alternative:**

☒ False

☐ True

---

Maks poeng: 1

- 4 Collisions may happen in a downstream HFC channel.

**Select one alternative:**

☒ False

☐ True

---

Maks poeng: 1

- 5 IPv6 allows for datagram fragmentation at intermediate routers.

**Select one alternative:**

☒ False

☐ True

---

Maks poeng: 1

- 6 Developers cannot implement reliable data transfer in their applications without using TCP.

**Select one alternative:**

☐ True

☒ False

---

Maks poeng: 1

- 7 DNS can be used to perform load distribution among replicated servers.

**Select one alternative:**

☐ False

☒ True

---

Maks poeng: 1

- 8 A single administrative entity can have more than one AS.

**Select one alternative:**

☐ False

☒ True

---

Maks poeng: 1

- 9 An Autonomous system can have two subnets whose routing policies are completely different.

**Select one alternative:**

☐ False

☒ True

---

Maks poeng: 1

- 10 In TCP, the round-trip time is estimated by using an exponential weighted moving average.

**Select one alternative:**

☐ False

☒ True

---

Maks poeng: 1

- 11 The following is a valid HTTP request.

```
GET * FROM \ WHERE
Host: www.google.com
User-Agent: Mozilla/5.0
Accept-Encoding: gzip, deflate
Accept: */*
Connection: keep-alive
```

**Select one alternative:**

☒ False

☐ True

---

Maks poeng: 1

12 It is always the case that  $d_{\text{nodal}} = d_{\text{trans}} + d_{\text{prop}}$ .

Select one alternative:

☒ False

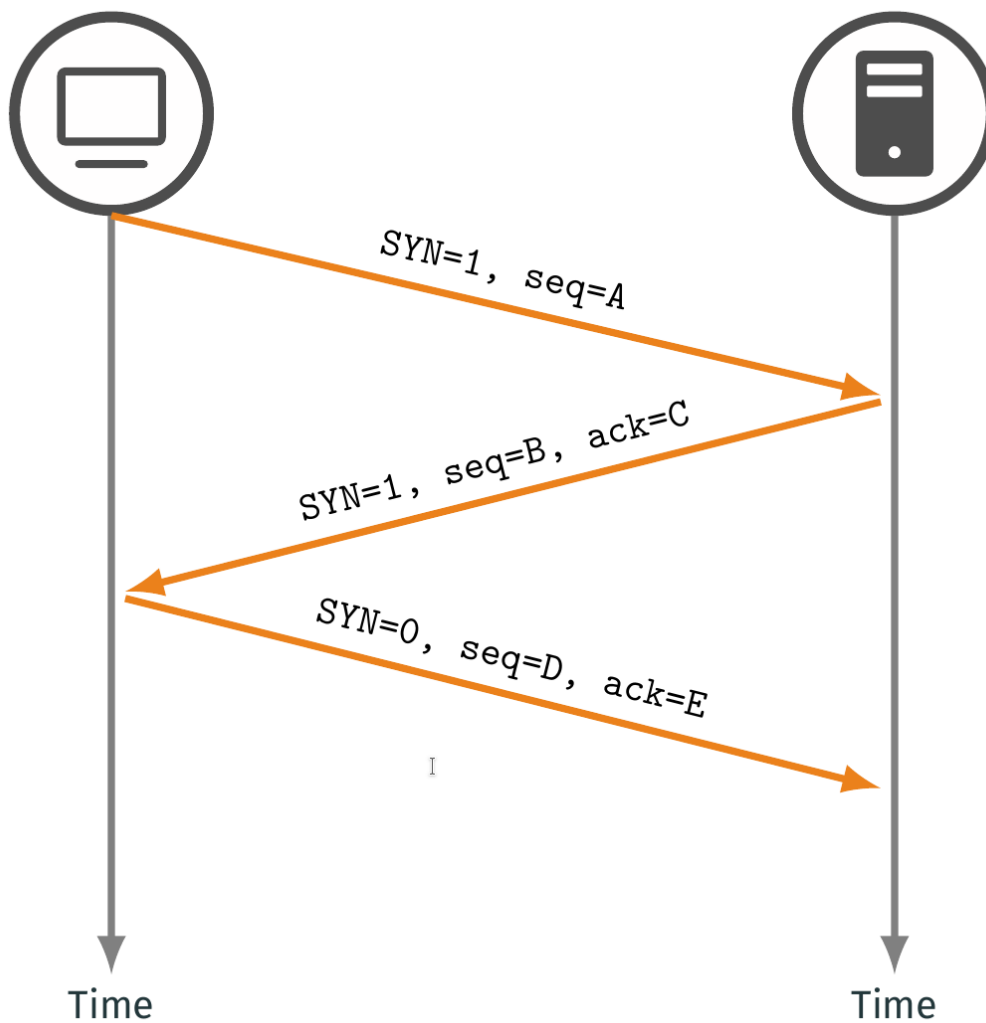
☐ True

---

Maks poeng: 1



- 13 The following figure illustrate a crucial step in the TCP connection management. To identify it, note that the SYN flag is set in two of the three packets. Here, seq and ack denote the sequence and the acknowledge number, respectively.



Assuming  $A=50$  and  $B=100$ , select the correct answer.

**Select one alternative:**

- ☒ C=51, D=51, E=101
- ☐ C=50, D=100, E=200
- ☐ C=51, D=101, E=102

Maks poeng: 1

**14** What network tool can be used to sniff packets?

**Select one alternative:**

☒ Wireshark

☐ Python.Socket

☐ Docker

☐ Flask

---

Maks poeng: 1

**15** In socket programming we regularly meet AF\_INET, both in UDP and TCP. What is it?

**Select one alternative:**

☐ It is just a shortcut for the developer to create only TCP sockets.

☒ It is a constant denoting the Address family for IPv4.

☐ Attribute For Internet Network Environment Transfer.

---

Maks poeng: 1

**16** Host A and Host B are connected by a link whose transmission rate is 2 Mbps and propagation delay is 20 ms. In a video call, Host A encodes video at 8 Mbps. Then, Host A groups the resulting bits into 1500-byte packets (assume that these packets do not have a header and that they are neither encapsulated nor split into other packets). As soon as Host A gathers a packet, it sends the packet to Host B. How much time elapses (in ms) from the time the first bit is created (because of the encoding at Host A) until Host B receives the first packet?

27.5 ms

---

Maks poeng: 4

- 17** Alice wants to send a 360 MB file to Bob. The path connecting Alice's and Bob's devices consists of three links, of rates 4 Mbps, 5 Mbps and 3 Mbps. Assuming no other traffic in the network, estimate the time it will take to transfer the file.

**Select one alternative:**

☐ 576 s

☐ 72 s

☐ 120 s

☒ 960 s

---

Maks poeng: 4

18 The following is a screenshot of Firefox that focuses on the headers of certain protocol.



1. Briefly and in your own words, explain the line that starts with GET.
2. Briefly and in your own words, explain one of the header fields in any of the two headers.
3. Write a short Python script using **requests** in Python to print all the response headers.

### Fill in your answer here

1. This line that starts with GET first get the path on 0.0.0.0:8000, where in this case it is just "/" then gets the protocol its supposed to use which in our instance is "HTTP/1.0"

2. The Accept-Language header shows what language the user prefer. Behind the "en" language in our instance we have the relative quality factor which tells how much the user prefers this language. So this user prefers the US-english and not the default english.

3.

Python:

```
import requests
```

```
response = requests.get(0.0.0.0:8000)
print(response)
print(response.headers)
```

Maks poeng: 5

- 19** Briefly and in your own words, compare two mail access protocols.

**Fill in your answer here**

#### POP3

Post office protol 3 is the simples one out of theese two. When you reccive a mail it is first stored on the mailserver, then whenever you acces the innbox folder on this server the mail is downloaded to your computer and then deleted off the mail server. So when you wish to view, you cannot use another computer since the mail is deleted off the server. This protocol also does not offer any syncornisation accros your computer, which means you can have diffrent folder structure on the same user on diffrent computers.

#### IMAP

Internet message access protocol is the oposite of POP3 and all of your computers will be at sync whenever they access the mailserver. Your mails is stored on the server until a computer deleates it, therefor you can acces the same mail over multiple devices.

Maks poeng: 3

- 20** The following figure illustrates the header of a fictional transport protocol called *tiny-UDP*.

8 bits	8 bits
Source port	Destination port
Length	Checksum

1. How many bytes can tiny-UDP carry as payload?

2. Consider the following tiny-UDP packet:

00010000 11101111 00000110 01101010 01001000 01101001

Does this packet have an error?

Explain your answers and calculations (if any).

**Fill in your answer here**

1. The maximum number of bytes that can be included in a tiny-UDP payload is  $(2^8 - 1)$  bytes plus the header bytes. This gives 255 bytes – 2 bytes = 253 bytes.

2.

Maks poeng: 6

- 21** 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 192.168.5.0/24. Also suppose that Subnet 1 is required to support up to 62 interfaces, Subnet 2 is to support up to 62 interfaces, and Subnet 3 is to support up to 14 interfaces. Provide three network addresses (of the form a.b.c.d/x) that satisfy these constraints and explain how you obtained them. In your explanation, detail how you obtain the routing prefix for each subnet and how you ensure that these subnets support the required number of interfaces.

**Fill in your answer here**

Network ID = 192.168.5.0/24 = 11000000 10101000 00000101 ????????

Subnet 1 = 11000000 10101000 00000101 0???????

Subnet 2 = 11000000 10101000 00000101 10???????

Subnet 3 = 11000000 10101000 00000101 11???????

a.b.c.0/25 -> 0+128=128 -> a.b.c.128/26 -> 128+64 -> a.b.c.192/26

Therefore:

Subnet 1 = 192.168.5.0/25

Subnet 2 = 192.168.5.128/26

Subnet 3 = 192.168.5.192/26

---

Maks poeng: 5

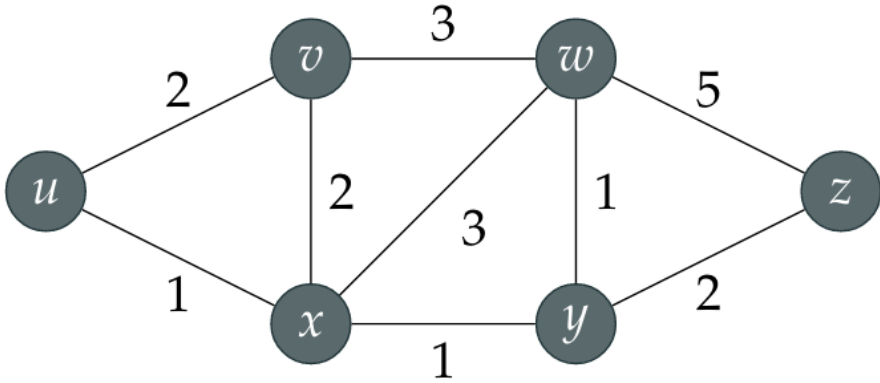
- 22** A 20-byte message gets encapsulated into a TCP segment which in turn gets encapsulated into an IPv6 datagram. What percentage of this datagram will be application data? Assume that the Options field is not being used.

25 .

---

Maks poeng: 3

23 Using Dijkstra's algorithm, find the least-cost path from **y** to every node in the following graph.



Show every step by filling in the table with header:

Step	$N'$	$D(u), p(u)$	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(z), p(z)$
------	------	--------------	--------------	--------------	--------------	--------------

Fill in your answer here

step	N		x	w	z	v	u
0	y		1,y	1,y	2,y	inf	inf
1	y,x		-	1,y	2,y	2,x	1,x
2	y,x,w		-	-	2,y	2,x	1,x
3	y,x,w,u		-	-	2,y	2,x	-
4	y,x,w,u,z		-	-	-	2,x	-
5	y,x,w,u,z		-	-	-	-	-

Maks poeng: 5

24 The following code corresponds to a *buggy* load balancer in Python using UDP.

```

load_balancer_buggy.py x
1  from socket import socket
2
3  # A list containing the hostnames of the available workers
4  workers = ["worker1", "worker2", "worker3"]
5
6  # Instantiate a UDP socket
7  sock = socket()
8  sock.bind("", 5550)
9
10 # Perform load balancing forever
11 while True:
12     for worker in workers:
13         # Receive a task
14         task, _ = sock.recvfrom(1024)
15         # Forward the task to the corresponding worker
16         sock.sendto(task, (worker, 5550))
17

```

Its job is to receive a task from a client and then forward it to one of the available workers (hosts that carry out tasks). The clients cannot send the tasks directly to the workers. However, once the task is completed, the worker sends its result directly to the corresponding client, i.e. the result does not go through the load balancer.

In your own words, explain whether the following statements are true or false.

1. Python will not allow to use the same UDP socket to both receive and forward tasks.
2. Lines 14 and 16 must be rewritten to let the workers know the addresses of the corresponding clients.
3. The content of **task** must be encoded before forwarding it.

*Note:* the comments mention what the next line should do, but there may be errors in that line. Not all errors appear on the list of statements that you have to explain.

**Fill in your answer here**

1.  
False
2.  
True, because we use UDP the workers have no way of knowing which client sent the request because UDP is not a connection oriented protocol
3.  
True, you must always encode before sending information, this is because they need to be converted to bits. If not the receiver cannot read the information it is given.

Maks poeng: 6



**25** The following is a screenshot of Wireshark that focuses on the header of a particular protocol.

~ User Datagram Protocol, Src Port: 63630, Dst Port: 443

Source Port: 63630

Destination Port: 443

Length: 1358

Checksum: 0x5e5e [unverified]

[Checksum Status: Unverified]

[Stream index: 2]

> [Timestamps]

UDP payload (1350 bytes)

Briefly and in your own words, answer the following questions.

1. What layer does this protocol belong to?
2. What services do this protocol provide?
3. Provide a short descriptions of two of its header fields as well as a short explanation of their respective values in the screenshot.

**Fill in your answer here**

1. This layer belong to the application layer of protocol beacuse this is UDP.
2. This protocol provides us the source port, destination port, lengt of the datagram and the checksum of the header and datafield
3. We have the source port which tells us what port this request was sent from, and we have the destination port which is the port it want to access.

---

Maks poeng: 4

**26** Hi! Hola! Hei!

Please, ignore this question.

It is just to have a field to add the marks of the mandatory quizzes and the mandatory assignment.

**Fill in your answer here**

---

Maks poeng: 40