Trådlös kommunikation och inbyggda system

Quiz Instructions

Questions: 16 Questions (1pt each)

The questions are shown one at a time, but you can go back to previous questions.

Time: 35min

The quiz responses are not shown until after the deadline.

Good Luck!

## Question 1 — 1 pts

Assume you want to write an application that collects data from temperature sensors in a network. To limit the amount of data packets to be sent to the base station, a node in the network does the following:

1. sense temperature,
2. wait for data from 2 neighboring nodes, but at most 10s
3. sum the values from the own sensor and the received data from the neighbors
4. send the result towards the base station

How would you write this application in contiki?
(receiving and forwarding queries from the basestation can be ignored)

Notation:

- sum is the variable to sum up the temperature values
- "wait" would be implemented as PROCESS_WAIT_EVENT_UNTIL(c)

- [ ] (Thread1): sense temperature and store value in the global variable sum → wait for 10s → send sum towards the basestation. (receive callback): add received value to the global variable sum for the first two received messages.
- [ ] (Thread1): sense temperature and store value in the global variable sum → wait 10s → add the value of the first two messages in the receive queue to the global variable sum and send sum towards the basestation.
- [ ] (Thread1): sense temperature and store value in the global variable sum (Thread1): wait 10s → send sum to towards the basestation (Thread3): wait for the first message → add received value to the global sum (Thread4): wait for the second message → add received value to the global sum → send sum towards the basestation
- [ ] (Thread1): sense temperature and store value in the global variable sum → wait 10s → send sum to towards the basestation (Thread2): wait for the first message → add received value to the global sum (Thread3): wait for the second message → add received value to the global sum → send sum towards the basestation

Next ▸

## Question 2 — 1 pts

Networked embedded systems are often implemented as event-driven applications. What is the main reason to do so?

- ○ energy efficient: event-driven operation allows nodes to sleep (put processor in low power mode) when nothing happens
- ○ memory efficient: an event-driven application uses less memory compared to a not event-driven application that implements the same functionality.
- ○ realtime: event-driven applications have better realtime properties
- ○ robustness: event-driven applications have less errors

◂ Previous  Next ▸

## Question 3 — 1 pts

TinyOS has bug prevention as design choice. What does this imply?

- ○ predictable memory allocation
- ○ put node asleep as much as possible
- ○ (computationally) efficient libraries
- ○ no blocking calls

## Question 5 — 1 pts

What is true about checkpointing in intermittent computing: [one correct answer]

- ○ Checkpointing guarantees that a process after reboot continues at the instruction after the last checkpoint.
- ○ Checkpointing guarantees that a process after reboot continues at the same instruction at which it run out of energy.
- ○ A checkpoint does not store the stack because its size is non-deterministic.
- ○ Checkpointing the process context is the most efficient option for applications with data freshness requirements.

## Question 6 — 1 pts

What is true about checkpointing in intermittent computing: [one correct answer]

- ○ Checkpointing guarantees that a process after reboot continues at the instruction after the last checkpoint.
- ○ Checkpointing guarantees that a process after reboot continues at the same instruction at which it run out of energy.
- ○ A checkpoint does not store the stack because its size is non-deterministic.
- ○ Checkpointing the process context is the most efficient option for applications with data freshness requirements.

## Question 7

1 pts

What statements about medium access are true? [more than one statement is true]

- [ ] TDMA and FDMA are channel partitioning protocols
- [ ] CSMA fulfills the desirable property that a node gets a throughput R/N in average (R: channel rate; N: number of nodes)
- [ ] Carrier sense does not guarantee the avoidance of collisions because of the propagation delay.
- [ ] Collision avoidance in 802.11: The random back-off timer is used to reduce the collision probability.

## Question 8

1 pts

Select the order of the different propagation ranges according to their range from the transmitter.

- ○ transmission range < detection range < interference range
- ○ transmission range < interference range < detection range
- ○ detection range < transmission range < interference range

## Question 9

1 pts

What statements relating frequency, wavelength and pathloss are true? [more than one true answer]

- [ ] The higher the frequency, the shorter the wavelength
- [ ] The longer the wavelength, the smaller the pathloss
- [ ] The higher the frequency, the lower the pathloss
- [ ] The longer the wavelength, the higher the pathloss

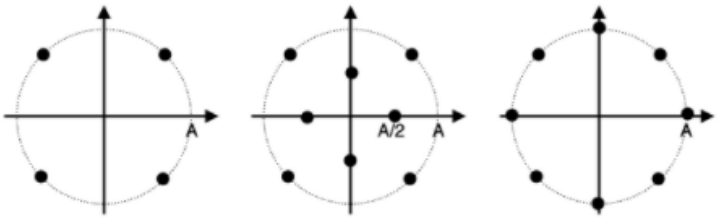
## Question 10 — 1 pts

QPSK: How many bits are encoded with one symbol?

- ○ 2
- ○ 1
- ○ 3
- ○ 4

## Question 11 — 1 pts

Consider the following constellations of the three modulation constellations X, Y and Z. Mark the true statements about these modulations. [more than one true statement]




modulation X  modulation Y  modulation Z

- ☐ Energy per symbol: modulation X has lowest energy per symbol.
- ☐ Energy per bit: modulation X has highest energy per bit.
- ☐ Data rate: assuming the same symbol duration, all modulations have the same data rate [bit/s]
- ☐ Low SNR: modulation X is expected to have the lowest symbol error rate in a low SNR scenario.

## Question 12 — 1 pts

There are three main alternatives to integrate IoT devices into the Internet from a protocol point of view:

- **Standard Internet protocol end-to-end** (example: IPv6, TCP, HTTP)
- **Lightweight Internet protocol** (example: 6LoWPAN, UDP, CoAP)
- **Non-Internet protocols** (example: ZigBee)

Which of the following statements is true? [more than one true answer]

- [ ] Lightweight Internet protocols and Non-Internet protocols require a gateway to translate between the protocols.
- [ ] The main reason to use Standard Internet protocols is performance (e.g., higher throughput)
- [ ] Lightweight Internet protocols are lightweight in the sense that they use less resources (e.g., shorter packets)
- [ ] Non-Internet protocols do not allow the IoT device to communicate with devices in the Internet.

## Question 13 — 1 pts

Header compression allows to reduce the amount of data to be transmitted between a sender and receiver.

In 6LoWPAN, what information is omitted in header fields, such that the receiver can reconstruct the omitted fields again? [one or more options are true]

- [ ] common values in all packets (e.g., version number, traffic class, ...)
- [ ] shared context (e.g., network prefix)
- [ ] duplicate information (e.g., payload length, link layer addresses, ...)
- [ ] information in the payload (e.g., host name)
- [ ] session context (e.g., values negotiated during session setup)

## Question 14 — 1 pts

Which of the below could be a Monte Carlo simulation? [one or more options are true]

- [ ] Simulating many 20min executions of a sensor network with COOJA simulator with different topologies.
- [ ] Calculating collisions that might occur when a large number of active tags send beacons randomly.
- [ ] Approximating the area of a figure with a complex shape.

## Question 15 — 1 pts

We distinguish two fundamental application models to build applications involving multiple IoT devices:

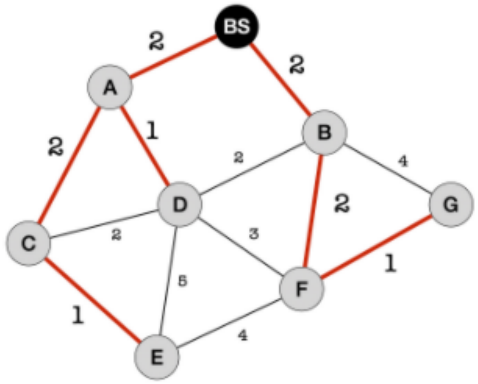
- Application logic on the small devices
- Application logic in the cloud

Mark advantages of the second one (Application logic in the cloud). [one or more options are true]

- [ ] lower communication latency
- [ ] easier to integrate devices from multiple manufacturers
- [ ] potentially more privacy as data is not sent to the cloud
- [ ] make use of existing security relations

## Question 16 — 1 pts

Consider the following network. The labels on the links give the expected number of transmissions (ETX) between the two nodes. The red edges mark the spanning tree.




Weight on edge: link ETX

Assume a data collection protocol along the spanning tree:

- The expected number of transmissions without aggregation is
- The expected number of transmissions with aggregation is

## Question 17 — 0 pts

Code of honor: I did do the quiz by myself and neither collaborated nor received help from anyone else.

**True**: I confirm that I followed the code of honor.
**False**: I admit that I did not follow the code of honor. [consider to not submit the quiz]

- ○ True
- ○ False

## Question 4 — 1 pts

Why does multi-threading require one stack per process?

- ○ every process has its own context (program control block, code, stack, data)
- ○ every process reacts on different events.
- ○ to avoid blocking calls

◂ Previous

Next ▸