Sample solution of the written examination in Operating Systems

November 17th 2014

Last name: ______________________________

First name: ______________________________

Student number: ______________________________

I confirm with my signature that I will process the written examination alone and that I feel healthy and capable to participate this examination.
I am aware, that from the moment, when I receive the written examination, I am a participant of this examination and I will be graded.

Signature: ______________________________

- Provide on all sheets (including the cover sheet) your *last name*, *first name* and *student number*.
- Use the provided sheets. Own paper must *not* be used.
- Place your *ID card* and your *student ID card* on your table.
- You are allowed to use a *self prepared, single sided DIN-A4 sheet* in the exam. Only *handwritten originals* are allowed, but no copies.
- You are allowed to use a non-programmable calculator.
- Answers, written with pencil or red pen are *not* accepted.
- Time limit: *90 minutes*
- Turn off your mobile phones!

Result:

Question:	1	2	3	4	5	6	7	8	9	10	Σ	Grade
Maximum points:	10	9	8	5	8	10	10	10	15	5	90	—
Achieved points:												

Last name: First name: Student number:

Question 1)

Points: ....

Maximum points: 0,5+0,5+1+1,5+3+0,5+3=10

a) Batch processing is always...

□ interactive ☒ non-interactive

b) Name an application of batch mode in practice, which is still popular today.

Batch files or shell scripts.

c) What is the difference between 8 bit, 16 bit, 32 bit and 64 bit operating systems?

The bit number indicates the memory address length, with which the operating system works internally.

d) Which three digital bus systems contains each computer system according to the Von Neumann Architecture?

Address bus, data bus und control bus.

e) Which tasks are carried out by the three digital bus systems of subtask d)?

Adress bus: Transmits memory addresses. Memory addresses and I/O devices are addressed via the address bus.
Data bus: Transmits data between CPU, main memory and I/O devices.
Control bus: Transmits commands (e.g. read and write requests) from the CPU and returns status signals from the I/O devices.

f) What is the Front Side Bus (FSB)?

It contains the address bus, data bus and control bus.

g) Name three possible ways for processes to read data from Input/Output devices.

Busy Waiting, Interrupt-driven and Direct Memory Access (DMA).

Question 2)

Points: ....

Maximum points: 1+1+1+1+1+2+1+1=9

a) What are tracks in HDDs?

The disc surfaces are magnetized in circular tracks by the heads.

b) What are sectors (= blocks) in HDDs?

The tracks are divided into small logical units (segments of a circle), which are called blocks or sectors.

c) What are cylinders in HDDs?

All tracks on all disks at a specific arm position are part of a cylinder.

d) What are clusters in HDDs?

Clusters are groups of sectors with a fixed size (e.g. 4 or 8 kB) and in modern operating systems, clusters are the smallest addressable unit of HDDs.

e) Which factors influence the access time of HDDs?

Average Seek Time and Average Rotational Latency Time.

f) Name four advantages of SSDs over HDDs.

Fast access time, low power consumption, no noise generation, mechanical robustness, low weight, the location of data has no effect of the data rate,. . . .

g) Name an advantage and a drawback of NOR memory.

Advantage: Random access for read and write operations $\Longrightarrow$ better latency compared with NAND memory.
Drawbacks: More complex ($\Longrightarrow$ expensive) construction, Higher power consumption compared with NAND memory, typically small capacities.

h) Name an advantage and a drawback of NAND memory.

Advantages: Lesser data lines $\Longrightarrow$ requires lesser surface area of NOR memory, lower manufacturing costs compared with NOR flash memory.
Drawback: No random access $\Longrightarrow$ poorer latency compared with NOR memory.

Last name: First name: Student number:

Question 3)

Points: ....

Maximum points: 8

a) How many drives are allowed to fail in a RAID 0 array without data loss?

If a drive fails, data loss occurs.

b) How many drives are allowed to fail in a RAID 1 array without data loss?

Data loss occurs only in case of the failure of all drives.

c) How many drives are allowed to fail in a RAID 5 array without data loss?

RAID 5 can handle the failure of up to one drive without data loss.

d) Please comment the statement: „A RAID array can be used to replace the regular backup of important data".

A RAID is not a backup replacement. Wrong file operations or virus attacks take place on all drives. Defects as caused by overvoltage (for example lightning damage) may destroy the entire system.

e) Why is it not useful to store all parity information on a single drive, but to distribute the parity information on all drives?

Each write operation on the RAID causes write operations on the dedicated parity drive $\Longrightarrow$ *bottleneck.*

f) What is the net capacity of a RAID 0 array?

The net capacity is n, if n is the number of drives.

g) What is the net capacity of a RAID 1 array?

The net capacity is equal to the capacity of the smallest drive.

h) What is the net capacity of a RAID 5 array?

The net capacity is $n - 1$, if n is the number of drives.

Last name: First name: Student number:

Question 4)

Points: ....

Maximum points: 5

Please mark for each one of the following statements, whether the statement is true or false.

a) Real mode is suited for multitasking systems.

□ True ☒ False

b) When static partitioning is used, internal fragmentation occurs.

☒ True □ False

c) When dynamic partitioning is used, external fragmentation cannot occur.

□ True ☒ False

d) When segmentation is used, the operating system maintains a segment table for each process.

☒ True □ False

e) Internal fragmentation cannot occur with segmentation.

☒ True □ False

f) External fragmentation cannot occur with segmentation.

□ True ☒ False

g) With paging, all pages have the same length.

☒ True □ False

h) Modern operating systems use only segmentation.

□ True ☒ False

i) One advantage of long pages is little internal fragmentation.

□ True ☒ False

j) Modern operating systems (for x86) operate in real mode and use only segmenting.

□ True ☒ False

Last name: First name: Student number:

Question 5)

Points: ....

Maximum points: 0,5+0,5+0,5+0,5+2+1+1+1+1=8

a) Do DOS/Windows file systems between uppercase and lowercase?

□ Yes ☒ No

b) Do UNIX file systems differentiate between uppercase and lowercase?

☒ Yes □ No

c) Do modern operating systems accelerate requests to stored data with a cache in the main memory.

☒ Yes □ No

d) Most operating systems operate according to the principle...

☒ write-back □ write-through

e) Name an advantage and a disadvantage of a cache in the main memory, which is used by the operating system to accelerate the requests to stored data.

Benefit: Better system performance.
Drawback: System crashes may cause inconsistencies.

f) What is an absolute path name?

It describes the entire path from the root to the destination (file or directory).

g) What is a relative path name?

A path, which does not begin with the root.

h) What sort of data processing is maximum accelerated by defragmenting?

A continuous arrangement would maximum accelerate to continuous forward reading of the data because no more seek times occur.

i) For which scenario is defragmenting useful?

Only if the seek times are huge.

Last name: First name: Student number:

Question 6)

Points: ....

Maximum points: 10

a) Why does the process control block not store all process context information?

The content of the address space (user context) is not stored in the process table because it is too huge in size.

b) What is the task of the dispatcher?

It carries out the state transitions of the processes.

c) What is the task of the scheduler?

It specifies the execution order of the processes by using a scheduling algorithm.

d) What is a zombie process?

A terminated process which still appears in the process table. Its PID can not yet be assigned to a new process.

e) What is the task of the process control block?

The operating system stores the hardware context and system context in the process control block.

f) What is the PID?

Process ID ⟹ the unique process identification.

g) What is the PPID?

Parent Process ID ⟹ the process identification of the parent process.

h) What is the effect of calling the system call `fork()`?

If a process calls `fork()`, an identical copy is started as a new process.

i) What is the effect of calling the system call `exec()`?

The system call `exec()` replaces a process with another one.

j) What is `init` and what is its task?

`init` is the first process in Linux/UNIX. It has the PID 1. All running processes originate from `init`. `init` = father of all processes.

Last name: First name: Student number:

Question 7)

Points: ....

Maximum points: 2+2+3+1+2=10

a) How does static multilevel scheduling work?

The list of processes in state `ready` *is split into multiple sublists. For every sublist, a different scheduling strategy is used. The sublists have different priorities or time multiplexes.*

b) How does multilevel feedback scheduling work?

It works like multilevel scheduling with multiple queues. Each queue has a different priority or time multiplex. Each new process is inserted in the top queue. This way it has the highest priority. For each queue, round-robin is used. Is a process resigns the CPU on voluntary basis, it is inserted in the same queue again. If a process utilized its complete time slice, it is inserted in the next lower queue, with has a lower priority.

c) Which scheduling strategies operate preemptive?

☐ First Come First Served
☒ Round Robin with time quantum
☐ Shortest Job First
☐ Longest Job First
☒ Shortest Remaining Time First
☒ Longest Remaining Time First
☒ Fair share
☒ Static multilevel scheduling
☒ Multilevel feedback scheduling

d) What is the advantage of signaling compared with busy waiting?

Computing time of the CPU is wasted when busy waiting is implemented because the CPI it is again and again occupied by the waiting process. Lesser CPU workload creates signaling because the waiting process gets blocked and later unblocked.

e) Which four conditions must be fulfilled at the same time as precondition that a deadlock can arise?

☐ Recursive function calls
☒ Mutual exclusion
☐ Frequent function calls
☐ Nested `for` loops
☒ No preemption
☒ Hold and wait
☐ > 128 processes in `blocked` state
☐ Iterative programming
☒ Circular wait
☐ Queues

Last name: First name: Student number:

Question 8)

Points: ....

Maximum points: 10

a) What must be considered, when inter-process communication via shared memory segments is used?

The processes need to coordinate the accesses themselves and to ensure that their memory accesses are mutually exclusive. A receiver process, cannot read data from the shared memory, before the sender process has finished its current write operation. If access operations are not coordinated carefully, inconsistencies occur.

b) According to which principle operate message queues?

□ Round Robin □ LIFO ☒ FIFO □ SJF □ LJF

c) How many processes can communicate with each other via a pipe?

2

d) What is the effect, when a process tries to write data into a pipe without free capacity?

The writing process is blocked.

e) Which two different types of pipes exist?

Anonymous pipes and named pipes.

f) Which two different types of sockets exist?

Connection-less sockets (also called: datagram sockets) and connection-oriented sockets (also called: stream sockets).

g) Communication via pipes works...

□ memory-based ☒ stream-based
□ object-based □ message-based

h) Communication via message queues works...

□ memory-based □ stream-based
☒ object-based □ message-based

i) Communication via shared memory segments works...

☒ memory-based □ stream-based
□ object-based □ message-based

j) Communication via sockets works...

□ memory-based □ stream-based
□ object-based ☒ message-based

Last name: First name: Student number:

Question 9)

Points: ....

Maximum points: 15

In a warehouse, packages are delivered constantly by a supplier and picked up by two deliverers. The supplier and the deliverers need to pass through a gate. The gate can always be passed only by a single person. The supplier brings three packages with every shipment to the incoming goods section. One of the deliverers can pick two packages with every pickup from the outgoing goods section. The other deliverer can pick only a single package per pickup from the outgoing goods section. Exactly one process `Supplier`, one process `Deliverer_X` and one process `Deliverer_Y` exist.

For synchronizing the three processes, create the required semaphores, assign them values and insert semaphore operations. These conditions must be met:

- Only a single process can pass through the gate.
  It is impossible that multiple processes pass though the gate simultaneously.
- Only one of both existing deliverers can access the outgoing goods section.
  It is impossible that both deliverers access the outgoing goods section simultaneously.
- It should be possible that the supplier and one of the deliverers can simultaneously unload and pick goods.
- The capacity of the warehouse is 10 packages.
- No deadlocks are allowed.
- At the beginning, the warehouse contains no packets and the gate, as well as the incoming goods section and the outgoing goods section are free.

Source: TU-München, Übungen zur Einführung in die Informatik III, WS01/02

These semaphores are required:

- *Boolean semaphore `gate` for the mutual exclusion of the gate with initial value `1`. Initially, the gate is empty.*
- *Boolean semaphore `outgoing_goods_section` for the mutual exclusion of the outgoing goods section with initial value `1`. Initially, the outgoing goods section is empty.*
- *Semaphore `free` for counting the free spaces in the warehouse with initial value `10`. Initially all spaces are free.*
- *Semaphore `occupied` for counting the occupied spaces in the warehouse with initial value `0`. Initially no space is occupied.*

Question 9 – Continuation)

```
sema gate                   = 1
sema outgoing_goods_section = 1
sema free                   = 10
sema occupied               = 0


Supplier                    Deliverer_X                 Deliverer_Y
{                           {                           {
  while (TRUE)                while (TRUE)                while (TRUE)
  {                           {                           {
    P(gate);                    P(gate);                    P(gate);
    <Pass through gate>;        <Pass through gate>;        <Pass through gate>;
    V(gate);                    V(gate);                    V(gate);

                                P(outgoing_goods_section);  P(outgoing_goods_section);
    <Enter incoming             <Enter outgoing             <Enter outgoing
    goods section>;             goods section>;             goods section>;

    P(free);
    P(free);                    P(occupied);
    P(free);                    P(occupied);                P(occupied);
    <Unload 3 packets>;         <Pick 2 packets>;           <Pick 1 packet>;
    V(occupied);                V(free);                    V(free);
    V(occupied);                V(free);
    V(occupied);

    <Leave incoming             <Leave outgoing             <Leave outgoing
    goods section>;             goods section>;             goods section>;
                                V(outgoing_goods_section);  V(outgoing_goods_section);

    P(gate);                    P(gate);                    P(gate);
    <Pass through gate>;        <Pass through gate>;        <Pass through gate>;
    V(gate);                    V(gate);                    V(gate);
    }                           }                           }
}                           }                           }
```

Last name: First name: Student number:

Question 10)

Points: ....

Maximum points: 1+1+1+1+0,5+0,5=5

a) What is the difference between Semaphores versus blocking?

In contrast to semaphores, can locks only be used to permit a single process entering the critical section at the same time.

b) What is a binary semaphore?

They are initialized with value 1 and ensure that two or more processes can not simultaneously enter their critical sections.

c) What is a strong semaphore?

Operates according to the FIFO principle. Typical sort of the semaphore, which is provided by the operating system.

d) What is a weak semaphore?

Does not set the order, in which the processes are fetched from the queue.

e) Which type of semaphores has the same functionality as the mutex?

Binary semaphore.

f) Which Linux/UNIX command returns information about existing shared memory segments, message queues and semaphores?

ipcs