Sample solution of the written examination

Operating Systems

July 22nd 2019

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: ______________________________

- Use the provided sheets. Own paper must *not* be used.
- 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.
- Do *not* use a red pen.
- The time limit ist *90 minutes.*
- Turn off your mobile phones!

Result:

Question:	1	2	3	4	5	6	7	8	9	10	11	Σ	Grade
Maximum points:	8	14	8	4	10	8	4	8	10	9	7	90	—
Achieved points:													

1.0: 90.0-85.5, **1.3**: 85.0-81.0, **1.7**: 80.5-76.5, **2.0**: 76.0-72.0, **2.3**: 71.5-67.5,
2.7: 67.0-63.0, **3.0**: 62.5-58.5, **3.3**: 58.0-54.0, **3.7**: 53.5-49.5, **4.0**: 49.0-45.0, **5.0**: <45

Last name: First name: Student number:

Question 1)

Points: ....

Maximum points: 1+1+2+2+2=8

a) At any given moment, only a single program can be executed. What is the technical term for this operation mode?

Singletasking.

b) What is the name of the quasi-parallel program or process execution?

Multitasking.

c) Describe the structure of a monolithic kernel.

Monolithic kernels contain functions for...

- memory management
- process management
- interprocess communication
- hardware management (drivers)
- file systems

Outside the kernel are the user processes.

d) Describe the structure of a microkernel.

The kernel contains only...

- essential functions for memory management and process management
- functions for process synchronization and interprocess communication
- essential drivers (e.g. for system start)

Device drivers, file systems, and services (servers) are located outside the kernel and run equal to the user applications in user mode.

e) Describe the structure of a hybrid kernel.

Hybrid kernels are a tradeoff between monolithic kernels and microkernels. They contain for performance reasons some components, which are never located inside microkernels. It is not specified which additional components are located inside hybrid kernels.

Last name: First name: Student number:

Question 2)

Points: ....

Maximum points: 1+2+2+3+6=14

a) Name the two groups of Input/Output devices for computer systems that are distinguished according to their minimum transfer unit.

Character devices and block devices.

b) Describe the different operating principles of the groups of subtask a).

Character devices: On arrival/request of each single character, communication with the CPU always takes place.

Block devices: Data transfer takes place only when an entire block (z.B. 1-4 kB) exists.

c) Name two examples for each group from subtask a).

Character devices: Mouse, keyboard, printer, terminal, magnetic tape...

Block devices: HDD, SSD, CD/DVD drive, floppy drive...

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

- *Busy waiting*
- *Interrupt-driven*
- *Direct Memory Access (DMA)*

e) Name a benefit and a drawback for each possible way from subtask d).

- *Busy waiting*
  - *Benefits: No additional hardware required*
  - *Drawbacks: Causes CPU workload, slows down simultaneous processing of multiple processes*
- *Interrupt-driven*
  - *Benefits: The CPU is not blocked, allows the simultaneous execution of multiple processes*
  - *Drawbacks: Additional hardware (interrupt controller) is required*
- *Direct Memory Access (DMA)*
  - *Benefits: Reading data causes no CPU workload, simultaneous execution of multiple processes is not slowed down*
  - *Drawbacks: Additional hardware (DMA controller) is required*

Question 3)

Points: ....

Maximum points: 2+2+2+2=8

A hard disk drive provides these information:

```
IBM Travelstar                                   MODEL: DBCA-204860 E182115 T
RATED: 5V 500mA                               MADE IN THAILAND BY IBM STORAGE
P/N: 21L9510   4090 MB                                                16NOV99
FRU: 22L0018   MLC:F41941                   (7944 CYL. 16 HEADS. 63 SEC/T)
```

a) Calculate the capacity of one side of one disk of the hard disk drive.
(Provide the calculation steps!)

Note: The number of cylinders (`CYL`) is equal to the number of tracks per disc. The size of the sectors (`SEC`) is 512 Byte.

Solution: The capacity of one side of a disk = number of cylinders ∗ Number of sectors per track (`SEC/T`) ∗ bytes per sector (block).

$7,944 * 63 * 512 = 256,241,664$ Byte

b) Calculate the capacity of one track of the hard disk drive.
(Provide the calculation steps!)

solution: Size of a track = number of sectors per track (`SEC/T`) ∗ bytes per sector (block).

$63 * 512 = 32,256$ Byte

c) Calculate the total capacity of the hard disk drive.
(Provide the calculation steps!)

*Solution: Total capacity = capacity of one disk * number of heads (`HEADS`).*

$256,241,664 * 16 = 4,099,866,624$ Byte

d) How many disks does the hard disk drive have? *Note: Each disk has two sides.*
(Explain your answer!)

If the logical geometry matches the physical geometry, then the hard disk has 16 surfaces because of the 16 heads and thus 8 disks.

Last name: First name: Student number:

Question 4)

Points: ....

Maximum points: 4

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) In protected mode, each process is executed in its own copy of the physical address space, which is protected from other processes.

☒ True □ False

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

☒ True □ False

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

□ True ☒ False

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

☒ True □ False

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

□ True ☒ False

g) A drawback of short page page table can become huge.

☒ True □ False

h) When paging is used, the MMU translates the logical memory addresses into physical memory addresses.

☒ True □ False

Last name: First name: Student number:

Question 5)

Points: ....

Maximum points: 10

a) Describe which information inodes store.

An inode stores a file's metadata, except the file name.

b) Name three examples of metadata in the file system.

Metadata are among others the size, UID/GID, permissions and date.

c) Describe what a cluster in the file system is.

File systems address clusters and not blocks of the storage device. Each file occupies an integer number of clusters.

d) Describe how a UNIX file system (e.g. ext2/3), which does not implement extents, can address more than 12 clusters.

If a file requires more than 12 clusters, these clusters are indirectly addressed via clusters that contain only cluster numbers.

e) Describe how directories in the Linux file systems are technically implemented.

Directories are just text files, which contain the names and paths of files.

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

☒ write-back ☐ write-through

g) `/home/<username>/Mail/inbox/` is an/a...

☒ absolute path name ☐ relative path name

h) Describe what information the boot sector of a file system stores.

The boot sector contains executable machine code („boot loader“), which starts the operating system, and information about the file system.

i) Describe what information the super block of a file system stores.

Contains information about the file system, e.g. number of inodes and clusters.

j) Explain why some file systems (e.g. ext2/3) do combine the clusters of the file system to block groups.

Inodes (metadata) are physically located close to the clusters, they address.

Last name: First name: Student number:

Question 6)

Points: ....

Maximum points: 2+1+1+3+1=8

a) Describe what the File Allocation Table (FAT) is and describe the information it stores.

The FAT (File Allocation Table) is a table of fixed size. For each cluster in the file system, an entry exists in the FAT with the following information about the cluster:

- *Cluster is free or the storage medium is damaged at this point.*
- *Cluster is occupied by a file and stores the address of the next cluster, which belongs to the file or it is the last cluster of the file.*

b) Describe the objective of the journal in a journaling file system.

In the journal, write operations are collected before being committed to the file system.

c) Describe a benefit of using a journaling file system compared with using a file system without a journal.

After a crash, only the files (clusters) and metadata must be checked, for which a record exists in the journal.

d) Name the three values that are required to store an extent.

Start (cluster number) of the area (extent) in the file.

Size of the area in the file (in clusters).

Number of the first cluster on the storage device.

e) Describe the benefit of using extents compared with direct addressing of the clusters.

Instead of multiple individual clusters numbers, only 3 values are required: Lesser overhead.

Question 7)

Points: ....

Maximum points: 4

a) Describe the result of defragmenting a file system.

Logically related clusters are placed physically close to each other on the storage device.

b) Describe the sort of data processing that is maximum accelerated by defragmenting.

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

c) Describe the scenario where defragmenting is useful.

Only if the seek times are huge, defragmentation makes sense.

d) Does defragmenting SSDs make sense? *(Explain your answer!)*

It doesn't make sense. For SSDs the position of the clusters is irrelevant for the latency. In addition, the storage cells of SSDs have a limited number of write/erase cycles.

Last name: First name: Student number:

Question 8)

Points: ....

Maximum points: 8

a) Describe the effect of calling the system call `fork()`.

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

b) Describe the effect of calling the system call `exec()`.

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

c) Describe what `init` is and what its task is.

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

d) Name the differences of a child process from the parent process shortly after its creation.

The PID and the memory areas.

e) Describe the effect, when a parent process is terminated before the child process.

If a parent process terminates before the child process, it gets `init` as the new parent process assigned. Orphaned processes are always adopted by `init`. The PPID of the cild process then becomes value 1.

f) Describe what data the Text Segment contains.

It contains the program code (machine code).

g) Describe what data the Heap contains.

Constants and variables, which get values assigned in global declarations (outside of functions).

h) Describe what data the Stack contains.

Command line arguments, environment variables of the program call, call parameters and return address of functions, local variables of functions.

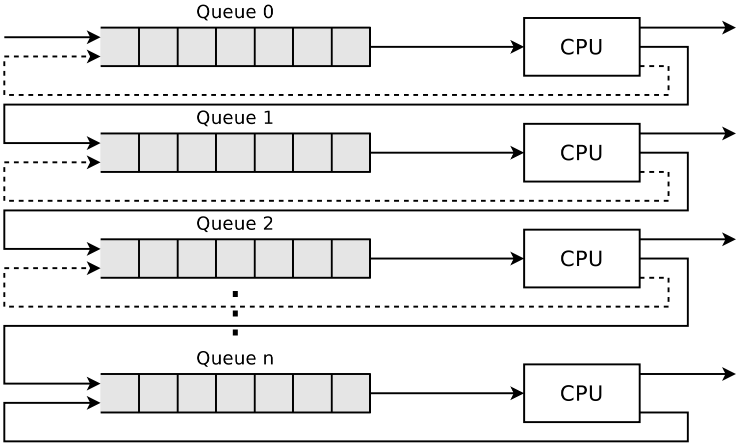
Question 9)

Points: ....

Maximum points: 6+2+2=10

a) Explain how multilevel feedback scheduling works.
(*An illustration can be useful here.*)

It works with multiple queues. Each queue has a different priority or time multiplex. Each new process is inserted in the top queue and this way it has the highest priority. For each queue, Round Robin is used. If 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.



Source: William Stallings. Betriebssysteme. Pearson Studium. 2003

b) Name four scheduling strategies that are fair.

First Come First Served, *Round Robin with time quantum*, *Earliest Deadline First*, *Fair share*, *Multilevel feedback scheduling.*

c) Name four scheduling strategies that do not need to know the *execution time* of the processes.
(Note: Only those scheduling procedures are searched, that can be used under realistic conditions.)

Priority-driven scheduling, *First Come First Served*, *Round Robin with time quantum*, *Earliest Deadline First*, *Fair share*, *Multilevel feedback scheduling.*

Question 10)

Points: ....

Maximum points: 2+7=9

a) Mark four conditions that must be fulfilled at the same time as precondition that a deadlock can occur.

☐ 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

b) Does a deadlock occur?
Perform the deadlock detection with matrices.

$$\text{Existing resource vector} = \begin{pmatrix} 4 & 8 & 6 & 6 & 5 \end{pmatrix}$$

$$\text{Current allocation matrix} = \begin{bmatrix} 0 & 2 & 1 & 0 & 0 \\ 2 & 3 & 1 & 0 & 4 \\ 1 & 0 & 2 & 1 & 1 \end{bmatrix} \qquad \text{Request matrix} = \begin{bmatrix} 3 & 3 & 2 & 4 & 5 \\ 0 & 3 & 1 & 4 & 0 \\ 0 & 2 & 3 & 5 & 4 \end{bmatrix}$$

The existing resource vector and the current allocation matrix are used to calculate available resource vector.

$$\textit{Available resource vector} = \begin{pmatrix} 1 & 3 & 2 & 5 & 0 \end{pmatrix}$$

Only process 2 can run with this available resource vector. The following available resource vector results when process 2 has finished execution and deallocates its resources.

$$\textit{Available resource vektor} = \begin{pmatrix} 3 & 6 & 3 & 5 & 4 \end{pmatrix}$$

Only process 3 can run with this available resource vector. The following available resource vector results when process 3 has finished execution and deallocates its resources.

$$\textit{Available resource vector} = \begin{pmatrix} 4 & 6 & 5 & 6 & 5 \end{pmatrix}$$

Process 1 is not blocked.

No deadlock occurs.

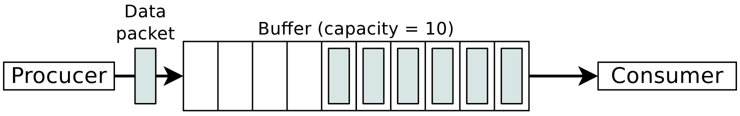
Last name: First name: Student number:

Question 11)

Points: ....

Maximum points: 7

- A producer writes data into a buffer and the consumer removes it.
- Mutual exclusion is necessary in order to avoid inconsistencies.
- If the buffer has no more free capacity, the producer must be blocked.
- If the buffer is empty, the consumer must be blocked.



For synchronizing the two processes, create the required semaphores, assign them initial values and insert semaphore operations.

```
typedef int semaphore;           // semaphores are of type integer
semaphore filled = 0;            // counts the occupied locations in the buffer
semaphore empty  = 10;           // counts the empty locations in the buffer
semaphore mutex  = 1;            // controls access to the critial sections

void producer (void) {
  int data;

  while (TRUE) {                 // infinite loop
    createDatapacket(data);      // create data packet
    P(empty);                    // decrement the empty locations counter
    P(mutex);                    // enter the critical section
    insertDatapacket(data);      // write data packet into the buffer
    V(mutex);                    // leave the critical section
    V(filled);                   // increment the occupied locations counter
  }
}

void consumer (void) {
  int data;

  while (TRUE) {                 // infinite loop
    P(filled);                   // decrement the occupied locations counter
    P(mutex);                    // enter the critical section
    removeDatapacket(data);      // pick data packet from the buffer
    V(mutex);                    // leave the critical section
    V(empty);                    // increment the empty locations counter
    consumeDatapacket(data);     // consume data packet
  }
}
```