

## Solution for the Examination Systemarchitecture

April 12, 2002

### Aufgabe/Question 1 (Zum Aufwärmen/Warm up, 2 + 4 + 1 + ... + 1 Punkte/marks)

1. „Worin unterscheiden sich **lokale** von **globalen Seitenersetzungstrategien**?“  
“*How do local page replacement policies differ from global ones.*”

Local page replacement takes place within the resident-set of the task that has initiated the page-fault (**1 mark**). Whilst with global page replacement *page-frame stealing* can occur, i.e. a task T may swap-in a page to a page-frame which has belonged to task T (**1 mark**). (notion: pf-stealing is not necessary)  
(if someone mentions, that only unpinned pages can be replaced, add an extra mark)

2. „Zählen Sie **vier praxisgerechte Plattenzugriffstrategien** (*disk scheduling policies*) auf:“  
“Enumerate **four usable disk scheduling policies.**”

4 out of FCFS, LCFS, Priority, SCAN, C-SCAN, N-step SCAN, F-SCAN,  
SSeekTF, SServiceTF, Proportional-Share Scheduler (**each 1 mark**).

Einige der folgenden Aussagen sind korrekt, einige inkorrekt. Unterstreichen Sie „korrekt“, wenn die Aussage korrekt ist, unterstreichen Sie „inkorrekt“, wenn die Aussage inkorrekt ist.  
*Some of the following statements are correct, some are incorrect. Underline “korrekt” if the statement is correct; underline “inkorrekt” if the statement is incorrect!*

3. „In UNIX werden **alle Betriebssystemfunktionen** im privilegierten Modus ausgeführt.“  
“*In UNIX all operating system functions are executed in privileged mode (kernel mode).*”

korrekt

inkorrekt

4. „Beim **symmetrischen Multiprocessing** kann **jede ausführbare System- oder Anwendungsaufgabe** (*system or user task*) von jedem Prozessor bearbeitet werden.“  
“*With symmetric multiprocessing each executable system- or user-task may run on every CPU.*”

korrekt

inkorrekt

5. „Ein **Vorteil** einer **größeren Zeitscheibe** (*larger time quantum*) beim **Round-Robin-Scheduling** ist der **reduzierte Scheduling-Overhead**.“  
“*One advantage of a larger time quantum with round-robin scheduling is the reduced scheduling overhead.*”

korrekt

inkorrekt

6. „In einem System mit **residentem Kern** benötigt man für den **Kernbereich keine** Speicher-  
verwaltung.“  
“*In a system with a resident kernel you don't need memory management for the kernel area.*”

korrekt

inkorrekt

7. „Das Attribut **Referenzzähler** (*reference count*) im **Dateikopf** (*inode*) einer **UNIX-Datei** repräsentiert die **Anzahl** von **symbolischen** oder **Softlinks** auf diese Datei.“

“*The attribute **reference count** in the **inode** of a **UNIX-file** keeps track of the **number of symbolic or soft links** to this file.*”

korrekt

inkorrekt

8. „Ein (Online-)Video-on-demand-Service gehört zur Klasse der Anwendungen mit **harten** Echtzeitanforderungen.“

“*An online-video-on-demand-service belongs to the class of applications with **hard** real-time requirements.*”

korrekt

inkorrekt

## Aufgabe/Question 2

(6 + 2 + 4 Punkte/marks)

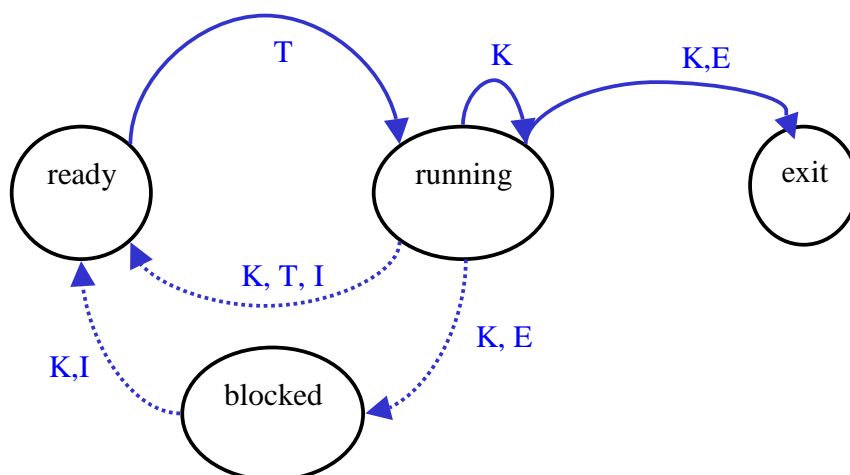
1. „Gegeben sei folgende Liste von Ereignissen, die Anlass zu einer **Kernaktivität** sein können:“

“*The following is a list of “events” which may cause the **kernel to run**:*”

Label	Event	Description
K	kernel-call exception	exception handler executes because of a kernel-call instruction
E	other exceptions	exception handler executes because of page-fault, arithmetic error, etc.
T	timer interrupt	timer interrupt handler executes
I	other interrupts	non-timer interrupt handler executes (e.g. console interrupt)

Ein **Einprozessorsystem** verwende ein verdrängendes Round-Robin Schedulingverfahren für alle Kernel-Level-Threads. Ergänzen Sie im unten angegebenen **Threadzustandsmodell** die möglichen **Threadzustandsübergänge** durch **entsprechende Pfeile** unter Verwendung der oben angegebenen „Labels“.

*A single-processor system uses a preemptive round-robin thread scheduler for all kernel-level threads. Complete the following **thread-state model** by adding the **corresponding arc** labeled by one of the above **labels** if you think that a **state-transition** may occur due to this system event.*



(0.5 marks for each labeled arc, 0.5 marks for all arcs)

2. „Erklären Sie den Unterschied zwischen einem **verdrängenden Scheduler** (*scheduling with preemption*) und einem **nicht-verdrängenden Scheduler!**“  
 “*Explain the difference between a preemptive scheduler and a non-preemptive scheduler.*”

With non-preemptive scheduling threads may run on a CPU as long as they voluntarily give up the CPU via `yield` or as they have to leave the CPU due to a blocking kernel-operation (**1 mark**). With preemptive scheduling threads may lose the CPU due to a new thread (with higher priority) in the ready-queue or because the time-slice has expired (**1 mark**).

3. „Eine Applikation enthalte zwei Threads, einen Produzenten und einen Konsumenten. Der folgende Pseudocode beschreibt das Verhalten der beiden Threads.“  
 “*An application consists of two threads, a Producer and a Consumer. The following pseudo-code describes the behavior of the two threads.*”

```
data buffer[10];          /*array of buffers to hold produced data*/
semaphore sem = 0;       /*counting semaphore to synchronize the
                           accesses to the buffers*/

Producer;
FOR I FROM 1 TO 10 DO
    read from file into buffer[I]
    V(sem)
ENDFOR
Consumer;
FOR I FROM 1 TO 10 DO
    P(sem)
    compute contents of buffer[I]
ENDFOR
```

Der Produzent muß bei jedem Lesen von der Datei  $t_{\text{read}}$  Zeiteinheiten warten, wobei er blockiert wird. Der Konsument benötigt  $t_{\text{compute}}$  Zeiteinheiten, um den Wert eines Pufferinhalts zu verarbeiten. Sonstige Bearbeitungszeiten beider Threads seien vernachlässigbar.

- Wenn beide Threads als **reine User-Level Threads** implementiert sind, wie lange dauert dann die **Gesamtbearbeitungszeit der Applikation?**
- Wenn beide Threads als **Kernel-Level-Threads** implementiert sind, wie lange dauert dann die **Gesamtbearbeitungszeit?**

Hinweis: Ihre Antwort sollte ein geschlossener Ausdruck aus  $t_{\text{read}}$  und/oder  $t_{\text{compute}}$  sein.

Suppose the Producer blocks for a time  $t_{\text{read}}$  each time it reads from the file, and the Consumer takes  $t_{\text{compute}}$  to compute a buffer's worth of data. The execution-time of the remaining parts of the pseudo-code for both threads is negligible.

- If the threads are implemented entirely as **pure user-level threads**, what will be the **total time required for the complete application?**
- If the threads are implemented entirely as **kernel-level threads**, what will be the **total time required for the complete application?**

Hint: Your answer should be a single expression involving  $t_{\text{read}}$  and/or  $t_{\text{compute}}$ .

a)  $10 \cdot (t_{\text{read}} + t_{\text{compute}})$  (**2 marks**)

b)  $9 \cdot \max\{t_{\text{read}}, t_{\text{compute}}\} + t_{\text{compute}} + t_{\text{read}}$  (**2 marks**)

**Aufgabe 3 / Question 3:****(4 + 4 + 4 Punkte/marks)**

1. „Beschreiben Sie das Phänomen **Prioritätsumkehr** (*priority inversion*) in einem System, das eine **prioritätsgesteuerte, verdrängende Umschaltung** (*scheduling*) mit **exklusiven Betriebsmitteln** (*mutual exclusive resources*) benutzt. Ihre Erläuterung sollte den Unterschied zwischen **kontrollierter** und **unkontrollierter Prioritätsumkehr** (auch begrenzte bzw. unbegrenzte Prioritätsumkehr genannt) beinhalten, ferner eine Begründung dafür, warum diese Unterscheidung wichtig ist.“  
*“Describe the phenomenon **priority inversion** in a system that uses **priority-driven preemptive scheduling** with **mutually exclusive resources**. You should include an explanation of the difference between **controlled** and **uncontrolled priority inversion** (also called **bound** and **unbound priority inversion**), and a reason why this distinction is important.”*

Priority inversion occurs when progress by a high priority task is prevented or delayed by a low priority task because the low priority task hold a resource the high priority task needs to complete. **(2 marks)**

Uncontrolled priority inversion is where the duration of priority inversion is potentially infinite. For example, an always-runnable medium priority task in the above scenario would prevent the low priority task from releasing the resource and hence the high priority task would suffer priority inversion indefinitely. **(1/2 mark)**

Controlled priority inversion is where priority inversion occurs but its duration is strictly limited by some combination of priority, scheduling, and resource allocation policy. **(1/2 mark)**

Distinction is important as control priority inversion guarantees progress, uncontrolled priority inversion can halt progress of the system. **(1 mark)**

2. „**Betriebsmittelvergabeprotokolle** können in einem System, das eine prioritätsgesteuerte, verdrängende Umschaltung mit exklusiven Betriebsmitteln benutzt, dazu verwendet werden, die Prioritätsumkehr zu begrenzen und zu minimieren.
- Nennen Sie **zwei solcher Protokolle**.
  - Beschreiben Sie deren **Wirkungsweise** im Detail. Sie sollten hierzu sowohl die **Umschaltstrategie**, die **Betriebsmittelvergabestrategie**, als auch die **Prioritäts-zuweisung** für jedes der beiden Protokolle erläutern.
  - Vergleichen Sie Vor- und Nachteile** beider Protokolle.“

*“Resource allocation protocols can be used in a system that uses priority-driven preemptive scheduling with mutually exclusive resources to bound and minimize priority inversion.*

- Name **two such protocols**. **(1 mark)***
- Describe how your two chosen protocols **work in detail**. You might need to describe **scheduling, allocation, and priority management rules** for each protocol. **(2 marks)***
- Compare the **advantages and disadvantages** of your chosen protocols.” **(1 mark)***

**Non preemptive critical sections:** Any job holding a resource is not preempted or any job holding a resource is boosted to the highest priority in the system.

Pro: simple, no a priori knowledge of resource requirements needed

Con: blocks higher priority jobs even with no conflicting resources

**Priority inheritance:** When a high-priority job blocks due to the fact that a low priority job holds a resource, than this low-priority job inherits the high-priority as long as it uses that resource.

Pro: no a priori knowledge of resource requirements needed

Con: chained blocking, does not prevent deadlock

**Priority Ceiling Protocol:** The resources required by all jobs are known a priori. The *priority ceiling* of a resource is equal to the highest priority of all jobs that use the resource. The priority ceiling of the system is the highest priority ceiling of all resources **currently in use**. Priority inheritance applies.

A job only receives a resource if the job's priority is greater than the priority ceiling of the system, or has the same priority and holds the resource cause the current system ceiling.

Pro: Prevents chained block, prevents deadlock

Con: a priori knowledge of resource requirements needed

**Stack-based Priority Ceiling:** The resources required by all jobs are known a priori. The *priority ceiling* of a resource is equal to the highest priority of all jobs that use the resource. The priority ceiling of the system is the highest priority ceiling of all resources **currently in use**.

After a job is released, it is blocked from starting until its assigned priority is higher than the current system ceiling. Resource allocation: Whenever a job requests a resource it receives the resource.

Pro: Prevents chained block, prevents deadlock, once a jobs starts it never blocks

Con: a priori knowledge of resource requirements needed

3. „Ein System bestehe aus den drei unten angegebenen Jobs. Es gebe drei **exklusive** Betriebsmittel **A, B** und **C**. Die Betriebsmittelanforderungen sind pro Job in der folgenden Form gegeben  $J_i: [A;2 [B;3]]$ , wobei  $i$  die **Priorität** von Job **J** bedeutet (kleinerer Index bedeutet höhere Priorität), und **J** das Betriebsmittel **A** **zwei** Zeiteinheiten lang benötigt. Während **J** dieses Betriebsmittel **A** besitzt, benötigt er zusätzlich das Betriebsmittel **B** drei Zeiteinheiten lang.“

“A system contains the three jobs below. There are three **exclusive** resources **A, B, and C**. The resource requirements are given below in the form of  $J_i: [A;2 [B;3]]$ , where  $i$  is the priority of **J** (given  $J_i$  and  $J_j$ , if  $i < j$ ,  $i$  has a higher priority), and **J** requires resource **A** for 2 units of time. While holding resource **A**, **J** requires resource **B** for 3 units of time.”

$J_1: [B ;1]$   
 $J_2: [C ;3 [A ;1]]$   
 $J_3: [B ;1 [A ;1]]$

„Wie groß wird die **maximale Blockiertzeit** für **jeden Job**, wenn zum einen das „**nichtverdrängende Protokoll** für **kritischer Abschnitte**“, zum anderen das „**Priority-Ceiling Protokoll**“ verwendet wird?“

“What is the **maximum blocking time** for **each job** under the “**non-preemptive critical sections protocol**”, and the “**priority ceiling protocol**”?”

**NPCS:**  
 $J_1: 4$   
 $J_2: 2$   
 $J_3: 0$  (2 marks)

**PCP:**  
 $J_1: 2$   
 $J_2: 2$   
 $J_3: 0$  (2 marks)

**Aufgabe/Question 4****(2 + 4 Punkte/marks)**

1. „Einer der Entwurfsparameter beim virtuellem Speicher ist die **Seitengröße**. Zählen Sie **einen Vor-** und **einen Nachteil** auf, wenn Sie sich für eine **große** statt für eine **kleine** Seitengröße entscheiden! Formulieren Sie **präzise und bestimmt!**“

*“One of the parameters of a virtual memory is its **page size**. Give **one advantage** and **one disadvantage** of choosing a **large page size** rather than a **small one**. Be **precise and specific**.”*

- Large pages suffer from internal fragmentation, and they may also contain information which will never be needed in the future.
- + But on the other hand many disk controllers favor larger swap-portions, i.e. it's more efficient to transfer one large page instead of transferring  $n > 1$  small pages. F
- + Less page-table entries  $\Rightarrow$  less TLB entries
- + Small pages don't suffer too much from internal fragmentation, in general only few information of a small page will never be used.
- But on the other hand it costs more time to transfer multiple small pages instead of one large page.
- Need more page-table entries, i.e. more TLB entries could be needed

**(1 mark for each + or -, but in total only 2 marks).**

2. „Oft ist der **tatsächlich genutzte Anteil** einer Applikation an virtuellem Speicher wesentlich kleiner als der **maximale virtuelle Benutzeradressraum**, der vom System zur Verfügung gestellt wird. In diesem Fall könnte das System wünschen, daß es **keine Seitentabelleneinträge** für den nicht genutzten Teil des Benutzeradressraums unterhalten muß. Welche Möglichkeiten bestehen **prinzipiell**, die Zahl der **nicht benötigten Seitentabelleneinträge** für kleine Applikationen zu **limitieren?**“

*“Often, the **actual amount** of virtual memory used by an application task is much less than **the maximum address space** of the **application** that is permitted by the system. In this case, the system may wish to avoid storing **page table entries** for unused pages of the user address space. Which possibilities do exist **in principle** for **limiting the number** of **unused page table entries** for small tasks?”*

Enhance MMU to control the length of a page table, i.e. unused page table entries are no longer needed. **(2 marks)**

Use an inverted page table instead **(2 marks)**

Hierarchical page tables eliminate the need to store never used page-tables. **(2 marks)**

**Guarded page-table is an extra mark.**

3. “Ein winziges System mit virtuellem Speicher habe nur 2 Kacheln (*page frames*) und benutze Seitenersetzung auf Verlangen (*paging on demand*). Beide Kacheln seien zunächst leer. Betrachten Sie den unten angegebenen Seitenreferenzstring. Geben Sie eine untere Schranke für die Zahl der Seitenfehler bei diesem Seitenreferenzstring an!“

*“A tiny system with virtual memory has only 2 page frames of physical memory, and uses demand paging. Both page frames are initially empty. Consider the following page reference string:*

**Seitenreferenzstring/page reference string: 1 2 3 2 3 1 3 1 2 1**

Give a lower bound on the number of page faults that will result from this page reference string.”

<b>frame_no</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>1</b>
<b>0</b>	<b>1</b>	<b>1</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>3</b>	<b>2</b>	<b>2</b>
<b>1</b>		<b>2</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>
<b>#pf</b>	<b>x</b>	<b>x</b>	<b>x</b>			<b>x</b>			<b>x</b>	

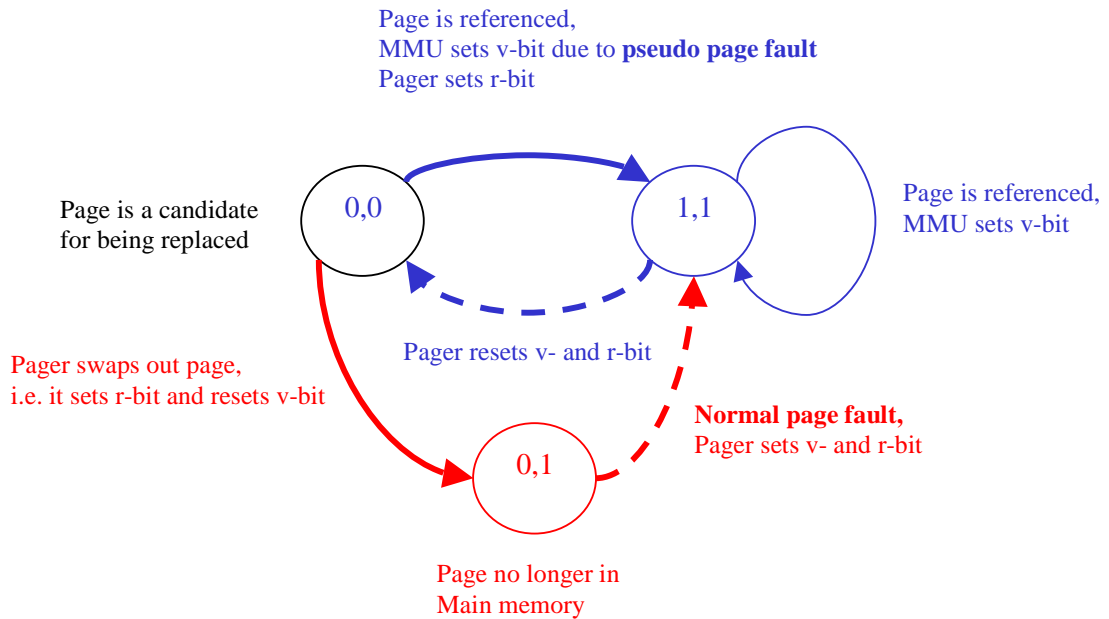
Using Bealdy’s paging algorithm delivers: there are at least **5 page faults**. (2 marks)

4. „Ein virtuelles Speichersystem soll auf einer MMU implementiert werden, die kein Referenzbit (R) in den Seitentabelleneinträgen unterstützt. Mit anderen Worten, die MMU setzt nicht automatisch ein R-Bit, wenn auf die entsprechende Seite zugegriffen wird. Die einzigen von der MMU unterstützten Kontrollbits sind das Modifiziertbit (D = dirty bit) und das Präsenzbit (V = valid bit). Zeigen Sie, wie man ein R-Bit in Software unter den oben gegebenen Hardwarevoraussetzungen effizient simulieren kann. Das simulierte R-Bit soll sich genauso wie ein hardware-unterstütztes R-Bit verhalten. Um dies zu erreichen, kann die Software beliebige Kontrollbits in den Seitentabelleneinträgen definieren und verwalten.“

“A virtual memory system is to be implemented on an MMU that does not support a reference (R) bit in page table entries. In other words, the MMU does not automatically set an R bit each time a page is referenced. The only page table entry bits supported by the MMU are a dirty (D) bit and a valid (V) bit. Show how an R bit can be simulated efficiently in software by the virtual memory system, using no hardware facilities other than the ones described above. This simulated R Bit should appear to behave just like an MMU-supported R bit would. To simulate the R bit, the software may define and use any page table control bits it wishes to, but such bits will not be used by the MMU.”

The hardware-supported dirty-bit is no help at all. The paging software can maintain a reference-bit in each page table entry. And it will use the valid-bit as follows:

Valid	Reference	Meaning
0	0	Page is in memory, but not referenced
0	1	Page is not in memory
<b>1</b>	<b>0</b>	<b>Not used at all</b>
1	1	Page is in memory and referenced



Only if it's absolutely clear how the r-bit is implemented in software **4 marks**.

### Aufgabe 5/ Question 5:

(2 + 5 + 2 + 3 Punkte/marks)

1. „Die **Auftragswarteschlange** des Plattentreibers enthalte Plattenanforderungen (*disk requests*) für Sektoren auf folgenden Zylindern (*cylinders*): **400, 20, 19, 74, 899**. Die aktuelle Plattenarmposition sei Zylinder 200. In welcher Reihenfolge werden obige Plattenaufträge abgearbeitet, wenn als Diskschedulingstrategie a) „**SSFT**“ (Shortest Seek Time First) und b) **Circular-SCAN**“ verwendet werde.“

“Suppose the disk request queue contains disk requests for sectors of the following cylinders: **400, 20, 19, 74, 899**. Suppose that the disk arm is currently positioned at cylinder 200. In which order will the requests be handled under a) **SSTF** (shortest seek time first) and b) **Circular-SCAN** disk head scheduling policy.”

a) SSTF: 74, 20, 19, 400, 899 (1 mark)

b) C-SCAN: 400, 899, 19, 20, 74 or  
74, 20, 19, 899, 400 (1 mark)

2. „Zählen Sie die **wesentlichen Aufgaben** eines **Dateisystems** auf und **erläutern** sie knapp deren **Sinn und Zweck!**“

“Enumerate the *essential functions of a file system* and briefly *explain their purposes*.”

Possibility to store data for a longer period of time (data objects =files)  
to overcome a disconnected system  
Offering a name space to enhance the handling (i.e. retrieving) of files  
Offering directories to ease the usage of files

Managing the secondary storage to establish proper locations for placing files and directories

Offering file protection to restrict file access only to allowed participants

Offering file security to enable data integrity, privacy etc.

Offering recoverability to restore a file-system after a system-crash

Offering file attributes like (type, reference counter, date of last update, etc.) to enable application adequate accessing file entities, enable sharing, etc.

### 1 mark for each task if it is explained

3. „Erläutern Sie, warum in **UNIX-ähnlichen** Systemen „**Hardlinks**“ auf **Verzeichnisse** (*directories*) für **Normalbenutzer nicht erlaubt** sind.“  
 „*Explain why in UNIX-like systems hard links onto directories are not allowed for normal users.*”

You have to avoid that normal users can establish circles within the file system, otherwise a ls-command might never end (**1 mark**)

You also want to avoid any bypass of intermediate more protected directories (**1 mark**)

4. „Zählen Sie **drei verschiedene Methoden** auf, wie sie mehr oder weniger **fälschungssichere Capabilities implementieren** können. **Ordnen** Sie diese drei **Implementierungs-alternativen** gemäß deren **Fälschungssicherheit** (beginnend mit der am wenigsten fälschungssicheren Variante).“  
 “*Enumerate three different methods how to implement more or less unforgeable capabilities. Order these three implementation alternatives according to their degree of unforgeability (starting with the method being least unforgeable).*”

Sparsity

Cryptographic

Hardware supported (i.e. tagged, splitted segments etc.)

**(1 mark for the correct order**

**1 mark and 2\* 0.5 marks for the 3 different implementation variants**