

**8.1**

a)

FIFO

Referenz	0	3	1	2	0	9	3	2	4	3	3	1	2	5	9	3	2	4	1	2
Rahmen 1	0	0	0	0	0	9	9	9	9	9	9	9	2	2	2	2	2	4	4	4
Rahmen 2		3	3	3	3	3	3	3	4	4	4	4	4	5	5	5	5	5	1	1
Rahmen 3			1	1	1	1	1	1	1	3	3	3	3	3	9	9	9	9	9	2
Rahmen 4				2	2	2	2	2	2	2	2	1	1	1	1	3	3	3	3	3
Priorität 1	0	3	1	2	2	9	9	9	4	3	3	1	2	5	9	3	3	4	1	2
Priorität 2		0	3	1	1	2	2	2	9	4	4	3	1	2	5	9	9	3	4	1
Priorität 3			0	3	3	1	1	1	2	9	9	4	3	1	2	5	5	9	3	4
Priorität 4				0	0	3	3	3	1	2	2	9	4	3	1	2	2	5	9	3
Seitenfehler	x	x	x	x		x			x	x		x	x	x	x	x		x	x	x

Anzahl Seitenfehler: 15

LRU

Referenz	0	3	1	2	0	9	3	2	4	3	3	1	2	5	9	3	2	4	1	2
Rahmen 1	0	3	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Rahmen 2		0	3	1	1	1	3	3	3	3	3	3	3	3	9	9	9	9	1	1
Rahmen 3			0	3	3	9	9	9	9	9	9	1	1	1	1	3	3	3	3	3
Rahmen 4				0	0	0	0	0	4	4	4	4	4	5	5	5	5	4	4	4
Priorität 1	0	3	1	2	0	9	3	2	4	3	3	1	2	5	9	3	2	4	1	2
Priorität 2		0	3	1	2	0	9	3	2	4	4	3	1	2	5	9	3	2	4	1
Priorität 3			0	3	1	2	0	9	3	2	2	4	3	1	2	5	9	3	2	4
Priorität 4				0	3	1	2	0	9	9	9	2	4	3	1	2	5	9	3	3
Seitenfehler	x	x	x	x		x	x		x			x		x	x	x		x	x	

Anzahl Seitenfehler: 13

SC (Hierbei steht i in  $X_i$  für das gesetzte Access-Bit der Page X)

Referenz	0	3	1	2	0	9	3	2	4	3	3	1	2	5	9	3	2	4	1	2
Rahmen 1	0	0	0	0	0	0	0	0	4	4	4	4	4	5	5	5	5	4	4	4
Rahmen 2		3	3	3	3	9	9	9	9	9	9	1	1	1	9	9	9	9	1	1
Rahmen 3			1	1	1	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Rahmen 4				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Priorität 1	0	3	1	2	2	9	3	3	4	4	4	1	1	5	9	9	9	4	1	1
Priorität 2		0	3	1	1	0	9	9	2	2	2	4	4	2	5	5	5	2	4	4
Priorität 3			0	3	3	2	0	0	3	$3_i$	$3_i$	2	$2_i$	3	2	2	$2_i$	3	2	$2_i$
Priorität 4				0	$0_i$	1	2	$2_i$	9	9	9	$3_i$	$3_i$	1	3	$3_i$	$3_i$	9	3	3
Seitenfehler	x	x	x	x		x	x		x			x		x	x			x	x	

Anzahl Seitenfehler: 12

LFU (Hierbei steht i in  $X_i$  für die Anzahl der Zugriffe auf die Page X seit seinem Laden)

Referenz	0	3	1	2	0	9	3	2	4	3	3	1	2	5	9	3	2	4	1	2
Rahmen 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Rahmen 2		3	3	3	3	9	9	9	4	4	4	1	1	5	9	9	9	4	1	1
Rahmen 3			1	1	1	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Rahmen 4				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Priorität 1	$0_1$	$3_1$	$1_1$	$2_1$	$2_1$	$9_1$	$3_1$	$3_1$	$4_1$	$4_1$	$4_1$	$1_1$	$1_1$	$5_1$	$9_1$	$9_1$	$9_1$	$4_1$	$1_1$	$1_1$
Priorität 2		$0_1$	$3_1$	$1_1$	$1_1$	$2_1$	$9_1$	$9_1$	$3_1$	$3_2$	$3_3$	$3_3$	$3_3$	$3_3$	$3_3$	$3_4$	$3_4$	$3_4$	$3_4$	$3_4$
Priorität 3			$0_1$	$3_1$	$3_1$	$1_1$	$2_1$	$2_2$	$2_2$	$2_2$	$2_2$	$2_2$	$2_3$	$2_3$	$2_3$	$2_3$	$2_4$	$2_4$	$2_4$	$2_5$
Priorität 4				$0_1$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$	$0_2$
Seitenfehler	x	x	x	x		x	x		x			x		x	x			x	x	

Anzahl Seitenfehler: 12

CLIMB

Referenz	0	3	1	2	0	9	3	2	4	3	3	1	2	5	9	3	2	4	1	2
Rahmen 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Rahmen 2		3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Rahmen 3			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Rahmen 4				2	2	9	9	2	4	4	4	4	2	5	9	9	2	4	4	2
Priorität 1	0	0	0	0	0	0	3	3	3	3	3	3	3	3	3	3	3	3	1	1
Priorität 2		3	3	3	3	3	0	0	0	0	0	1	1	1	1	1	1	1	3	3
Priorität 3			1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
Priorität 4				2	2	9	9	2	4	4	4	4	2	5	9	9	2	4	4	2
Seitenfehler	x	x	x	x		x		x	x				x	x	x		x	x		x

Anzahl Seitenfehler: 13

## OPT

Referenz	0	3	1	2	0	9	3	2	4	3	3	1	2	5	9	3	2	4	1	2
Rahmen 1	0	0	0	0	0	9	9	9	4	4	4	4	4	4	4	4	4	4	1	1
Rahmen 2		3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Rahmen 3			1	1	1	1	1	1	1	1	1	1	1	5	9	9	9	9	9	9
Rahmen 4				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Priorität 1	0	0	0	0	3	3	2	3	3	3	1	2	3	3	3	2	4	2	2	9
Priorität 2		3	3	3	2	2	3	1	1	1	2	3	2	2	2	4	2	9	9	3
Priorität 3			1	2	1	1	1	2	2	2	3	4	4	4	4	9	9	3	3	1
Priorität 4				1	0	9	9	9	4	4	4	1	1	5	9	3	3	4	1	2
Seitenfehler	x	x	x	x		x			x					x	x					

Anzahl Seitenfehler: 8

(Angenommen, dass eine Page, die nicht mehr gebraucht wird, unter ältere sortiert wird, die ebenfalls nicht mehr gebraucht werden.)

### b)

Ein entstehendes Problem dabei ist, dass so Seiten, welche zum Beispiel am Anfang einige Zugriffe hatte, unter Umständen nicht mehr verdrängt wird, falls ständig neue Pages geladen und wieder verdrängt werden, obwohl die alte Seite mit anfänglich vielen Zugriffen gar nicht mehr benutzt wird, wodurch sie effektiv anderen Seiten den Platz wegnimmt.

## 8.2

### FIFO

Es wird Seite 2 verdrängt, da diese sich am längsten im Hauptspeicher befindet im Vergleich zu den restlichen Seiten. (Zeitpunkt 120 kleiner als alle Anderen in der ersten Spalte.)

### LRU

Es wird Seite 4 verdrängt, da diese am längsten nicht mehr benutzt wurde im Vergleich zu den restlichen Seiten. (Zeitpunkt 255 kleiner als alle Anderen in der zweiten Spalte.)

### Second Chance

Es wird Seite 1 gelöscht, da dies die älteste Seite mit nicht gesetztem A-Bit ist. (Seiten 0 und 2 sind zwar älter, haben aber ein gesetztes A-Bit.)

## NRU

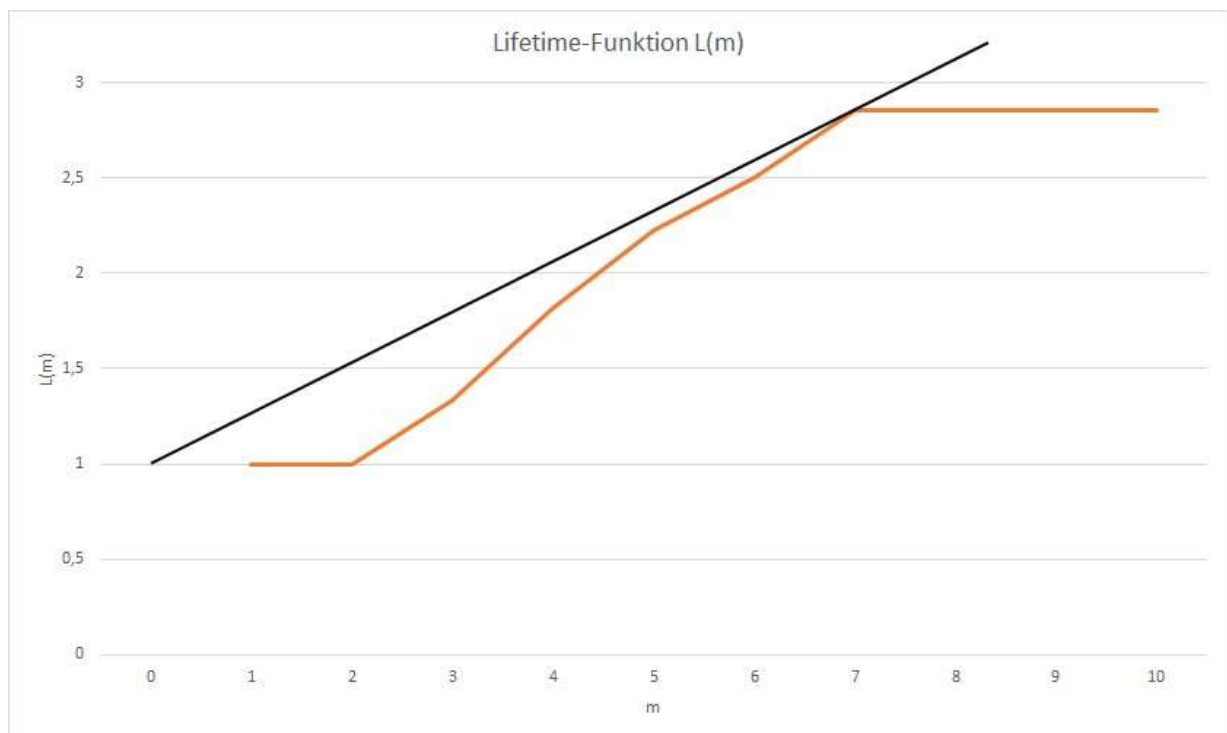
Es wird Seite 3 verdrängt, da sich Seite 3 als einzige Seite mit  $A=0$  und  $D=0$  in der niedrigsten NRU-Klasse befindet, und somit Klasse 1 nicht leer ist und bei der Verdrängung priorisiert wird.

### 8.3

a)

Rahmenzahl $m$	1	2	3	4	5	6	7	8	9	10
Seitenfehler	20	20	15	11	9	8	7	7	7	7
Mittlere Zeit zwischen Seitenfehlern	1,000	1,000	1,333	1,818	2,222	2,500	2,857	2,857	2,857	2,857

(Zeit zwischen Seitenfehlern in "Zeitschritten")



(orange:  $L(m)$ , schwarz: Tangente an das primäre Knie  $m_{\text{opt}}=7$ )

b)

Die optimale Speichergröße (=Anzahl der Rahmen)  $m_{\text{opt}}$  liegt bei  $m=7$ , da aber dieser Rahmenanzahl  $L(m)$  nicht mehr steigt.

## **8.4**

### **a)**

Unabhängig vom 64-bit System wird aufgrund der Definition der I-Nodes ein Datenblock mit 4 Byte adressiert, was bei einer Blockgröße von 1024 Byte 256 Blockadressen pro Indirektionsstufe einspricht.

Die Gesamtzahl an möglichen Datenblöcken ergibt sich also durch 12 (direkte Datenpointer) + 256 (einfache Indirektion) +  $256^2$  (doppelte Indirektion) +  $256^3$  (dreifach) = 16843020.

da jeder Block 1024 Byte groß ist, kann also eine Datei maximal  $16843020 \cdot 1024 = 17247252480$  Bytes  $\approx 16$  GiB groß sein.

### **b)**

Zuerst wird der Directory-Block des root-Verzeichnisses aus dem "First I-Node" im Superblock ausgelesen. In diesem steht ein Zeiger auf den I-Node vom Unterverzeichnis /home/. Der Typ dieses I-Nodes ist "dir" und enthält im direct 0-Feld einen Zeiger auf den Directory-Block von /home/. Man geht immer abwechselnd von I-Node zu Directory-Block, bis man den Directory-Block des gewünschten Verzeichnisses erreicht hat. Im Beispiel ergibt sich also:

First I-Node -> Directory-Block "root" -> I-Node "home" -> Directory-Block "home" -> I-Node "BuS" -> Directory-Block "BuS" -> I-Node "test" -> Directory-Block "test"