




11.07.2021

LÖSUNG

MODUL:
BETRIEBSSYSTEME UND DIGITALE SPUREN I
ÜBUNGEN ZU BETRIEBSSYSTEMEN UND ZUR BETRIEBSSYSTEMARCHITEKTUR

PROF. BODACH
HOCHSCHULE MITTWEIDA



BS I

Übungen zur Betriebssystemarchitektur

Binäre Zahlenarithmetik und Zeichenkodierung

1. Bilden Sie aus den Dezimal- und Hexadezimalzahlen jeweils das Einerkomplement und das Zweierkomplement in Binär Schreibweise und in Hexadezimal bzw. Dezimal Schreibweise.

Zahl 1: -56

Zahl 2: 0xF9

Lösung 1:

Binärcode ohne VZ:	00111000	Hexcode ohne VZ:	0x38
Einerkomplement:	11000111	Hexcode Einerkomplement:	0xC7
Zweierkomplement:	11001000	Hexcode Zweierkomplement:	0xC8

Lösung 2:

Binärcode:	11111001	Dezimal:	-7
Einerkomplement:	00000110	Dezimal Einerkomplement:	6
Zweierkomplement:	00000111	Dezimal Zweierkomplement:	7

2. In der heutigen Zeit gibt es viele standardisierte Zeichensätze. Eines davon ist der ASCII – American Standard Code for Information Interchange. Dieser 7-Bit lange Code codiert Zeichen und Steuerzeichen im englischen Alphabet.

Jeder Buchstabe wird mit 7 Bits codiert und hat folgenden Aufbau: b7 b6 b5 b4 b3 b2 b1

Um ein Zeichen in ASCII zu codieren, schauen Sie zuerst oben wie sie b7, b6, b5 codieren (Spaltenauswahl). Danach schauen Sie links in welcher Zeile das das Zeichen steht. Mit dieser Zeile codieren Sie b4, b3, b2, b1.

Bsp: b7 b6 b5 b4 b3 b2 b1

H → 1 0 0 1 0 0 0

S → 1 0 1 0 0 1 1

ASCII-Code: 10010001010011

Hex Code = 0x4853

BS I

Übungen zur Betriebssystemarchitektur

		0	0	0	0	1	1	1	1	1	1
		0	0	1	0	1	0	0	1	0	1
		0	1	2	3	4	5	6	7		
BITS	b ₇ b ₆ b ₅ b ₄ b ₃ b ₂ b ₁										
	0 0 0 0	0	NUL	DLE	SP	0	@	P	.		p
	0 0 0 1	1	SOH	DC1	!	1	A	Q	a		q
	0 0 1 0	2	STX	DC2	"	2	B	R	b		r
	0 0 1 1	3	ETX	DC3	#	3	C	S	c		s
	0 1 0 0	4	EOT	DC4	\$	4	D	T	d		t
	0 1 0 1	5	ENQ	NAK	%	5	E	U	e		u
	0 1 1 0	6	ACK	SYN	&	6	F	V	f		v
	0 1 1 1	7	BEL	ETB	'	7	G	W	g		w
	1 0 0 0	8	BS	CAN	(8	H	X	h		x
	1 0 0 1	9	HT	EM)	9	I	Y	i		y
	1 0 1 0	10	LF	SUB	*	:	J	Z	j		z
	1 0 1 1	11	VT	ESC	+	;	K	[k		{
	1 1 0 0	12	FF	FS	,	<	L	\	l		
	1 1 0 1	13	CR	GS	-	=	M]	m		}
	1 1 1 0	14	SO	RS	.	>	N	^	n		~
	1 1 1 1	15	SI	US	/	?	O	_	o		DEL

a) Erstellen Sie ein ASCII-Code für: HSMW

b) Welcher Text verbirgt sich hinter: 0x 46 6F 72 65 6E 73 69 63 20 69 73 74 20 63 6F 6F 6C

Lösung a:

01001000 01010011 01001101 01010111 = 0 x 48 53 4D 57

Lösung b:

Forensic ist cool

3. Wie viele unterschiedliche Zeichen kann man mit 7 Bit codieren?

Lösung: $2^7 = 128$ Zeichen

4. Wie viele unterschiedliche Zeichen kann man mit 8 Bit codieren?

Lösung: $2^8 = 256$ Zeichen

5. Wie viele unterschiedliche Zeichen kann man mit 16 Bit codieren?

Lösung: $2^{16} = 65.536$ Zeichen

BS I

Übungen zur Betriebssystemarchitektur

6. Folgende zwei Szenarien sind gegeben:

- a) Ein Textdokument, welches UTF-8 codiert ist, wird mit einem ASCII-Editor geöffnet
- b) Ein Textdokument, welches ASCII codiert ist, wird mit einem UTF-8-Editor geöffnet

In welchem Szenario werden Fehler auftreten? Begründen Sie.

Lösung:

Im Szenario a) werden Fehler auftreten, weil der ASCII-Editor jedes Byte als separates Zeichen angesehen wird. In der UTF-8-Codierung kann ein Unicode-Zeichen aus mehr als einem Byte bestehen.

Im Szenario b) werden keine Fehler auftreten, weil UTF-8 abwärtskompatibel ist. Außerdem ist jedes ASCII-Zeichen im UTF-8-Code derselbe.

BS I

Übungen zur Betriebssystemarchitektur

Betriebssystem und Betriebssystemarchitektur

7. *Nennen Sie fünf Betriebssystemkategorien!*

Lösung:

- Betriebssysteme für Großrechner
- Betriebssysteme für Server
- Betriebssysteme für Laptops und Personal Computer
- Echtzeitbetriebssysteme
- Betriebssysteme für Embedded Systems
- Betriebssysteme für Chipkarten

8. *Was ist ein Von-Neumann-Rechner und wie unterscheidet er sich von einem Harvard-Rechner?*

Ein Von-Neumann-Rechner besteht aus den vier Funktionseinheiten Rechenwerk und Steuerwerk, Speicher (Memory) sowie Ein-/Ausgabe (Input/Output, E/A). Das Steuerwerk holt die Maschinenbefehle nacheinander in den Speicher und führt sie aus. Damit stellt das Steuerwerk den „Befehlsprozessor“ dar. Das Rechenwerk, auch Arithmetisch-Logische Einheit (ALU) genannt, stellt den „Datenprozessor“ dar und führt logische und arithmetische Operationen aus. Im Speicher liegen die Maschinenbefehle und die zu verarbeitenden Daten. Maschinenbefehle und Daten sind also in einem gemeinsamen Speicher. Rechenwerk und Steuerwerk werden heute in der Regel in einem Prozessor, der als Zentraleinheit (CPU, Central Processing Unit) bezeichnet wird, zusammengefasst.

Der Harvard-Rechner, der nach der Struktur des Mark-I-Rechners benannt ist, hat im Unterschied zum Von-Neumann-Rechner zwei getrennte Speicher, einen für die Daten und einen für die Maschinenbefehle. Beide Speicher werden auch über einen getrennten Bus mit der CPU verbunden.

9. *Nennen Sie vier Betriebsmittel, welche das Betriebssystem verwaltet! Welche davon sind hardware- und welche softwaretechnische (virtuelle) Betriebsmittel?*

Betriebsmittel sind u.a.:

- Speicher (Hardware)
- Prozessor (Hardware)
- E/A Geräte (Hardware)
- Dateien (Software)
- Gerätedateien (Software)
- Speichersegmente (Software)

10. *Nennen Sie einen Vorteil der Schichtenarchitektur bei Betriebssystemen!*

Die Abhängigkeiten von der Hardware sind in einer Schicht gekapselt, was eine Portierung auf eine andere Hardwareplattform erleichtert.

BS I

Übungen zur Betriebssystemarchitektur

11. Was versteht man unter einem Mikrokern?

Im Gegensatz zu monolithischen Architekturen wird eine Betriebssystemarchitektur, die einen reduzierten - leichtgewichtigen Kernel enthält, als Mikrokern-Architektur bezeichnet. Der Kernel wird bei dieser Architekturvariante dadurch entlastet, dass Funktionalität in Anwendungsprozesse, sog. Systemprozesse, ausgelagert wird. Der Kernel übernimmt hier im Wesentlichen die Abwicklung der Kommunikation zwischen Client- und Systemprozessen. Clientprozesse greifen bei dieser Betriebssystemarchitektur auf den Mikrokern über eine Systemschnittstelle zu. Der Mikrokern leitet die Requests an die entsprechenden Systemprozesse, die in Anwendungsprozessen ablaufen, weiter und stellt umgekehrt den Clientprozessen die Ergebnisse zu.

Mikrokern enthalten nur noch elementare Funktionen, z.B. zur Speicherverwaltung, IPC, Prozessverwaltung, Scheduling-Mechanismus, sowie einige hardwarenahe E/A-Funktionen. Systemprozesse realisieren dagegen den Rest der Verwaltung wie z.B. die Dateiverwaltung.

12. Was klassifiziert einen Makrokern?

Ein Makrokern wird auch als Hybridkernel bezeichnet und stellt einen Kompromiss zwischen einem Mikrokern und einem monolithischen Kernel dar, bei dem aus Performancegründen Teile des monolithischen Kernel in den Kern integriert werden.

Vorteile liegen hier in der geringeren Fehler Anfälligkeit wie dies bei monolithischen Kernel der Fall ist, da nicht alle Treiber im privilegierten Modus laufen und bei einem Absturz das ganze System zum Absturz bringen. Zudem werden höhere Geschwindigkeiten der Abarbeitung erreicht, da nicht so viele Kontextwechsel und Kommunikation nötig werden wie bei einem Mikrokern.

13. Nennen Sie den Unterschied zwischen Multiprozessor und Multicore Systemen.

Multiprozessorsysteme enthalten mehrere Prozessoren, die sich auf verschiedenen Chips befinden. Diese Chips bzw. CPU Prozessoreinheiten besitzen jeweils einen eigenen Cache und MMU Bereich.

Multicore-Prozessoren werden von einer „Familie“ von Prozessoren gebildet, die mehrere Prozessoren, z. B. 2, 4 oder 8, auf einem Chip integrieren. Sie besitzen jeweils nur einen gemeinsam genutzten Cache und MMU Bereich.

BS I

Übungen zur Betriebssystemarchitektur

Prozesse, Tasks und Threads

14. Was ist ein Prozess?

Ein Prozess ist ein in Ausführung befindliches (nicht zwingend aktives) Programm (Maschinenprogramm, Anwendung) und seine aktuell genutzten Daten. Es handelt sich dabei um eine konkrete Ausführungsumgebung für ein Programm, mit den dazu notwendigen Betriebsmittel: Speicher, Rechte, Verwaltungsinformation (verbrauchte Rechenzeit,...).

15. Was versteht man unter Multitasking/Multiprogramming?

Im Gegensatz zum Singleprogrammbetrieb (Batch Processing) eröffnet Multitasking die Möglichkeit der „gleichzeitigen“ oder aber auch „quasi-gleichzeitigen“ Ausführung von Programmen in einem Betriebssystem.

16. Benötigt man für Multitasking/ Multiprogramming mehrere CPUs?

Nein, nicht unbedingt. Multiprogramming kann auch über eine CPU realisiert werden, die nach einer vorgegebenen Strategie des Betriebssystems (Scheduling und Dispatching) auf die nebenläufigen Programme (Prozesse) aufgeteilt wird.

17. Was bezeichnet man als Timesharing?

Die Zuordnung der Zeitanteile der Rechenzeit eines Prozessors an die nebenläufigen Programme bzw. Prozesse nach Zeitintervallen wird als Timesharing bezeichnet.

18. Was ist in der Prozessverwaltung ein PCB, wozu dient er und welche Inhalte hat er? Nennen Sie dabei drei wichtige Informationen, die im PCB verwaltet werden!

Ein Eintrag in der Prozesstabelle heißt Process Control Block (PCB). Je nach Betriebssystem gibt es deutliche Unterschiede im Aufbau. Einige Informationen sind aber prinzipiell sehr ähnlich. Hierzu gehört u.a die Information zur Identifikation des Prozesses, die Information zum aktuellen Prozesszustand sowie Informationen zu sonstigen Ressourcen, die dem Prozess zugeordnet sind (Dateien, offene Netzwerkverbindungen).

Weitere Informationen, die für einen Prozess im PCB verwaltet werden, sind der Programmzähler, der Prozesszustand, die Priorität und die Prozessnummer sowie die aktuellen Registerinhalte.

BS I

Übungen zur Betriebssystemarchitektur

19. *Threads werden heute von den meisten Betriebssystemen unterstützt. Was versteht man unter einem Thread in Hinblick auf Prozesse?*

Ein Thread stellt eine nebenläufige Ausführungseinheit innerhalb eines Prozesses dar.

Threads werden Charakterisiert durch:

- gemeinsame Nutzung von Ressourcen wie Instruktionen, Datenbereiche und E/A Ressourcen
- separate Nutzung von Programmzählern, Registersätzen und Stackbereichen

20. *Was versteht man unter User-Level-Threads im Vergleich zu Kernel-Level-Threads und welche Beziehungen zwischen beiden sind möglich?*

User-Level-Threads laufen auf der Benutzerebene (Benutzermodus), Kernel-Level-Threads dagegen auf der Kernelebene (Kernelmodus, Systemmodus) ab.

21. *Kann es unter Windows sein, dass ein Thread mehreren Prozessen zugeordnet ist? Begründen Sie Ihre Entscheidung!*

Nein, ein Thread wird innerhalb eines Prozesses erzeugt und ist diesem bis zur Terminierung zugeordnet.

22. *Warum ist der Einsatz von Threads sinnvoll?*

Threads sind leichtgewichtiger als Prozesse. Ein Thread-Kontextwechsel innerhalb des laufenden Prozesses kann schneller sein als ein Prozess-Kontextwechsel, weil meist Speicherbereiche des gleichen Prozesses verwendet werden. Ein Austausch von Speicherbereichen ist daher oft nicht erforderlich, was den Betriebssystem-Overhead reduziert.

23. *Welche Aufgaben haben im Prozess-Management der Dispatcher und der Scheduler?*

Die Komponente im Prozessmanager, die für die Planung der Betriebsmittelzuteilung zuständig ist, heißt Scheduler. Die Komponente, die dann einen tatsächlichen Prozesswechsel ausführt, wird als Dispatcher bezeichnet.

24. *Welche Scheduling Strategien kenne Sie die in Betriebssystemen zu Anwendung kommen können?*

Lösung:

- First Come First Served
- Shortest Job First
- Prioritäts-Scheduling (non präemptiv)
- Round Robin
- Prioritäts-Scheduling (präemptiv) mit Multilevel-Queue Scheduling oder Multilevel-Feedback-Queue Scheduling

BS I

Übungen zur Betriebssystemarchitektur

25. *Erläutern Sie den Unterschied zwischen präemptiven und non-präemptive Scheduling und nennen Sie jeweils zwei Scheduling-Strategien, die in diese Kategorien passen.*

Im non-präemptive, auch „run-to-completion“-Verfahren genannt, darf ein Prozess nicht unterbrochen werden, bis er seine Aufgaben vollständig erledigt hat. Beispiele: First Come First Serve (FCFS) und Shortest Job First (SJF).

Im Gegensatz dazu darf im präemptive Scheduling eine Unterbrechung stattfinden. Rechenbereite Prozesse können somit suspendiert werden. Dies setzt natürlich eine Strategie zur Vergabe der CPU voraus, die vom Betriebssystem unterstützt werden muss und in der Regel auf Time Sharing basiert. Dieses Verfahren ist für die Unterstützung konkurrierender Benutzer geeignet. Beispiele: Round Robin (RR) und Priority Scheduling (PS).

BS I

Übungen zur Betriebssystemarchitektur

Synchronisation und Parallelität

26. Was bezeichnet man in der Prozessverwaltung als Blockieren, Verklemmen und Verhungern?

- *Blockieren*: Ein Prozess P1 belegt Betriebsmittel, ein zweiter Prozess P2 benötigt dasselbe Betriebsmittel ebenfalls und wird daher blockiert, bis es P1 freigegeben hat.
- *Verhungern (Starvation)*: Ein Prozess erhält trotz Rechenbereitschaft keine CPU-Zeit zugeteilt, z.B. weil ihm immer wieder Prozesse mit höherer Priorität vorgezogen werden.
- *Verklemmung*: Zwei oder mehrere Prozesse halten jeder für sich ein oder mehrere Betriebsmittel belegt und versuchen ein weiteres zu belegen, das aber von einem anderen Prozess belegt ist. Es liegt ein Zyklus von Abhängigkeiten vor. Kein Prozess gibt seine Betriebsmittel frei und alle Prozesse warten daher ewig. Dieser Zustand wird auch als Deadlock bezeichnet.

27. Bei Betriebssystemen, bei systemnahen Programmen, aber auch bei Anwendungssoftware muss man sich als Entwickler von nebenläufig auszuführenden Aktionen mit kritischen Abschnitten befassen. Was versteht man unter einem kritischen Abschnitt?

Ein kritischer Abschnitt ist ein Codeabschnitt, der zu einer Zeit nur durch einen Prozess bzw. Thread durchlaufen und in dieser Zeit nicht durch andere nebenläufige Prozesse bzw. Threads betreten werden darf. Ein Prozess bzw. Thread, der einen kritischen Abschnitt betritt, darf nicht unterbrochen werden. Sofern das Betriebssystem in dieser Zeit aufgrund einer Scheduling-Entscheidung eine Unterbrechung zulässt, darf der Prozess bzw. Thread, der den kritischen Abschnitt belegt, durch andere Prozesse, die die CPU erhalten, nicht beeinflusst werden.

28. Welche Maßnahmen sind zu treffen, damit es beim Durchlaufen eines kritischen Abschnitts nicht zu Inkonsistenzen kommt? Gehen Sie dabei auf den Begriff des gegenseitigen Ausschlusses (*mutual exclusion*) ein!

Um Inkonsistenzen zu vermeiden, muss ein kritischer Abschnitt geschützt werden. Dies kann durch gegenseitigen (oder wechselseitigen) Ausschluss (engl.: *mutual exclusion*) erreicht werden. Prozesse, die einen kritischen Abschnitt ausführen wollen, müssen warten, bis dieser frei ist. Mit einem wechselseitigen Ausschluss wird also die Illusion einer atomaren Anweisungsfolge geschaffen, denn echt atomar wird sie natürlich nicht ausgeführt. Es kann ja immer noch vorkommen, dass ein nebenläufiger Prozess zwischendurch die CPU erhält.

BS I

Übungen zur Betriebssystemarchitektur

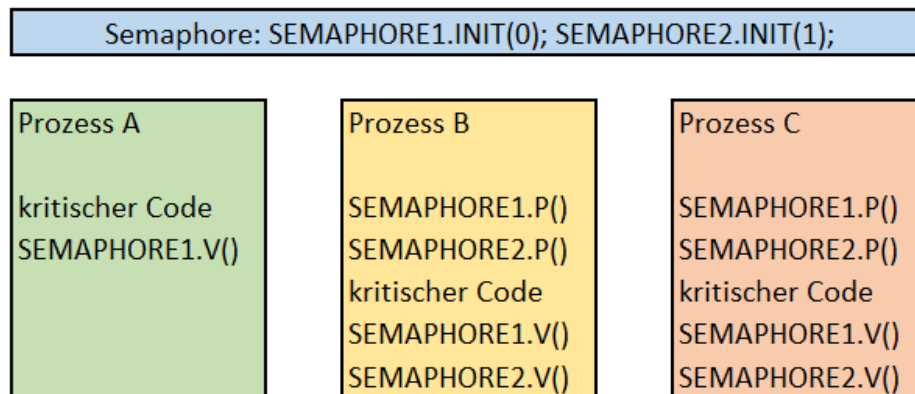
29. Was sind Semaphore? Gehen Sie dabei kurz auf die Semaphoreoperationen P() und V() ein!

Das Semaphore-Konzept ist ein Konzept zur Lösung des Mutual-Exclusion-Problems auf Basis von Sperren. Ein Semaphore verwaltet intern eine Warteschlange für die Prozesse bzw. Threads, die gerade am Eingang eines kritischen Abschnitts warten müssen, und einen Semaphorezähler. Es kommt auf die Initialisierung des Semaphorezählers an, wie viele Prozesse in den kritischen Abschnitt dürfen.

Für den Eintritt in den bzw. Austritt aus dem kritischen Abschnitt gibt es zwei Operationen:

- P() wird beim Eintritt in den kritischen Abschnitt aufgerufen. Der Semaphorezähler wird um 1 reduziert, sofern er größer als 0 ist. Wenn er gerade auf 0 steht, wird der Eintritt verwehrt, der Prozess/Thread wird in die Warteschlange eingereiht und suspendiert.
- V() wird beim Verlassen des kritischen Abschnitts aufgerufen. Der Semaphorezähler wird wieder um 1 erhöht, so dass ein weiterer Prozess/ Thread in den kritischen Abschnitt darf.

30. Gegeben seien Prozesse A, B und C, die über Semaphore synchronisiert werden.



Aufgabenbeschreibung:

- SEMAPHORE1 und SEMAPHORE 2 stellen die beiden Semaphore dar und werden anfangs mit 0 und 1 initialisiert.
- in den Vorlesungsfolien können Sie die Funktionsweise der Semaphore mit den V und P Operatoren nachschauen
- in den in den Vorlesungsfolien aufgeführten Szenarien können Sie die unterschiedlichen Problemstellungen die mit Semaphore gelöst werden können nachvollziehen

Die Aufgabenstellung vereint verschiedene Semaphore Szenarien, welche sind dies und wie ist die Funktionsweise im Beispiel?

- Welche Synchronisationsbedingung (Signalisierungsfunktion oder wechselseitiger Ausschluss) wird bei SEMAPHORE1 und SEMAPHORE2 durchgesetzt?*
- Geben Sie die Prozessreihenfolge an, die durch die Semaphore Operationen durchgesetzt wird. Begründen Sie.*

BS I

Übungen zur Betriebssystemarchitektur

Lösung a):

- *SEMAPHORE1*: Signalisierungsfunktion
- *SEMAPHORE2*: wechselseitiger Ausschluss

Lösung b):

- Prozess A wird zuerst ausgeführt, weil *SEMAPHORE1* dies ermöglicht.
- dann werden Prozess B und C im wechselseitigen Ausschluss ausgeführt ungeachtet einer Reihenfolge

BS I

Übungen zur Betriebssystemarchitektur

Speicherverwaltung

31. Welche grundlegenden Gedanken stecken hinter dem Konzept des virtuellen Speichers?

Folgende Gedanken sind grundlegend für die virtuelle Speichertechnik:

- Ein Prozess sollte auch dann ablaufen können, wenn er nur teilweise im Hauptspeicher ist. Wichtig ist hierbei, dass die Teile des Prozesses (Daten und Code) im physikalischen Speicher sind, die gerade benötigt werden.
- Der Speicherbedarf eines Programms sollte größer als der physikalisch vorhandene Hauptspeicher sein können.
- Ein Programmierer sollte am besten nur einen kontinuierlichen (linearen) Speicherbereich beginnend bei Adresse 0 sehen und sich nicht um die Zerstückelung (Fragmentierung) des Hauptspeichers auf mehrere Benutzer kümmern.
-

32. Gegeben sei ein Swapping-Speicher, welcher folgende Speicherbereiche in dieser Reihenfolge enthält: 10, 5, 20, 15, 6, 9, 12, 18, 7 Kilobyte. Die Speicherplatzanforderungen sind sukzessiv und sehen wie folgt aus: 15, 10, 9 Kilobyte.

Welche Speicherbereiche werden mit den jeweiligen Belegungsstrategien ausgewählt?

Schraffieren Sie in der Tabelle alle vollen Speicherbereiche. Wenn ein Rest übrigbleibt, ist dieser in der Zelle bzw. in der nächsten Speicherbelegung zu schreiben. Die Belegungsstrategien sind Best-Fit, First-Fit und Next-Fit.

Strategie	Speicherbereiche								
Aufteilung bevor	10	5	20	15	6	9	12	18	7
Best-Fit									
First-Fit									
Next-Fit									

Lösung:

Strategie	Speicherbereiche								
Aufteilung bevor	10	5	20	15	6	9	12	18	7
Best-Fit									
First-Fit			5	6					
Next-Fit			5	5					

BS I

Übungen zur Betriebssystemarchitektur

33. *Wie viele Seitentabellen müssen bei virtueller Adressierung vom Betriebssystem verwaltet werden, wenn mit einer einstufigen Seitentabelle gearbeitet wird und gerade 10 Prozesse aktiv sind?*

Es müssen 10 Seitentabellen verwaltet werden, eine je Prozess.

34. *Was ist ein Seitenfehler (Page Fault)?*

Ein Prozess möchte den Speicherbereich der Seitentabelle aufrufen der nicht im Hauptspeicher eingelagert ist. Dadurch wird durch den Aufruf ein Seitenfehler (Page Fault) generiert der als Interrupt die weitere Prozessbearbeitung unterbricht und die Seiten Einlagerung in den Hauptspeicher durchführt.

35. *Ein virtueller Adressraum wird mit 32 Bit langen virtuellen Adressen adressiert. Eine virtuelle Adresse enthält jeweils 10 Bit für den Index in der Haupt-Seitentabelle und 10 Bit für den Index in der Unter-Seitentabelle.*

- a) *Wie viele Unter-Seitentabellen gibt es maximal je Prozess?*
- b) *Wie groß sind die Seitenrahmen im Hauptspeicher?*
- c) *Wie groß ist der gesamte virtuelle Adressraum eines Prozesses?*

Lösung a:

$2^{10} = 1024$ je Prozess, da 10 Bit für den Index der Hauptseiten-Tabelle zur Verfügung stehen. Die 1024 Einträge verweisen auf 1024 Unterseiten-Tabellen.

Lösung b:

2^{12} Byte = 4096 Byte, da 12 Bit für die Größe einer virtuellen Seite zur Verfügung stehen und die Größe einer physikalischen Seite (Frame) der Größe der virtuellen Seite entspricht.

Lösung c:

2^{32} Byte = 4 GB, da insgesamt 32 Bit für die Adressierung zur Verfügung stehen.

BS I

Übungen zur Betriebssystemarchitektur

36. Folgende virtuelle 32 Bit Speicheradresse wurde festgestellt:

0x14020E7

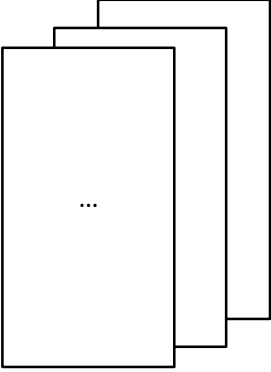
diese besteht aus 12 Bit Offset Wert

und jeweils gleich großen Seitentableneinträgen

Zudem sind folgende Informationen über die Hauptseitentabelle wie folgt erfasst:

0	3
1	4
2	7
3	5
4	2
5	1
...	
1.023	

Zudem wurden folgende Unterseitentabellen erfasst:

	1	2	3	
0	1	3	18	
1	2	5	19	
2	7	10	17	
3	4	12	16	
4	9	6	13	
5	8	14	15	
...				
1.023				

Berechnen Sie die Adresse im Hauptspeicher!

Lösung:

- 0 x 14020E7 ist in binär wie folgt aufzuteilen:

0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 0 0 1 1 1

- Hauptseitentabellenoffset: $000000101_2 = 5$
- Unterseitentabellenoffset: $000000010_2 = 2$
- Offset: $00011100111_2 = 231$
- Hauptseitenrahmen = $2^{12} = 4096$

$$\text{Adresse im Hauptspeicher} = 7 * 4.096 + 231 = 28.903$$

BS I

Übungen zur Betriebssystemarchitektur

37. Welche Möglichkeiten gibt es, die hohe Belastung der Seitentabellenverwaltung insbesondere bei großen Adressräumen zu optimieren?

Zur Optimierung der Speicherzugriffe gibt es Adressumsetzpuffer (Translation Lookaside Buffers, kurz: TLB).

38. Was versteht man unter einem Shared Memory? Nennen Sie Einsatzmöglichkeiten!

Ein Shared Memory ist ein von mehreren Prozessen gemeinsam genutzter Speicherbereich, der nur einmal in den Hauptspeicher geladen wird. Auf gemeinsam genutzte Speicherbereiche verweisen mehrere Seitentabelleneinträge verschiedener Prozesse.

Shared Memory kann genutzt werden, um Codeteile, die mehrere Prozesse benötigen, nur einmal in den Hauptspeicher zu laden. Hier spricht man von Shared Libraries. Weiterhin kann man Shared Memory verwenden, um mehreren Prozessen globale Datenbereiche zur Verfügung zu stellen. Mit Hilfe dieser Speicherbereiche kann dann eine Art Prozesskommunikation stattfinden (Beispiel: Caches von Datenbankmanagementsystemen, auf die mehrere Datenbank-Benutzerprozesse nebenläufig zugreifen).