Grundlagen der Systemsoftware

Übungsblatt 6 · Gruppe G07-A · Krabbe, Flunder, Epplee, Mönnich, Gresens · SoSe 2014

Disclaimer: Leider hatten wir diese Woche nicht besonders viel Zeit und haben daher nur einen kleinen Teil der Aufgaben fertig bekommen. Sorry!

Aufgabe 1: Speicherverwaltung

Aufgabe 1.1:

Bei einer virtuellen Adressgröße von 16 Bit ergibt sich ein virtueller Adressraum mit der Größe $2^16 = 65536$ Bit. Mit der Word größe von 1 Bryte ergibt es 65536 Byte. Da es 16 Seiten gibt, folgt daraus eine Seitengröße von $\frac{65536}{16} = 4096$ Bytes.

Aufgabe 1.2:

Das Present/Absent-Bit gibt an, ob eine Seite aktuell im Hauptspeicher geladen ist. Es können also maximal so viele Seiten geladen sein, wie Platz im Hauptspeicher ist. Da die Länge der physikalischen Adressen 15 Bit ist, kann man von einer Speichergröße von $2^{15} * 8$Bit, also 32768 Byte ausgehen. Bei einer Seitengröße von 4096 Bytes passen somit $\frac{32768}{4096} = 8$ Seiten in den Hauptspeicher und es können maximal 8 Present/Absent-Bits gleichzeitig auf 1 gesetzt sein.

Aufgabe 1.3:

0x5fe8: Ersten 4 Bit geben Adresse der Seite an. Prüfung ob Seite Nr. 5 existiert und im Arbeitsspeicher liegt (Present/Absent-Bit gesetzt). Physikalsiche Adresse: 0x1fe8

Aufgabe 1.4:

Für kleinere Seiten spricht, dass bei speicherarmen Prozessen nicht viel Speicher für leere Seiten vergeudet wird. Ebenso bieten kleinere Seiten mehr Flexibilität bei der Seitenverdrängung: Wenn nur ein kleiner Teil Speicher gebraucht wird, müssen auch nur wenige Daten auf die Platte geschrieben werden.

Für große Seiten bzw. gegen kleinere Seiten spricht der Overhead, den eine kleine Seitengröße verursacht: zum einen braucht man dann größere Seitentabellen und zum anderen muss man eventuell öfter Seiten im Speicher hin- und herschieben und hat eine höhere Fragmentierung des Speicherbereichs.

Aufgabe 1.5:

p=4MB L=8 B s=? Seitengröße: $\frac{p}{s} \cdot L$ Verschwendung: $V(s) = \frac{p}{s} \cdot L + \frac{s}{2}$ $V'(s) = -\frac{p}{s^2} + \frac{1}{2} = 0$ $\rightarrow s = \sqrt{2pL} = \sqrt{64 \cdot 2^{20}}B = 8KiB$

Aufgabe 2:

Aufgabe 3: Synchronisation

Aufgabe 3.1: S

emaphore w =1 int Num = 0 Semaphore Mutex =1 process Writer: P(w) modifyData V(w)

processReader P(Mutex) Num++ if(Num == 1) P(w)

readData

if(Num == 1) V(w) Num – V(Mutex)