

Übungsblatt 10

Punktzahl gesamt: 60 Punkte

Abgabe der Lösungen bis spätestens 05.02.21 10:00 Uhr

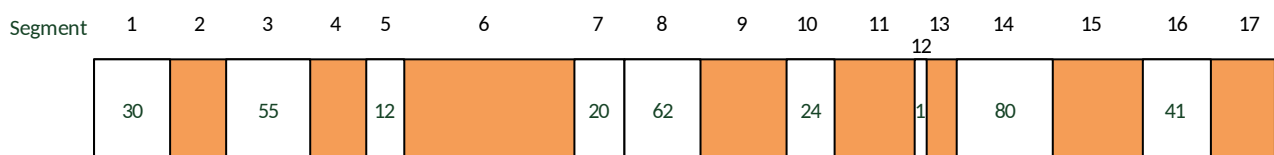
Aufgabe 1 – Verkettete Listen

35 Punkte

Die nachfolgende Abbildung zeigt die Speicherbelegung eines Systems. Dunkle Segmente sind belegt, und freie Segmente sind mit ihrer Größe in Kibibytes angegeben. Die Segmente sind durchnummeriert. Das System soll nacheinander Anforderungen von 52 KiB, 25 KiB, 62KiB befriedigen. Bei einer Teilung eines Segments kommen immer zuerst der belegte Teil und dann der verbleibende freie Rest. Geben Sie jeweils die Nummer der zugeteilten Segmente an, wenn die Zuteilung anhand der folgenden Algorithmen abläuft:

- 1) First Fit
- 2) Last Fit
- 3) Best Fit
- 4) Worst Fit
- 5) Next Fit

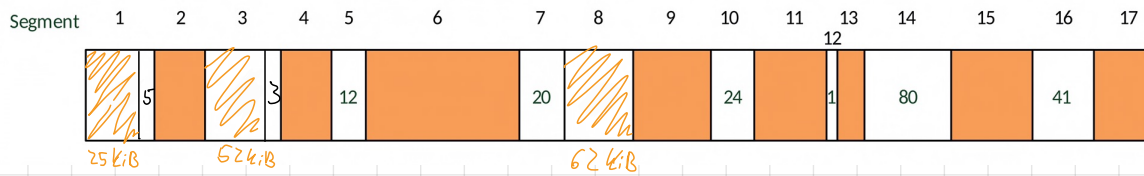
Ermitteln Sie außerdem jeweils die Zeit, die nötig ist, um alle Anforderungen zu befriedigen, wenn das Durchsuchen eines Listeneintrags 20 ns und das eventuelle Aufteilen 200 ns dauert.



12 Freies Segment mit Angabe der Größe Belegtes Segment

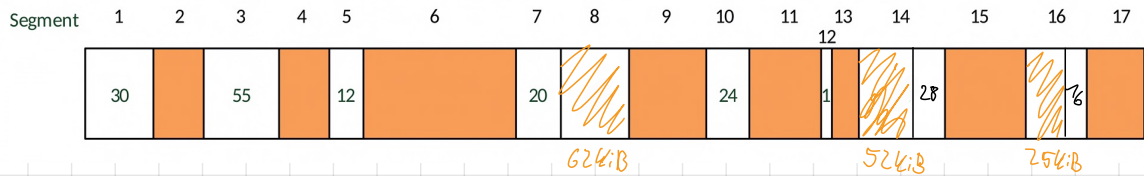
Aufgabe 7

First Fit



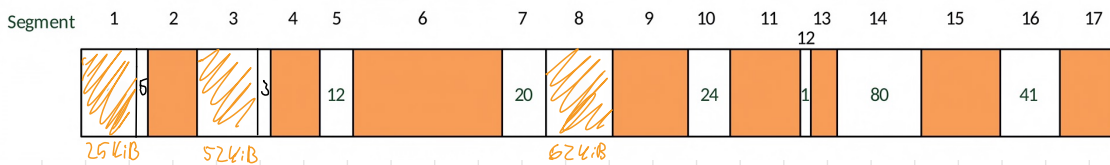
$$t = 220 \text{ ns} + 260 \text{ ns} + 760 = 640 \text{ ns}$$

Last Fit



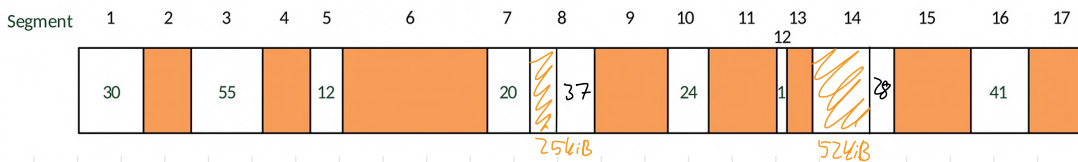
$$t = 160 \text{ ns} + 480 \text{ ns} + 520 \text{ ns} = 7760 \text{ ns}$$

Best Fit



$$t = 540 + 540 + 160 \text{ ns} = 7240 \text{ ns}$$

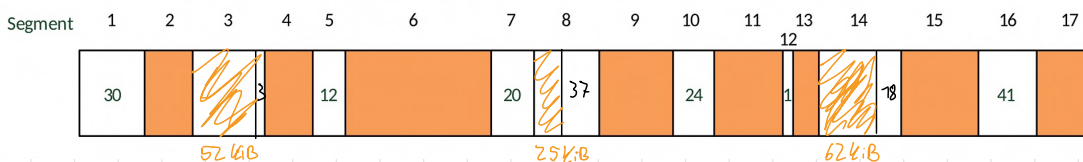
Worst Fit



62 KiB können erst bedient werden wenn neuer Speicher frei wird.

$$t = 540 \text{ ns} + 540 \text{ ns} + 340 \text{ ns} = 1420 \text{ ns}$$

Next Fit



$$t = 260 \text{ ns} + 320 \text{ ns} + 340 \text{ ns} = 920 \text{ ns}$$

Aufgabe 2 – Bitmaps

10 Punkte

Bei Bitmaps wird ein Feld von Bits zur Freispeicherverwaltung genutzt, wobei jeweils ein Bit für einen Speicherblock der Länge n steht (n wird auch als Blockungsfaktor bezeichnet). Ist das Bit gesetzt (=1), so ist der Block belegt, ist das Bit nicht gesetzt (=0) so ist er frei. Speicher kann nur in ganzen Blöcken vergeben werden.

- a) Wie groß ist der Anteil s , den ein Bitmap im Hauptspeicher einnimmt, wenn der Blockungsfaktor 8 Byte beträgt? Wie groß ist er bei 1 MiByte? Antwort muss begründet werden! **6 Punkte**
- b) Wie viel Speicher belegen die obigen Bitmaps, wenn 16 GiByte RAM zu verwalten sind? **4 Punkte**

Aufgabe 3 – Seitentabellen

15 Punkte

Ausgehend von einem System mit 24 Bit Adressbusbreite und einer einstufigen Seitentabelle mit einer Seitengröße von 8 KiB (KibiByte) sind die folgenden Fragen zu beantworten:

- a) Wie viele Bits sind für den Offsetting der virtuellen Adresse notwendig? **3 Punkte**
- b) Wie viele Bits bleiben für die Indizierung der Seitentabelle übrig, wenn alle restlichen Bits dafür genutzt werden? **3 Punkte**
- c) Wie lang und groß kann damit die Seitentabelle maximal werden, wenn jeder Eintrag 3 Byte belegt? **4 Punkte**
- d) Wir schalten auf mehrstufige Seitentabellen (2 stufige) um. Seitengröße und Adressbusbreite bleiben unverändert. Für die erste Stufe sind die ersten 3 Bit der Adresse reserviert. Prozess P_1 benötigt 5 MiB (MebiByte) an Speicherplatz. Wie viele Seitentabellen werden je Stufe für P_1 mindestens benötigt? **5 Punkte**

Aufgabe 2

a)

$$n = 8 \text{ Byte}$$

$$s = \frac{1}{1 + 8 \cdot 2^3} = \frac{1}{65}$$

$$n = 1 \text{ MiByte}$$

$$s = \frac{1}{1 + 8 \cdot 2^{20}} = \frac{1}{8388609}$$

b)

8 Byte Blockgröße

$$n = 8 \text{ Byte} \Rightarrow 16 \text{ GiByte} \cdot \frac{1}{65} = 8 \cdot 2^{34} \text{ Byte} \cdot \frac{1}{65} = 2^{37} \text{ Bit} \cdot \frac{1}{65} = 2114445438 \text{ Bit}$$

1 MiByte Blockgröße

$$\begin{aligned} n = 1 \text{ MiByte} &\Rightarrow 16 \text{ GiByte} \cdot \frac{1}{8388609} = 8 \cdot 2^{34} \text{ Byte} \cdot \frac{1}{8388609} \\ &= 2^{37} \text{ Bit} \cdot \frac{1}{8388609} \approx 16384 \text{ Bit} \end{aligned}$$

Aufgabe 3

24 Bit Adressbreite

$$8 \text{ KiB} = 8 \cdot 2^{13} \text{ Byte} = 2^{16} \text{ Byte} \quad \text{Seitengröße}$$

$$\text{Adressraum} = 2^{24} \text{ Byte} = 16 \text{ MiByte}$$

$$\text{virtuelle Adressen} \quad 8/16 \text{ Bit}$$

a) 16 Bit

b) 8 Bit

c)

$$\text{Eintrag } 3 \text{ Byte} = 24 \text{ Bit}$$

$$2^8 \cdot 24 = 6144 \text{ Bit} = 768 \text{ Byte}$$

d))