



### Aufgabe 3-3 Speicherbelegungsstrategien

In dieser Aufgabe sollen die drei Speicherbelegungsverfahren First Fit, Best Fit und Next Fit miteinander verglichen werden.

Die Güte eines solchen Verfahrens kann z.B. durch Vergleiche der Frei-Listen, die diese Verfahren auf eine bestimmte Situation und eine Folge von Anforderungen durch das Betriebssystem herstellen, bestimmt werden.

Offensichtlich erzeugen verschiedene Verfahren angewandt auf dieselbe Frei-Liste zu Anfang und dieselbe Folge von Anforderungen auch dieselbe Größe freien Speichers – vorausgesetzt, dass jede Anforderung auch bedient werden kann. Das bedeutet, dass die Summe über alle Elemente der entstehenden Frei-Listen in allen Fällen gleich ist.

Der freie Speicher kann jedoch am Ende in verschieden große Stücke aufgeteilt sein. Wir nennen an dieser Stelle ein Verfahren  $F_1$  schlechter als  $F_2$  bezüglich einer gegebenen Frei-Liste und einer Folge von Anforderungen, wenn die Frei-Liste, die  $F_1$  erzeugt, länger ist als diejenige, die  $F_2$  erzeugt. In diesem Fall hat  $F_1$  also den freien Speicher in mehr Stücke aufgeteilt als  $F_2$ .

Finden Sie für jeden der folgenden Fälle (a), (b) und (c) eine anfängliche Frei-Liste und eine Folge von Anforderungen, so dass bei allen Verfahren alle Anforderungen bedient werden können und bezüglich dieser

- (a) First Fit schlechter ist als Next Fit und Best Fit,
- (b) Next Fit schlechter ist als First Fit und Best Fit,
- (c) Best Fit schlechter ist als First Fit und Next Fit.

Geben Sie bei jeder Aufgabe jeweils folgendes an:

- die anfängliche Frei-Liste,
- die Liste der Anforderungen,
- jeweils die Frei-Liste, die durch First Fit, Next Fit bzw. Best Fit nach Abarbeitung aller Anforderungen entstanden ist.

Um die Korrektur dieser Aufgabe zu erleichtern, verwenden Sie bitte die Vorlage, die wir Ihnen auf der Website zum Download bereitstellen.

### Aufgabe 3-4 Seiteneretzungsstrategien

Beim Paging werden angeforderte Seiten nach Bedarf in freie Seitenrahmen des Hauptspeichers eingelagert. Eine priorisierte Seitenliste ist eine endliche Zahlenfolge und enthält die Nummern der sich momentan im Hauptspeicher befindlichen Seiten. Die Seiten sind in der Liste nach Priorität geordnet. Ganz rechts steht die Seite mit der niedrigsten Priorität, d.h. diejenige Seite, die beim nächsten Seitenfehler in den Hintergrundspeicher ausgelagert wird.

Es gibt eine Reihe verschiedener Seiteneretzungsstrategien, z.B.:

- Die *optimale Strategie* (OPT) ersetzt die Seite, die in Zukunft am längsten nicht angefordert werden wird.
- *First In First Out* (FIFO) ersetzt die Seite, die sich am längsten im Hauptspeicher befindet.
- *Least Recently Used* (LRU) ersetzt die Seite, die am längsten nicht mehr angefordert wurde.

Die optimale Strategie kann in der Praxis nicht verwendet werden, weil zukünftige Seitenzugriffe nicht im Voraus bekannt sind. Sie ist jedoch ein guter Maßstab zum Vergleich mit anderen Strategien.

Wir betrachten einen Hauptspeicher, der in drei Seitenrahmen unterteilt ist. Welche priorisierten Seitenlisten entstehen, wenn vom Betriebssystem die Seiten

2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2

in dieser Reihenfolge bei Verwendung der Verfahren

(a) OPT

(b) FIFO

(c) LRU

angefordert werden, wenn sich am Anfang keine Seite im Hauptspeicher befindet? Wieviele Seitenfehler ergeben sich bei den verschiedenen Verfahren?