

# Aufgabe 10: Tafelübung Speicherverwaltung

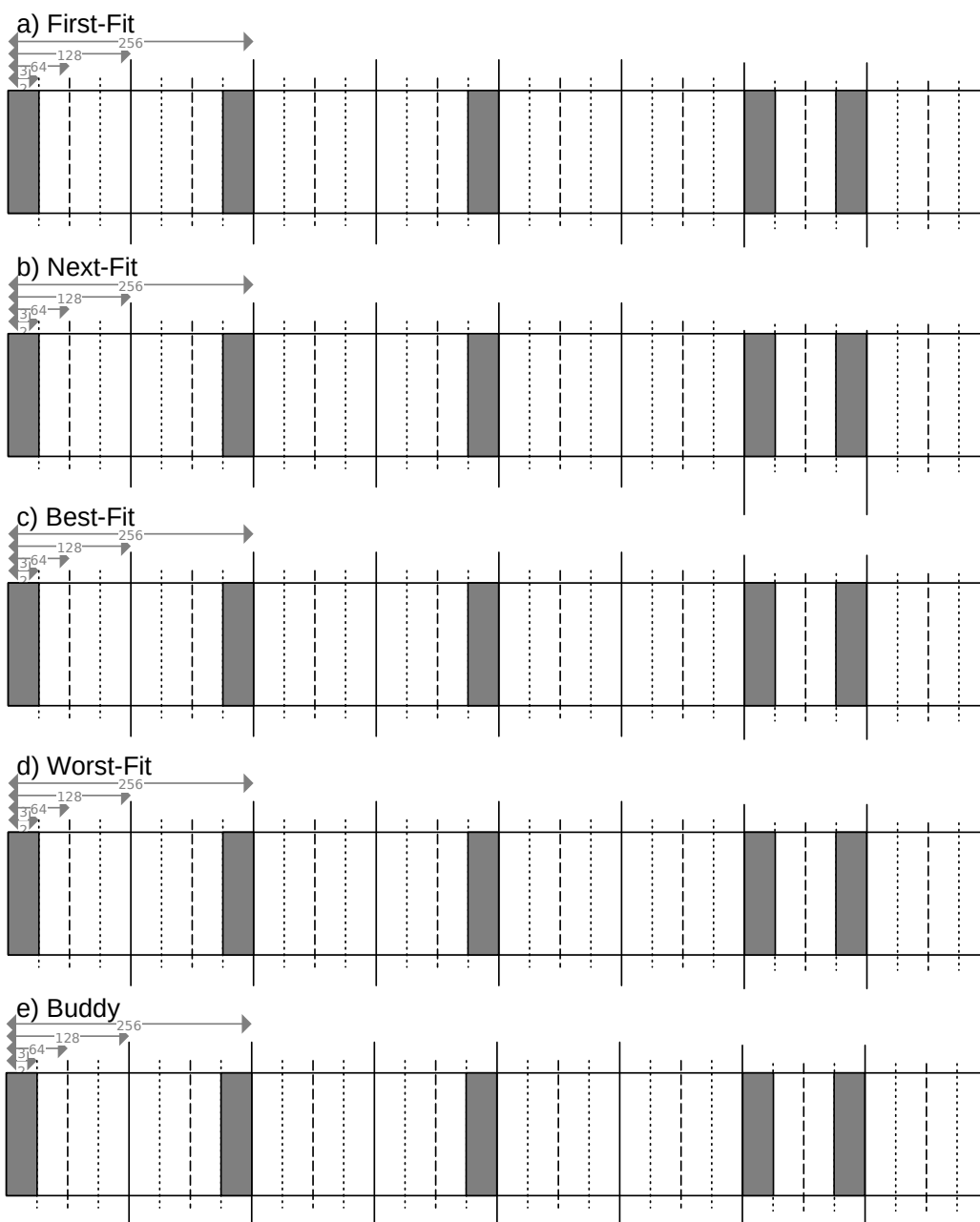
## 10.1 Speicherverwaltung

Zur Verwaltung von freiem Speicher (z.B. in Funktionen wie **malloc** und **free**) gibt es verschiedene Strategien bei der Speicherzuteilung.

Gegeben ist ein Speicherbereich von 1024 Byte mit den Adressen 0 bis 1023, der mittels **malloc** verwaltet werden soll. Ein Programm führt Speicheranfragen durch, die in der nachfolgenden Tabelle gegeben sind.

Bestimmen Sie für die Verfahren *First-Fit*, *Next-Fit*, *Best-Fit*, *Worst-Fit* und *Buddy* (Blockgröße  $2^n$ ), an welcher Adresse die neuen Speicherbereiche angelegt werden. Skizzieren Sie außerdem in der unten stehenden Grafik die Speicherbelegung nach der Ausführung aller Speicheranfragen und kennzeichnen Sie jeden Block mit der Nummer des jeweiligen **malloc**-Aufrufs. Gehen Sie davon aus, dass die in der Grafik ausgegrauten Speicherbereiche bereits belegt sind. Der Freispeicher wird als einfach verkettete Liste verwaltet und der Platz für Verwaltungsstrukturen soll bei dieser Aufgabe **nicht** berücksichtigt werden.

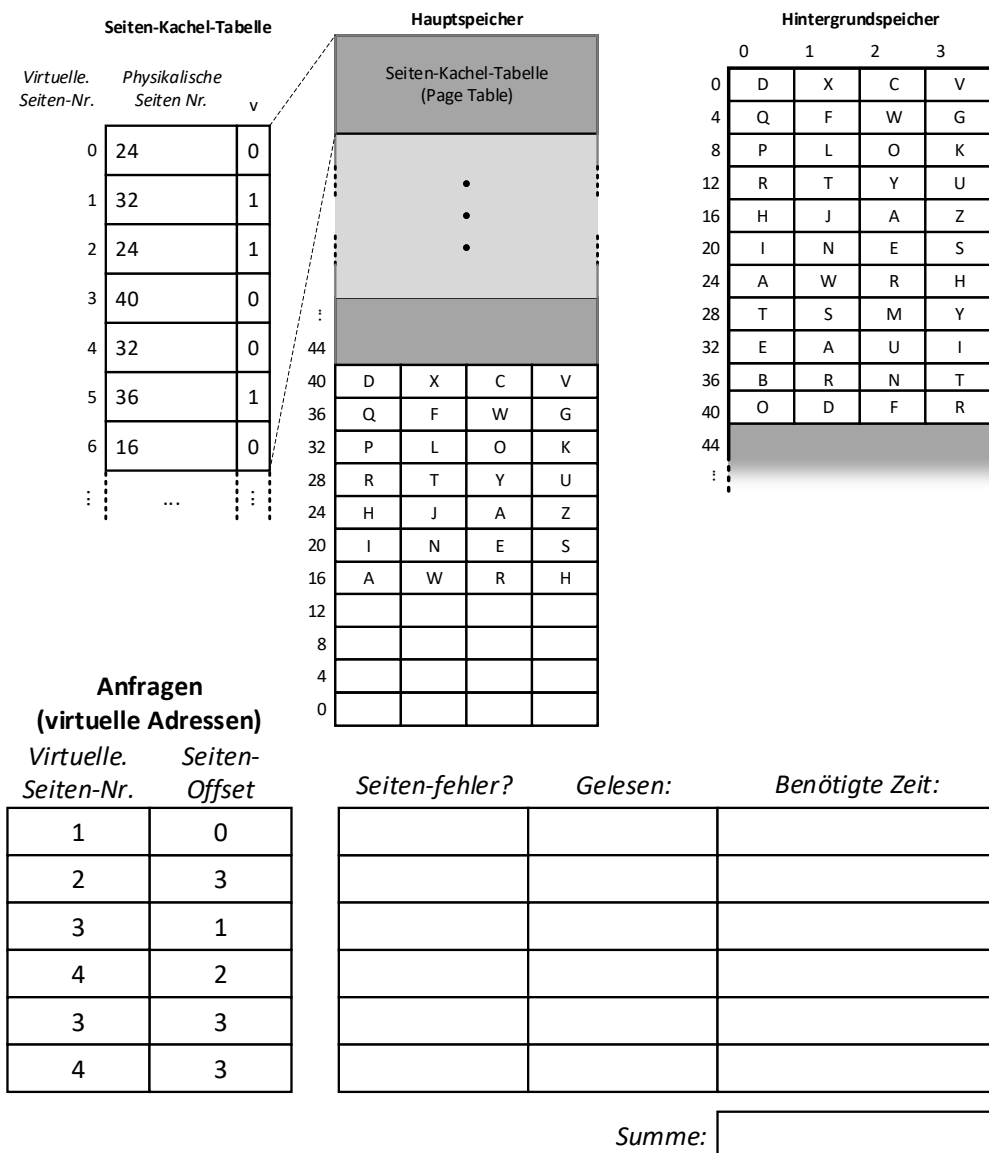
Aufruf	a) <b>First-Fit</b>	b) <b>Next-Fit</b>	c) <b>Best-Fit</b>	d) <b>Worst-Fit</b>	e) <b>Buddy</b>
1.) malloc( 88)					
2.) malloc( 48)					
3.) malloc( 12)					
4.) malloc(144)					
5.) malloc( 40)					



## 10.2 Virtueller Speicher

- a) Erklären Sie die Idee hinter Kachelverwaltung (Paging).

- b) In der unten stehenden Abbildung sind der Haupt- und der Hintergrundspeicher sowie eine Sequenz von lesenden Speicherzugriffen abgebildet. Bestimmen Sie den Inhalt der gelesenen Daten und geben Sie an, welche Speicherzugriffe zu einem Seitenfehler führen. Übertragen Sie im Fall von Seitenfehlern die Seiten in die leeren Zeilen des Hauptspeichers und passen Sie die Seiten-Kachel-Tabelle (SKT) entsprechend an.
- c) Angenommen das Kopieren einer Seite aus dem Hintergrundspeicher in den Hauptspeicher benötigt 10 ms, während der direkte Zugriff auf den Hauptspeicher nur 10 ns benötigt. Wie viel Zeit wird dann zum Lesen der Daten aus Aufgabenteil b) benötigt? Gehen Sie davon aus, dass im Hauptspeicher Zugriffe auf eine Seite mit Daten und auf die SKT nicht gleichzeitig erfolgen können.



- d) Ausgehend von einer 32-Bit Prozessorarchitektur verfügt ein Computer meist über einen virtuellen Adressraum von 32 Bit. Angenommen die Adressierung erfolgt Byte-weise und der virtuelle Adressraum teilt sich auf in 21 Bit für die Adressierung der Seiten-Kachel-Tabelle und 11 Bit für Seiten-Offset.

1. Wie viel Speicher umfasst eine Seite?

2. Wie viele Seiten hat die SKT?

3. Angenommen jede Zeile in der SKT umfasst 32 Bit (21 Bit Speicheradresse + Valid-Bit für Auslagerung in Hintergrundspeicher + 1 Dirty Bit + 9 Bit für diverse Funktionen), wie groß ist der Speicherbedarf für die Seiten-Kachel-Tabellen von 135 Prozessen?

4. Wiederhole die Schritte 1. bis 3. für einen virtuellen Adressraum von 64-Bit (44 Bit SKT-Adressen und 20 Bit Seiten-Offset) und einer SKT-Zeilenzlänge von 55 Bit. Was fällt auf?

### 10.3 Ersetzungsstrategien

Gegeben ist eine fiktive Seiten-Kachel-Tabelle mit drei Kacheln und die Referenzfolge:

3, 2, 5, 4, 2, 6, 2, 4, 3, 4

Bestimmen Sie den Ablauf und die Anzahl der Ersetzungen für folgende Ersetzungsstrategien.

a) *first-in, first-out* (FIFO)

Referenzfolge		3	2	5	4	2	6	2	4	3	4
Hauptspeicher	Kachel 1										
	Kachel 2										
	Kachel 3										
Kontrollzustände	Kachel 1										
	Kachel 2										
	Kachel 3										

Anzahl an Ersetzungen:

b) *least-recently-used* (LRU)

Referenzfolge		3	2	5	4	2	6	2	4	3	4
Hauptspeicher	Kachel 1										
	Kachel 2										
	Kachel 3										
Kontrollzustände	Kachel 1										
	Kachel 2										
	Kachel 3										

Anzahl an Ersetzungen:

b) *Second-Chance* (Clock)

Referenzfolge		3	2	5	4	2	6	2	4	3	4
Hauptspeicher	Kachel 1										
	Kachel 2										
	Kachel 3										
Kontrollzustände	Kachel 1										
	Kachel 2										
	Kachel 3										
	Umlaufzeiger										

Anzahl an Ersetzungen:

## 10.4 Scheduling-Verfahren für Schreib-/Lesekopf

Die Geschwindigkeit, mit der auf die Festplatte zugegriffen wird, hängt stark von dem Scheduling-Verfahren des Lese- bzw. Schreibkopfes ab. Angenommen Sie haben eine Festplatte mit 64 Spuren (0-63) und der Lese-/Schreibkopf befindet sich an Spur 29. Gegeben ist eine zeitlich geordnete Folge von Zugriffen auf die Spuren:

55, 57, 35, 27, 19, 48, 40, 42, 1, 53.

Bringen Sie in den folgenden Tabellen die Zugriffsfolge in eine dem Scheduling-Verfahren entsprechende Reihenfolge. Berechnen Sie dann die Suchzeit (engl. search-time), die für die Umstellung des Lese-/Schreibkopfes benötigt wird, und geben Sie den Zugriffszeitpunkt an. Gehen Sie davon aus, dass für den Wechsel zwischen zwei direkt benachbarten Spuren 2 ms benötigt werden.

a) Scheduling-Verfahren: *first-in, first-out* (FIFO).

Spur										
Suchzeit										
Zugriffszeitpunkt										

b) Scheduling-Verfahren: *shortest seek time first* (SSTF).

Spur										
Suchzeit										
Zugriffszeitpunkt										

c) Scheduling-Verfahren: *elevator*. (Gehen Sie zuerst in Richtung kleinerer Spurnummern.)

Spur										
Suchzeit										
Zugriffszeitpunkt										

Besprechung der Lösung am 22.01.19 in der großen Übung