

Praktikum Systemprogrammierung

## Versuch 3

### *Allokationsstrategien*

Lehrstuhl Informatik 11 - RWTH Aachen

6. April 2023

Commit: b3175525

# Inhaltsverzeichnis

<b>3</b>	<b>Allokationsstrategien</b>	<b>3</b>
3.1	Next-Fit . . . . .	3
3.2	Best-Fit . . . . .	4
3.3	Worst-Fit . . . . .	4

---

Dieses Dokument ist Teil der begleitenden Unterlagen zum *Praktikum Systemprogrammierung*. Alle zu diesem Praktikum benötigten Unterlagen stehen im Moodle-Lernraum unter <https://moodle.rwth-aachen.de> zum Download bereit. Folgende E-Mail-Adresse ist für Kritik, Anregungen oder Verbesserungsvorschläge verfügbar:

support.psp@embedded.rwth-aachen.de

## 3 Allokationsstrategien

Die Platzierung von angeforderten Speicherbereichen eines dynamischen Speichers kann auf unterschiedliche Arten bestimmt werden. In Versuch 3 wurde bereits die Allokationsstrategie *First-Fit* vorgestellt. Im Folgenden werden in Ergänzung dazu drei weitere Allokationsstrategien vorgestellt. Zusätzlich wird jede dieser Strategien am Beispiel eines konkreten Speicherbereichs mit einer Größe von 25 Byte veranschaulicht. In der folgenden Abbildung 3.1 sind 9 Byte in Form von 6 Abschnitten belegt (gelb) und noch 16 Byte frei (weiß).

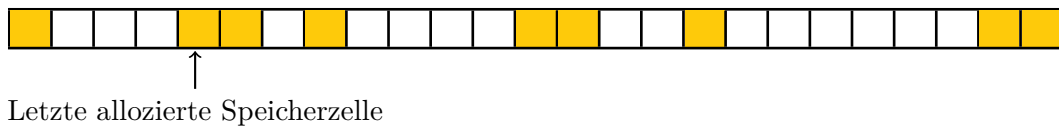


Abbildung 3.1: Fragmentierter Speicher mit 25 Byte Größe

### 3.1 Next-Fit

Die Allokationsstrategie *Next-Fit* bzw. *Rotating-First-Fit* sucht analog zu *First-Fit* den ersten passenden Speicherbereich. Die Suche nach einem freien Speicherbereich mit ausreichender Größe beginnt jedoch bei der ersten Adresse nach dem zuletzt zugewiesenen Speicherbereich. Wurde zuletzt ein Speicherbereich der Größe  $k$  an Adresse  $p$  vergeben, wird die Suche an der Adresse  $p + k$  begonnen, sofern diese Adresse gültig ist. Anderenfalls wird am Anfang des Speichers begonnen. Hierbei ist der Fall zu berücksichtigen, dass wenn die Suche am Ende des Speichers angekommen ist, es unter Umständen Bereiche am Anfang gibt, die noch nicht untersucht wurden. Abbildung 3.2 zeigt exemplarisch die Speicherbelegung nach Anwendung dieser Strategie. Bitte beachten Sie, dass die zwischengespeicherte Adresse des zuletzt vergebenen Speicherbereichs nur dann verändert werden soll, wenn ein Speicherbereich mit Next-Fit alloziert wurde.



Abbildung 3.2: Allokation von zwei Byte mit Next-Fit

### 3.2 Best-Fit

Die Allokationsstrategie *Best-Fit* liefert auf Speicherbereichsanfragen den kleinsten passenden Speicherbereich zurück. Dazu muss für jede Anfrage der gesamte Speicherbereich, beginnend bei der ersten Adresse, durchsucht werden. Wird ein exakt passender Speicherbereich gefunden, kann die Suche vorzeitig abgebrochen werden und der gefundene Speicherbereich als Ergebnis zurückgegeben werden. Abbildung 3.3 zeigt exemplarisch die Speicherbelegung nach Anwendung dieser Strategie.



Abbildung 3.3: Allokation von zwei Byte mit Best-Fit

### 3.3 Worst-Fit

Die Allokationsstrategie *Worst-Fit* liefert im Gegensatz zu *Best-Fit* den größten passenden Speicherbereich zurück. Auch hier wird mit der ersten Adresse begonnen. Die Suche nach einem solchen Speicherbereich kann nur dann vorzeitig abgebrochen werden, falls ein freier Speicherbereich gefunden wurde, dessen Größe mindestens die Hälfte des insgesamt vorhandenen dynamischen Speichers beträgt. Abbildung 3.4 zeigt exemplarisch die Speicherbelegung nach Anwendung dieser Strategie.



Abbildung 3.4: Allokation von zwei Byte mit Worst-Fit