

# Approximations- und Online-Algorithmen SS 2004

## Übungsblatt 1

Abgabe: Montag, den 3. Mai 2004, 14.00 Uhr (in der Vorlesung)

### Aufgabe 1: Ski-Ausleihe (6 Punkte)

Ein Student möchte mit dem Skilaufen beginnen und hat die Möglichkeit, ein Paar Ski für 1 Kosteneinheit zu leihen oder für  $n \in \mathbb{N}, n \geq 2$  zu kaufen. Wir nehmen an, dass sich der Preis im Laufe der Zeit nicht ändert. Entwickeln Sie eine  $(2 - \frac{1}{n})$ -kompetitive Online-Strategie für dieses Problem. Dabei sei im voraus nicht bekannt, wie oft der Student Ski fährt.

### Aufgabe 2: Scheduling (6 Punkte)

In der Vorlesung haben wir das Scheduling-Problem betrachtet,  $n$  Jobs auf  $m$  Maschinen so einzuplanen, dass das Bearbeitungsende des zuletzt fertiggestellten Jobs minimiert wird. Zeigen Sie, dass der in der Vorlesung vorgestellte Algorithmus **Greedy** keinen kompetitiven Faktor kleiner als  $2 - \frac{1}{m}$  besitzt.

### Aufgabe 3: Paging (6 Punkte)

In der Vorlesung wurde auch der Paging-Algorithmus **FIFO** vorgestellt. Zeigen Sie, dass **FIFO**  $k$ -kompetitiv ist, wobei  $k$  die Größe des schnellen Speichers bezeichnet.

### Aufgabe 4: Paging (6 Punkte)

Es sei wieder  $k$  die Größe des schnellen Speichers und  $\sigma$  eine Anfragesequenz. Wir unterteilen  $\sigma$  in  $k$ -Phasen: Die  $k$ -Phase 0 ist die leere Sequenz. Für alle  $i \in \mathbb{N}$  ist die  $k$ -Phase  $i$  die maximale Sequenz nach der  $k$ -Phase  $i - 1$ , die  $k$  verschiedene Seiten enthält. Die letzte  $k$ -Phase kann weniger als  $k$  verschiedene Seiten enthalten.

Zu Beginn einer  $k$ -Phase werden alle Seiten unmarkiert. Wenn eine Seite in einer  $k$ -Phase zum ersten Mal angefragt wird, so wird sie markiert.

Ein Paging-Algorithmus ist genau dann ein **Marking**-Algorithmus, wenn der Algorithmus bei einem Seitenfehler nur unmarkierte Seiten aus dem schnellen Speicher entfernt. Geben Sie ein Beispiel für einen **Marking**-Algorithmus an und zeigen Sie, dass jeder **Marking**-Algorithmus  $k$ -kompetitiv ist.

### Informationen zur Vorlesung

Die neuesten Informationen zur Vorlesung und Übung befinden sich unter:

<http://www.informatik.uni-freiburg.de/~salbers/lehre/ss04/alg.html>

Die Übungsblätter können in Gruppen bis zu zwei Studierenden bearbeitet werden.

Um Kreditpunkte zu erhalten, müssen Studierende

- 50% der Übungsaufgaben bearbeiten,
- ihre Lösungen mindestens zweimal in den Übungen präsentieren und
- eine Abschlussprüfung bestehen.