
Effiziente Algorithmen und Datenstrukturen II

Abgabetermin: 11.07.2008 vor der Vorlesung

Aufgabe 1 (10 Punkte)

- (a) Geben Sie einen Algorithmus an, der einen optimalen Schedule für das Problem $1|p_i = 1|\sum U_i$ mit $d_i \in \mathbb{N}$ in Zeit $O(n)$ berechnet.
- (b) Wir betrachten das Scheduling-Problem $P|intree; p_i = 1|C_{\max}$ mit m Prozessoren. Zeigen Sie, dass der Algorithmus von Hu (Highest-Level-First) zur Lösung dieses Problems mit Laufzeit $O(n)$ implementiert werden kann.

Aufgabe 2 (10 Punkte)

Gegeben seien 9 Tasks t_i , $1 \leq i \leq 9$, mit den Bearbeitungszeiten p_i

i	1	2	3	4	5	6	7	8	9
p_i	3	2	2	2	4	4	4	4	9

und den Vorrangbedingungen $t_1 \rightarrow t_9$, $t_4 \rightarrow t_5$, $t_4 \rightarrow t_6$, $t_4 \rightarrow t_7$ und $t_4 \rightarrow t_8$ (d.h. der Task t_9 kann erst dann ausgeführt werden, wenn der Task t_1 vollständig bearbeitet ist). Weiterhin sei eine Prioritätsliste $L = (t_1, t_2, \dots, t_9)$ gegeben.

Ein Schedule wird nun wie folgt berechnet: Zu Beginn scannen alle Prozessoren die Liste L und suchen nach Tasks, die zur Ausführung bereit stehen. Der erste derartige Task wird dann von einem freien Prozessor M_j bearbeitet. Falls es keinen Task ohne Vorrangbedingungen gibt, wartet M_j solange, bis ein Prozessor einen Task abgearbeitet hat, und überprüft die Liste L anschließend erneut. Konkurrieren mehrere Prozessoren M_j um einen Task t_i , so wird t_i vom Prozessor mit kleinstem Index j bearbeitet.

- (a) Es stehen $m = 3$ Prozessoren zur Verfügung. Geben Sie den berechneten Schedule sowie die zur Ausführung benötigte Zeit C_{\max} an.
- (b) Zeigen Sie, dass eine Erhöhung der Anzahl Prozessoren die Zeit C_{\max} erhöhen kann.
- (c) Zeigen Sie, dass eine Verringerung der Bearbeitungszeiten C_{\max} erhöhen kann.

Aufgabe 3 (10 Punkte)

Wir betrachten das Scheduling-Problem $1|outtree|\sum w_i C_i$. Gegeben seien Tasks t_i mit $1 \leq i \leq 5$ und den folgenden Gewichten w_i sowie Bearbeitungszeiten p_i :

i	1	2	3	4	5
w_i	4	3	2	8	9
p_i	6	2	2	3	4

Die Vorrangbedingungen sind $t_1 \rightarrow t_2$, $t_1 \rightarrow t_3$, $t_3 \rightarrow t_4$ und $t_3 \rightarrow t_5$.

Bestimmen Sie eine optimale Auftragsreihenfolge und den zugehörigen Zielfunktionswert.

Aufgabe 4 (10 Punkte)

Eine Spedition muss entscheiden, wie sie eine Ladung, die aus n Containern besteht, auf ihre Lastwagen verteilt. Sei w_i mit $1 \leq i \leq n$ das Gewicht des Containers i . Die Lastwagen sind alle identisch und können jeweils Ladung mit Gesamtgewicht W transportieren. Das Ziel ist nun, die Container so aufzuteilen, dass die Anzahl der dabei verwendeten Lastwagen minimal ist.

Geben Sie einen polynomiellen Algorithmus an, der eine Zuordnung findet, die – im Vergleich zu einer optimalen Zuordnung – höchstens doppelt so viele Lastwagen benutzt.