

# PS Algorithmen und Datenstrukturen 2024

## Aufgabenblatt 11

### Aufgabe 31

Entwickeln Sie einen Divide-and-Conquer Algorithmus, der die Höhe eines binären Suchbaums mit  $n$  Knoten berechnet, und zeigen Sie, dass seine Laufzeit  $O(n)$  beträgt.

*Hinweis:* Stellen Sie eine rekursive Laufzeitabschätzung auf und lösen Sie diese direkt mit einer geeigneten Induktionshypothese (anstatt zu versuchen, das Master-Theorem anzuwenden).

### Aufgabe 32

Ein Matching  $M$  eines ungerichteten Graphen ist eine Teilmenge der Kanten für die gilt, dass jeder Knoten inzident zu höchstens einer Kante aus  $M$  ist. Ein maximales Matching (auch nicht-erweiterbares Matching genannt) ist ein Matching, zu dem keine Kante mehr hinzugefügt werden kann, ohne die oben genannte Eigenschaft zu verletzen.

Entwickeln Sie einen Algorithmus mit Laufzeit  $O(|V| + |E|)$ , der ein maximales Matching eines ungerichteten Graphen  $G = (V, E)$  berechnet.

### Aufgabe 33

Auf einem einzelnen Prozessor werden  $n$  Jobs abgearbeitet. Jeder Job  $i$  besitzt eine Dauer  $t_i$  und eine Deadline  $d_i$ , bis zu der er beendet sein soll.

Wir betrachten das Scheduling-Problem, eine Abarbeitungsreihenfolge der Jobs zu erstellen, sodass möglichst alle Jobs bis zu ihrer jeweiligen Deadline beendet sind. Ist dies nicht möglich, können einzelne Jobs verspätet beendet werden. Ein Job  $i$  wird verspätet beendet, wenn für seinen Beendigungszeitpunkt  $f_i$  gilt  $f_i > d_i$ . Dieser Job weist dann eine Verspätung  $\ell_i = f_i - d_i$  auf. Wird Job  $i$  rechtzeitig beendet, dann wird  $\ell_i = 0$  gesetzt. Das Ziel der Erstellung der Abarbeitungsreihenfolge ist, dass die Summe aller Verspätungen minimal ist.

Zeigen Sie, dass keine der folgenden Greedy-Strategien eine optimale Lösung liefert:

1. *Job Length:* Die Jobs werden nach aufsteigender Dauer  $t_i$  sortiert.
2. *Earliest Deadline:* Die Jobs werden nach aufsteigender Deadline  $d_i$  sortiert.
3. *Slack Time:* Die Jobs werden nach aufsteigender Slack Time  $d_i - t_i$  sortiert.