

Klausur Algorithmen und Datenstrukturen 2 (Wintersemester 2011/2012) Lösungshinweise

(Alle Angaben ohne Gewähr¹)

Aufgabe 1. Die Wortlänge der Schlüssel ist $w = 16$. Deshalb muss zunächst A durch eine 16-Bit Zahl s approximiert werden:

$$s = \lceil A \cdot 2^{16} \rceil = 34315$$

Zur Berechnung des Hashwerts des Schlüssels $k = 12345$ berechnet man

$$\begin{aligned} r &\equiv 34315 \cdot 12345 && (\text{mod } 65536) \\ &\equiv 423618675 && (\text{mod } 65536) \\ &= 59507 && (\text{mod } 65536) \end{aligned}$$

Die Größe der Hashtabelle ist $m = 2^{10}$. Deshalb werden die 10 höchstwertigsten Bits von 59507 zur Berechnung des Slots verwendet:

$$59507 = (\underbrace{1110100001}_{=929} 110011)_2$$

Also ist $h(12345) = 929$. Alternativ kann man 59507 um 6 Bits nach rechts verschieben, um den Slot zu berechnen: $\lfloor \frac{59507}{2^6} \rfloor = 929$. Auf diese Weise spart man sich die Umwandlung von und in Binärzahlen.

Aufgabe 2.

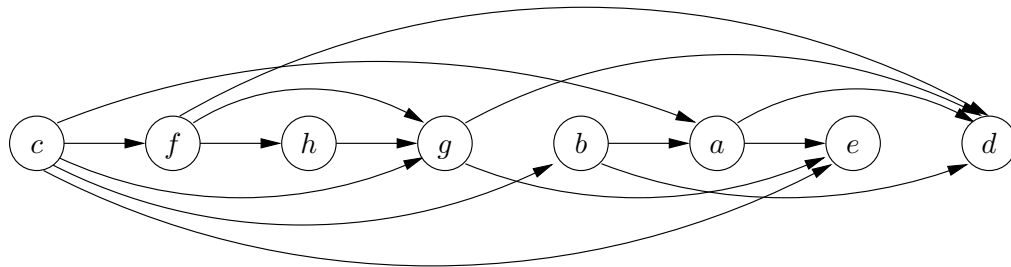
a) Die Tiefensuche liefert folgendes Ergebnis:

| v | $d[v]$ | $f[v]$ | $\pi[v]$ |
|-----|--------|--------|----------|
| a | 1 | 6 | — |
| d | 2 | 3 | a |
| e | 4 | 5 | a |
| b | 7 | 8 | — |
| c | 9 | 16 | — |
| f | 10 | 15 | c |
| g | 11 | 12 | f |
| h | 13 | 14 | f |

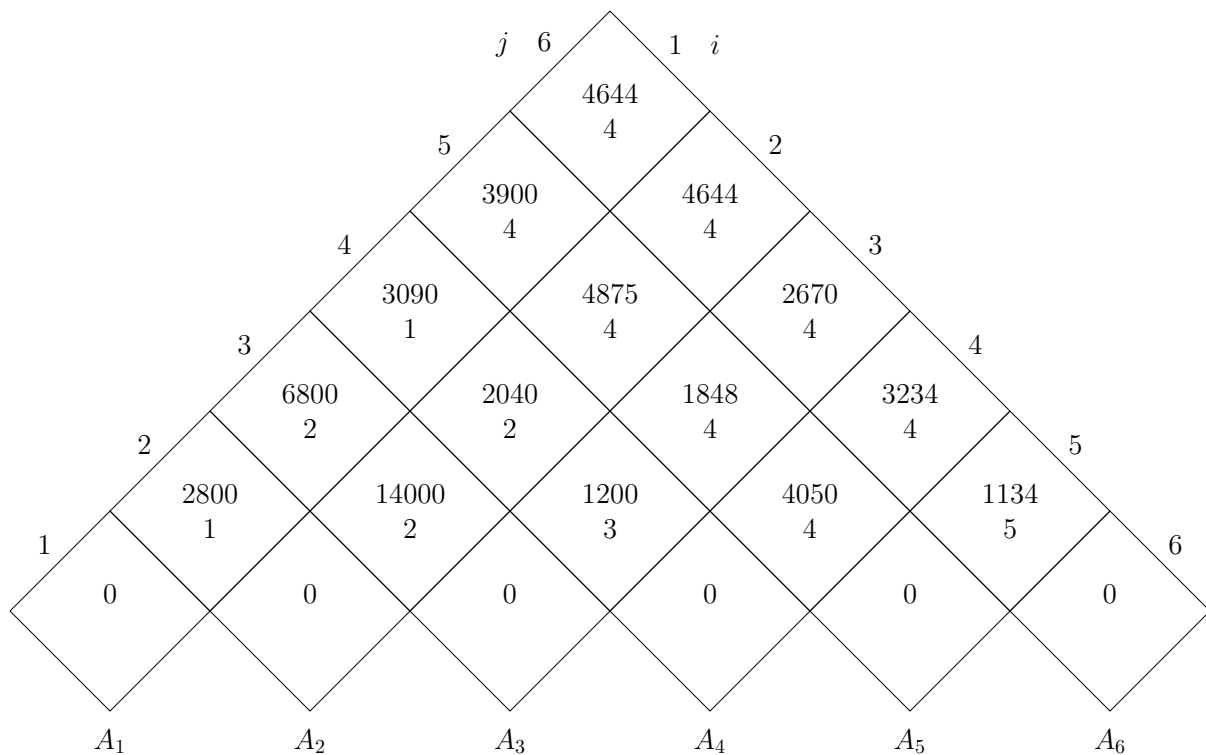
Hierbei wurden die Knoten immer in alphabetischer Reihenfolge durchlaufen.

¹Sachdienliche Hinweise zur Fehlerbekämpfung senden Sie bitte an christoph.karg@htw-aalen.de

- b) Der Graph ist topologisch sortierbar, denn er enthält keinen Zyklus. Dies kann daran erkennen, dass die Tiefensuche keine Back Edges findet. Die topologische Sortierung wird anhand der Endzeiten der Tiefensuche erstellt:



Aufgabe 3. Die vollständige Tabelle ist:



a) $i = 3, j = 4: 0 + 0 + 8 \cdot 50 \cdot 3 = 1200$

b) $i = 1, j = 5$:

$$k = 1 : \quad 0 + 4875 + 10 \cdot 35 \cdot 27 = 14325$$

$$k = 2: \quad 2800 + 1848 + 10 \cdot 8 \cdot 27 = 6808$$

$$k = 3: \quad 6800 + 4050 + 10 \cdot 50 \cdot 27 = 24350$$

$$k = 4 : \quad 3090 + 0 + 10 \cdot 3 \cdot 27 = 3900$$

Optimum bei $k = 4$

c) Die optimale Klammerung ist: $(A_1(A_2(A_3A_4)))(A_5A_6)$.

Aufgabe 4. Zur Berechnung des Optimums werden die Aktivitäten nach aufsteigender Endzeit sortiert. Dann werden die Aktivitäten gemäß dem Algorithmus aus der Vorlesung ausgewählt:

| i | s_i | f_i | $Auswahl$ |
|-----|-------|-------|-----------|
| 1 | 1 | 4 | ★ |
| 3 | 3 | 8 | |
| 4 | 5 | 8 | ★ |
| 5 | 2 | 9 | |
| 2 | 6 | 10 | |
| 10 | 8 | 12 | ★ |
| 8 | 4 | 13 | |
| 9 | 10 | 15 | |
| 7 | 12 | 16 | ★ |
| 6 | 14 | 18 | |

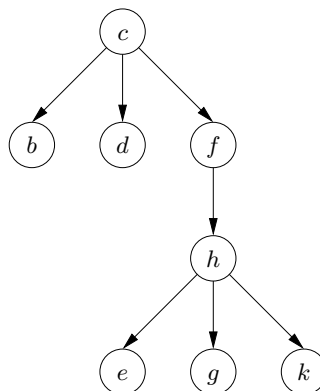
Die Menge der ausgewählten Aktivitäten ist $\{1, 4, 7, 10\}$.

Aufgabe 5.

a) Die Knoten des Graphen werden in folgender Reihenfolge untersucht:

| x | c | b | d | f | h | e | g | k | a |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|----------|
| $d(x)$ | 0 | 1 | 1 | 1 | 2 | 3 | 3 | 3 | ∞ |
| $\pi(x)$ | — | c | c | c | f | h | h | h | — |

b) Der Breitensuche-Baum ist wie folgt:



Der Knoten a wird von der Breitensuche nicht gefunden.

Aufgabe 6.

a) Berechnung der kürzesten Pfade mit dem Dijkstra-Algorithmus:

| <i>Auswahl</i> | <i>v</i> | <i>a</i> | <i>b</i> | <i>c</i> | <i>d</i> | <i>e</i> | <i>f</i> | <i>g</i> | <i>h</i> |
|----------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| — | $d[v]$ | 0 | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ |
| | $\pi[v]$ | — | — | — | — | — | — | — | — |
| <i>b</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | ∞ | 23 | ∞ |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | — | <i>b</i> | — |
| <i>e</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | 19 | 23 | ∞ |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | <i>e</i> | <i>b</i> | — |
| <i>a</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | 19 | 23 | ∞ |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | <i>e</i> | <i>b</i> | — |
| <i>c</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | 10 | 13 | 21 |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | <i>c</i> | <i>c</i> | <i>c</i> |
| <i>f</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | 10 | 13 | 15 |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | <i>c</i> | <i>c</i> | <i>f</i> |
| <i>g</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | 10 | 13 | 15 |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | <i>c</i> | <i>c</i> | <i>f</i> |
| <i>h</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | 10 | 13 | 15 |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | <i>c</i> | <i>c</i> | <i>f</i> |
| <i>d</i> | $d[v]$ | 8 | 0 | 8 | 19 | 3 | 10 | 13 | 15 |
| | $\pi[v]$ | <i>b</i> | — | <i>b</i> | <i>b</i> | <i>b</i> | <i>c</i> | <i>c</i> | <i>f</i> |

b) Ein kürzester Pfad von *b* nach *h* mit einer Länge von 15 Einheiten ist

$$b \xrightarrow{8} c \xrightarrow{2} f \xrightarrow{5} h.$$

Aufgabe 7. Ein Binomial Heap mit den Schlüsseln als Inhalt ist beispielsweise:

