Musterlösung zur Vordiplomklausur Datenstrukturen und Programmierverfahren Sommersemester 2001

1. (a) Gegenbeispiel: $f(n) = 2^n$.

$$\begin{aligned} & 0 \le 2^n \le c2^{n/2} \\ \Leftrightarrow\ & 0 \le 2^{n/2}2^{n/2} \le c2^{n/2} \\ \Leftrightarrow\ & 0 \le 2^{n/2} \le c \end{aligned}$$

Was ein Widerspruch zur Unbeschränktheit der Natürlichen Zahlen ist.

(b)

$$\begin{aligned} & f(n) = O(g(n)) \\ \Rightarrow\ & 0 \le f(n) \le cg(n) \\ \Rightarrow\ & 0 \le 1/cf(n) \le g(n) \\ \Rightarrow\ & g(n) = \Omega(f(n)) \end{aligned}$$

(c) Gegenbeispiel: $f(n) = n^2$.
Für $f(n) = n^2$ gilt $n^3 = \omega(f(n))$ und $n = o(f(n))$. Da $o(f(n)) + \omega(f(n)) = \Theta(Max(o(f(n)), \omega(f(n))))$ gilt, und $n^3 \neq \Theta(n)$, ist die Aussage falsch.

2. (a) Zu lösen:

$$T(n) = T(n/2) + n \lg^2 n$$

Lösen mittels Master-Theorems (3.Fall):
Da $\log_b a = \log_2 1 = 0$ gilt $n \lg^2 n = \Omega(n^{\log_b a + \epsilon})$ $(\epsilon = 1)$.
Zu zeigen, $(n/2)\lg^2(n/2) \le cn \lg^2 n$ für genügend große n und $c < 1$.

$$\begin{aligned} & (n/2)\lg^2(n/2) \le cn \lg^2 n \\ \Leftrightarrow\ & (n/2)(\lg(n/2)\lg(n/2)) \le cn \lg^2 n \\ \Leftrightarrow\ & (n/2)(\lg n - 1)^2 \le cn \lg^2 n \end{aligned}$$

Ungleichung ist erfüllt für alle $1/2 < c < 1$.

(b) Zu lösen:

$$T(n) = 2T(n/2) + n/\lg n$$

Lösen mittels Iterationsmethode:

$$
\begin{aligned}
T(n) &= 2T(n/2) + n/\lg n \\
&= n/\lg n + 2T(n/2) \\
&= n/\lg n + 2\left(\frac{n/2}{\lg(n/2)} + 2T(n/4)\right) \\
&= n/\lg n + \frac{n}{\lg(n/2)} + 4T(n/4) \\
&= n/\lg n + \frac{n}{\lg(n/2)} + \frac{n}{\lg(n/4)} + 8T(n/8) \\
&\vdots \\
&= \sum_{i=0}^{k-1} \frac{n}{\lg(n/2^i)} + 2^k T(n/2^k)
\end{aligned}
$$

Da $T(1) = \Theta(1)$ und $n/2^k = 1$ gdw. $k = \lg n$, gilt:

$$
\begin{aligned}
T(n) &= \sum_{i=0}^{\lg n-1} \frac{n}{\lg(n/2^i)} + n\Theta(1) \\
&= n \sum_{i=0}^{\lg n-1} \frac{1}{\lg n - i} + n\Theta(1) \\
&= n \sum_{j=1}^{\lg n} \frac{1}{j} + n\Theta(1) \\
&= n \lg\lg n + n\Theta(1) \\
&= \Theta(n \lg^2 n)
\end{aligned}
$$

3. (a)
   - $\langle 1 \rangle$
   - $\langle 7, 1 \rangle$
   - $\langle 7, 1, 2 \rangle$
   - $\langle 7, 6, 2, 1 \rangle$
   - $\langle 7, 6, 2, 1, 3 \rangle$
   - $\langle 7, 6, 5, 1, 3, 2 \rangle$
   - $\langle 7, 6, 5, 1, 3, 2, 4 \rangle$

   (b) Maximale Elementanzahl für einen Heap der Höhe h: $2^{h+1}-1$, minimale Anzahl: 2^h.

4. (a) Mögliche gemeinsame Teilsequenz von $X := \langle 3, 2, 1, 4, 7, 6 \rangle$ und $Y := \langle 1, 2, 3, 4, 6, 7 \rangle$ mit maximaler Länge:
   - $\langle 3, 4, 7 \rangle$
   - $\langle 3, 4, 6 \rangle$
   - $\langle 2, 4, 7 \rangle$
   - $\langle 2, 4, 6 \rangle$
   - $\langle 1, 4, 7 \rangle$

- $\langle 1, 4, 6 \rangle$

(b) Die vorangehende Teilaufgabe liefert den Hinweis für den Algorithmus. Dieser besteht saus zwei Schritten. Um für eine Zahlensequenz X der Länge n eine monoton aufsteigende Teilsequenz maximaler Länge zu bestimmen, gilt es zunächst X zu sortieren. Das Ergebnis der Sortierung ist die Sequenz X' Im zweiten Schritt wird mittels des in der Vorlesung vorgestellten LCS-Algorithmus eine gemeinsame Teilsequenz maximaler Länge von X und X' bestimmt. Diese ist die gesuchte monoton aufsteigende Teilsequenz maximaler Länge von X.

5. Als Algorithmus mit dem gewünschten Laufzeitverhalten kann die in der Vorlesung beschriebene Tiefensuche verwendet werden. Um Zyklen zu erkennen, es es lediglich notwendig, daß der Algorithmus abbricht, sobald er einen Knoten über zwei verschieden Kanten erreicht. Dabei muß beachtet werden, daß jede Kannte in einem ungerichteten Graphen in zwei Adjazenslisten enthalten ist.

   Da in einem ungerichteten azyklischen Graphen maximal $|V| - 1$ Kanten vorhanden sind, müssen vom Graphen maximal $|V| - 1$ Kanten untersucht werden bevor ein Zyklus entdeckt wird. Die Laufzeit beträgt somit $\Theta(|V|)$.

6. - $S := \{a\}$

Knoten	d	π
a	0	NIL
b	1	a
c	2	a
d	3	a
e	∞	NIL
f	∞	NIL
g	∞	NIL

   - $S := \{a, b\}$

Knoten	d	π
a	0	NIL
b	1	a
c	2	a
d	3	a
e	2	b
f	∞	NIL
g	∞	NIL

   - $S := \{a, b, c\}$

Knoten	d	π
a	0	NIL
b	1	a
c	2	a
d	3	a
e	2	b
f	6	c
g	∞	NIL

- $S := \{a, b, c, e\}$

Knoten	d	π
a	0	NIL
b	1	a
c	2	a
d	3	a
e	2	b
f	6	c
g	5	e

- $S := \{a, b, c, e, d\}$

Knoten	d	π
a	0	NIL
b	1	a
c	2	a
d	3	a
e	2	b
f	6	c
g	4	d

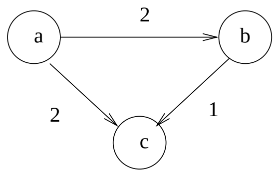
- $S := \{a, b, c, e, d, g\}$

Knoten	d	π
a	0	NIL
b	1	a
c	2	a
d	3	a
e	2	b
f	5	g
g	4	d

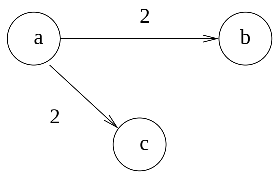
- $S := \{a, b, c, e, d, g, f\}$

Knoten	d	π
a	0	NIL
b	1	a
c	2	a
d	3	a
e	2	b
f	5	g
g	4	d

7. Gegenbeispiel:



Kürzeste-Wege-Baum des Beispielgraphen:



Der Graph ist aber kein MST! Der entsprechende MST des Beispielgraphen lautet: