

**Midterm Klausur zur Vorlesung Datenstrukturen und Datenorganisation**  
Prof. R. Bayer, TUM, 30.11.2001

**Bitte eintragen:** Name.....Vorname.....Matrikelnr.....

**Alle Hilfsmittel sind erlaubt!**

**Aufgabe 1: 9 Punkte**

Konstruieren Sie eine Klassenhierarchie nach dem KNF2 Verfahren für Kraftfahrzeuge unter Verwendung folgender Kurzbezeichnungen für Attribute:

- G    Höchstgeschwindigkeit
- H    Hersteller
- M    Fahrzeugmodell
- N    Nutzlast
- S    Anzahl der Sitze
- T    Treibstoff Art
- Z    Anzahl der Zylinder

Für die Anwendungen werden drei Klassen benötigt, nämlich **Lastwagen, PKW, Motorrad** mit folgenden Attributen:

<b>Lastwagen</b>	{ H, M, N, Z }
<b>PKW</b>	{ G, H, M, S, T }
<b>Motorrad</b>	{ G, H, M, Z }

Unterstreichen Sie die Attribute jeweils in ihrer Owner Klasse.

## Aufgabe 2:

Gegeben sei die Zeichenreihe mit darunter stehenden Positionsnummern:

<b>T</b>	<b>R</b>	<b>E</b>	<b>E</b>	<b>T</b>	<b>R</b>	<b>I</b>	<b>E</b>	<b>\$</b>
1	2	3	4	5	6	7	8	9

### Teilaufgabe 2.1: 6 Punkte

Konstruieren Sie den Positionsbaum mit der rechts-links Konstruktion

### Teilaufgabe 2.2: 2 Punkte

Zeichnen Sie in obiger Zeichenreihe Anfang und Ende aller Positionsidentifikatoren ein, um das Monotonielemma zu demonstrieren.

### Teilaufgabe 2.3: 2 Punkte

Der obige Text soll zu

<b>T</b>	<b>R</b>	<b>E</b>	<b>E</b>	<b>N</b>	<b>O</b>	<b>D</b>	<b>E</b>	<b>\$</b>
1	2	3	4	5	6	7	8	9

geändert werden.

Für welche Positionen müssen die Positionsidentifikatoren geändert werden?

(Sie sollen nur die Positionen angeben, aber nicht den Baum umkonstruieren!)

### Teilaufgabe 2.4: 4 Punkte

Konstruieren Sie den partiellen Positionsbaum, der bei der links-rechts Konstruktion entsteht, sobald die Teilzeichenreihe

**T R E E T R**

eingelezen wurde.

### Aufgabe 3:

Die natürlichen Zahlen sollen als Strings von Dezimalziffern dargestellt werden, intern sollen die einzelnen Dezimalziffern durch 4 Stellen binär codiert werden

In einem bestimmten Kontext (z.B. Modellbezeichnungen der Firmen BMW und Mercedes) sollen die folgenden Strings von Dezimalziffern mit den angegebenen Häufigkeiten vorkommen. Um Ihnen die Arbeit zu erleichtern sind auch die Binärcodes der Dezimalstrings angegeben.

Strings	Binärcodes	Häufigkeiten
„220“	0010 0010 0000	375
„318“	0011 0001 1000	1630
„420“	0100 0010 0000	890
„520“	0101 0010 0000	1257
„523“	0101 0010 0011	1705
„600“	0110 0000 0000	940
„730“	0111 0011 0000	624
„740“	0111 0100 0000	310
„850“	1000 0101 0000	40

#### Teilaufgabe 3.1: 7 Punkte

Konstruieren Sie den *digitalen Suchbaum* für die obigen Strings

#### Teilaufgabe 3.2: 5 Punkte

Konstruieren Sie den *sorted frequency Tree* für die obigen Strings

#### Teilaufgabe 3.3: 5 Punkte

Die Formel  $\sum f(k) \cdot h(k)$  beschreibt die Kosten eines Suchbaumes, wobei  $f(k)$  die Häufigkeit (frequency) und  $h(k)$  die Höhe (von der Wurzel aus gezählt) der Knoten  $k$  bedeuten und über  $k$  summiert wird. Welcher der beiden Bäume ist billiger und um welchen Betrag?