

## Musterklausur "Propädeutikum aus Informatik"

o. Univ.-Prof. Jörg R. Mühlbacher

1) Welche Zahl im Dual- bzw. im Hexadezimalsystem liefert die Dezimalzahl 206? Geben Sie auch den Rechengang an:

Dualsystem	11001110
Hexadezimalsystem	CE

Rechengang:

**Eine Binärzahl ist strukturell aufgebaut wie eine Dezimalzahl, allerdings wird ihr Wert nicht dargestellt aus Potenzen von 10, sondern aus Potenzen der Zahl 2:**

Die nullt-rechteste Binärziffer hat also das dezimale Gewicht  $2^0 = 1$   
 die erst-rechteste Binärziffer hat das dezimale Gewicht  $2^1 = 2$   
 die zweit-rechteste Binärziffer hat das dezimale Gewicht  $2^2 = 4$   
 die dritt-rechteste Binärziffer hat das Gewicht  $2^3 = 8$   
 die viert-rechteste Binärziffer hat das Gewicht  $2^4 = 16$   
 die fünft-rechteste Binärziffer hat das Gewicht  $2^5 = 32$   
 die sechst-rechteste Binärziffer hat das Gewicht  $2^6 = 64$   
 die siebt-rechteste Binärziffer hat das Gewicht  $2^7 = 128$   
 die acht-rechteste Binärziffer hat das Gewicht  $2^8 = 256$   
 die neunt-rechteste Binärziffer hat das Gewicht  $2^9 = 512$   
 die zehnt-rechteste Binärziffer hat das Gewicht  $2^{10} = 1024$   
 usw.

**Um z.B den dezimalen Wert der Binärzahl 11001110 zu berechnen geht man so vor:**

$$\begin{array}{r}
 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 = \\
 1 \cdot 128 + 1 \cdot 64 + 0 \cdot 32 + 0 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 0 \cdot 1 = \\
 128 \quad + 64 \quad \quad \quad + 8 \quad + 4 \quad + 2 \quad = 206
 \end{array}$$

**Umwandlung der Zahl 206 in eine Binärzahl:**

$2^7 = 128$	$206 - 128 = 78$	Binärzahl = 1....
$2^6 = 64$	$78 - 1 \cdot 64 = 14$	Binärzahl = 11...
$2^5 = 32$	$14 - 0 \cdot 32 = 14$	Binärzahl = 110....
$2^4 = 16$	$14 - 0 \cdot 16 = 14$	Binärzahl = 1100....
$2^3 = 8$	$14 - 1 \cdot 8 = 6$	Binärzahl = 11001...
$2^2 = 4$	$6 - 1 \cdot 4 = 2$	Binärzahl = 110011..
$2^1 = 2$	$2 - 1 \cdot 2 = 0$	Binärzahl = 1100111.
$2^0 = 1$	$0 - 0 \cdot 1 = 0$	Binärzahl = 11001110

Hexadezimalzahlumrechnung detto

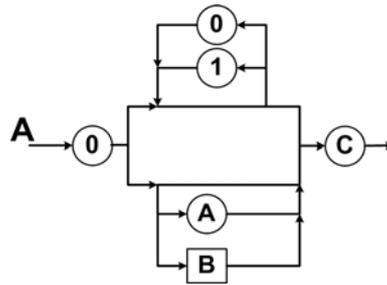
Achtung: Hier nicht zur Basis 2, sondern zur Basis 16. Die Zahlen 10-15 werden mit A-F bezeichnet.

2a) Erläutern Sie die EBNF-Regel  $A = "0" \{ ("0" | "1") \} | [ ("A" | B) ] "C"$ .  
 $B = "0"$ . durch Angabe von Beispielen, also  $A = ???????$

1	0C
2	000001111C
3	0AC

Falls die Regel keine weiteren Beispiele erlaubt, so streichen Sie bitte die noch leeren Boxen durch jeweils einen Querstrich.  
 Anmerkung: Die Leerzeichen in obiger Regel wurden aus Übersichtlichkeitsgründen mit angegeben.

2b) Zeichnen Sie das zur Regel A (aus A2a) zugehörige Syntaxdiagramm:

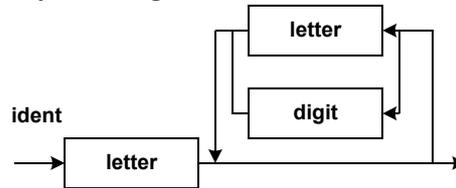


Wichtig:

“0“ und “C“ müssen in jeden Fall geschrieben werden. Dazwischen entweder der erste Zweig {“0“|“1“} oder der zweite Zweig [“A“|B]. Der erste Zweig wird entweder leer durchlaufen oder **beliebig oft** zwischen “0“ und “1“ ausgewählt. (Rücklauf immer in dieser Form darstellen!) Der zweite Zweig wird **genau einmal** durchlaufen und dabei entweder nichts oder “A“ oder B ausgewählt.

Terminalsymbole werden in Kreisen dargestellt, Nonterminalsymbole in Rechtecken.

3) Gegeben ist das folgende Syntaxdiagramm. Geben Sie die äquivalente EBNF dazu an.



Antwort:

$ident = letter\{letter|digit\}.$

4) Erklären Sie folgende EBNF-Konstrukte:  $(A|B)$ ,  $[A]$ ,  $\{A\}$

Antwort:

$(A B)$	Entweder einmal A oder alternativ einmal B
$[A]$	Entweder einmal A oder nullmal A, $A \rightarrow A / \epsilon$
$\{A\}$	Konkatenation beliebig vieler A's, $A \rightarrow \epsilon / A / AA / AAA / \dots$

5) Stellen Sie die Zahl  $-28$  im Zweierkomplement dar (Länge der Dualzahl: genau 8 Bit) und geben Sie den Rechengang an:

Antwort:

Dezimalzahl in Binärzahl umrechnen: 28: 0001 1100  
 Invertieren: 1110 0011  
 +1: 0000 0001  
Ergibt  $-28$  im Zweierkomplement: 1110 0100

## Propädeutikum Musterklausur

6) Ergänzen Sie die folgenden Sätze:

Eine endliche Folge von Zeichen aus einem Alphabet $A$ heißt <u>Wort über dem Alphabet</u>
Jedes Wort aus $A^*$ ist <u>endlich</u> , hat also eine <u>endliche</u> Länge. (endlich oder unendlich?)
Die Menge $A^*$ selbst ist eine <u>unendliche</u> Menge. (endlich oder unendlich?)
Man gibt eine Syntax $G$ der Sprache $L$ an, um entscheiden zu können, ob ein vorgelegtes Wort $W$ (bzw. Folge von Wörtern = Satz) zur Sprache gehört oder nicht. Der terminus technicus dafür lautet: <u><math>W</math> ist aus <math>L</math> ableitbar.</u>

Wichtig: Wort **über** dem Alphabet!!

7) Gegeben sei  $W = \{\text{Kind, Eltern, Buch, liest, hat}\}$  und

- 1)  $\langle \text{Satz} \rangle ::= \langle \text{Subjekt} \rangle \langle \text{Verb} \rangle \langle \text{Objekt} \rangle$
- 2)  $\langle \text{Objekt} \rangle ::= \langle \text{Hauptwort} \rangle$
- 3)  $\langle \text{Subjekt} \rangle ::= \langle \text{Hauptwort} \rangle$
- 4)  $\langle \text{Verb} \rangle ::= \text{liest} \mid \text{hat}$
- 5)  $\langle \text{Hauptwort} \rangle ::= \text{Buch} \mid \text{Kind} \mid \text{Eltern}$

Lassen sich folgende Sätze daraus ableiten?

Richtig JA	Falsch NEIN	Weiß nicht	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Eltern haben Kind
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Kind liest Buch
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Kind hat Eltern
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Eltern liest ein Buch
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Buch liest Buch
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Ein Kind hat ein Buch

Erläuterung:

Eltern haben Kind: haben ist nicht als Verb angeführt.

Eltern liest ein Buch: ein ist in der Konstruktionsvorschrift nicht angeführt.

Ein Kind hat ein Buch: In der Konstruktionsvorschrift kein Artikel für ein Subjekt und kein Artikel für ein Objekt vorgesehen.

8) Wie hoch ist der Informationsgehalt der Nachricht "Er hat eine 6 gewürfelt und dann eine 1 gewürfelt" nach 2 Würfelwürfen? (Vergessen Sie nicht, die Einheit anzugeben, zB: sagt man etwas ist 5 kg schwer!)

Antwort:

$p = 1/36$ ,  $I = -\log_2(1/36) = \log_2(36) = 5,17$  bit

Beim ersten Wurf ist die Wahrscheinlichkeit ( $p_1$ ) eine 6 zu würfeln  $1/6$  (eine aus 6 möglichen Ausgängen), beim zweiten Wurf ist die Wahrscheinlichkeit ( $p_2$ ) eine 1 zu würfeln ebenfalls  $1/6$ .  $p = p_1 \cdot p_2$ ,  $p = 1/6 \cdot 1/6 = 1/36$ .

Die Einheit des Informationsgehalts ist **bit**.

## Propädeutikum Musterklausur

9) Schreiben Sie die Wahrheitstabellen für folgende Ausdrücke an:

$a := (x \text{ OR } z)$ ,  $b := \text{NOT } y$ ,  $c := a \text{ OR } (b \text{ OR } c)$

x	z	a:=(x OR z)	y	b:= NOT y
0	0	0	0	1
0	1	1	1	0
1	0	1		
1	1	1		

a	b	c	a OR (b OR c)
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

10) Was versteht man unter einem Modell?

Antwort:

Unter einem Modell versteht man ein Abbild der Wirklichkeit.

Oft ist etwas zu komplex, um es real vollständig zu erfassen und zu untersuchen. Ein Modell entsteht durch Abstraktion und es findet eine Reduktion auf das Wesentliche statt. Aus der Modellbildung folgt eine Verminderung der Komplexität. Das Modell ist leichter handhabbar als die Realität. Größenverhältnisse sind wichtig, da diese die Genauigkeit des Abbilds widerspiegeln. Die Darstellung erfolgt meist in symbolischer Form. Beispiel für ein Modell: Globus

11) Geben Sie eine Beschreibung des Halteproblems an? Wenn Sie eine Beweisskizze angeben, bekommen Sie Sonderpunkte.

Antwort:

Es soll ein universelles Testprogramm T geben, welches jedes **beliebige Programm P** und **jeden beliebigen Input I** dazu als Eingabe erlaubt:  $T(P, I)$  und nach endlich vielen Schritten:

(i) stehen bleibt

(ii) sagt, ob P für die Eingabe I stehen bleibt oder nicht

So ein universelles Testprogramm T gibt es nicht! Das Halteproblem ist nicht lösbar, ist nicht berechenbar, es gibt so einen Algorithmus nicht!

Beweis:

Unter Verwendung des Algorithmus  $T(P, I)$  können wir einen anderen Algorithmus  $X(P, I)$  schreiben, der wie folgt aussieht:

```
X: WHILE T(P, D) DO beep END;
```

Wenn  $T(P, I)$  ein Algorithmus wäre, dann wäre auch X ein Algorithmus. Die Daten, die an X übergeben werden, könnte jeder beliebige Text eines Programms zusammen mit seinen Eingabedaten I sein. Daher können wir X sich selbst testen lassen: wir benutzen den Text von X als Eingabe P und haben somit formal  $X(X, I)$ .

Wenn  $T(X, I)$  TRUE liefert, sprich X anhält, dann würde dies `WHILE TRUE DO beep END` ergeben und X würde nie anhalten und für immer „beepen“. Dies ist ein Widerspruch zur zugrundeliegenden Annahme, dass ein Prüfalgorithmus  $T(P, I)$  existiert. Der gleiche Widerspruch wird erzielt, wenn  $T(X, I)$  FALSE liefert, sprich X nicht anhält. In diesem Fall ergibt sich `WHILE FALSE DO beep END` und X würde sofort anhalten.

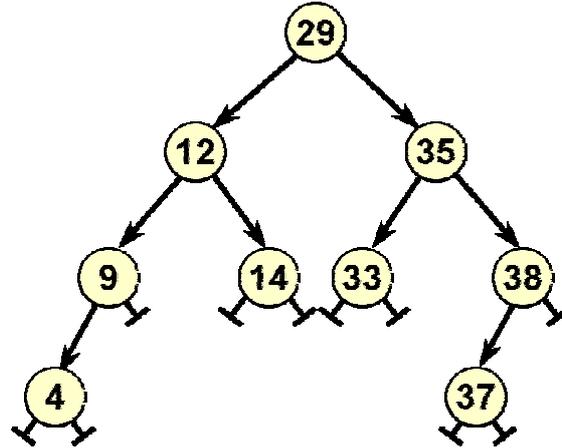
**Propädeutikum Musterklausur**

12) Was versteht man unter einem Prozess?

Antwort:

Ein Prozess ist eine kontrollierte Folge von Zustandsänderungen.  
Jeder Prozess ist ein in sich geschlossener Vorgang.

13) Welche Sequenz ergibt sich bei einem Pre-/In- und Postorder-Durchlauf des folgenden Baumes:



Preorder	29, 12, 9, 4, 14, 35, 33, 38, 37
Inorder	4, 9, 12, 14, 29, 33, 35, 37, 38
Postorder	4, 9, 14, 12, 33, 37, 38, 35, 29

Anmerkung:

PreOrder, Hauptreihenfolge: Wurzel, linker Unterbaum, rechter Unterbaum  
InOrder, symmetrischer Durchlauf: linker Unterbaum, Wurzel, rechter Unterbaum  
PostOrder, Nebenreihenfolge: linker Unterbaum, rechter Unterbaum, Wurzel