

Vorname

Nachname

Matrikel-Nr.

Diese Klausur besteht aus **5 Aufgaben**. Schreiben Sie jede Lösung auf die Vorderseite eines *neuen* Blattes (und lassen Sie die Rückseiten Ihrer Lösungsblätter *leer*).

Notation für Grammatiken: Fassen Sie *Endsymbole* in doppelte Anführungszeichen ein, um sie von den Zwischensymbolen zu unterscheiden. Benutzen Sie einen Doppelpunkt `:` als Trennsymbol (zwischen der linken und rechten Seite einer Regel, wie in Gentle).

Aufgabe 1 (20 Punkte): Geben Sie eine (kontextfreie, Typ-2-) Grammatik an für die Menge aller Zahlen im **4-er-System**, die (ohne Rest) **durch 5 teilbar** sind. Startsymbol: $RK0$ (wie "Restklasse 0").

Aufgabe 2 (20 Punkte): Betrachten Sie die folgende Grammatik für Ausdrücke (Endsymbole sind in doppelte Anführungszeichen eingefaßt, z.B. "\$1" oder "c". AUS1 ist das Startsymbol):

AUS1 : AUS2 "\$1" AUS1

AUS1 : AUS2

AUS2 : AUS2 "\$2" AUS3

AUS2 : AUS3

AUS3 : "\$3" AUS3

AUS3 : AUS4

AUS4 : "\$4" AUS5

AUS4 : AUS5

AUS5 : LIT

AUS5 : "(" AUS1 ") "

LIT : "a"

LIT : "b"

LIT : "c"

In dieser Grammatik kommen die Operatoren $\$1$, $\$2$, $\$3$, $\$4$ vor. Geben Sie als Antworten auf die folgenden Fragen die zutreffenden Operatoren an:

2.01. Alle zweistelligen Operatoren?
2.02. Alle einstelligen Operatoren?
2.03. Alle linksassoziativen Operatoren?
2.04. Alle rechtsassoziativen Operatoren?
2.05. Welcher zweistellige Operator bindet am schwächsten?
2.06. Welcher einstellige Operator bindet am stärksten?

Welche der folgenden Worte kan man aus obiger Grammatik ableiten (JA) und welche nicht (NEIN)?

2.07. $\$3$ $\$3$ c
2.08. $\$4$ $\$4$ a
2.09. a $\$2$ b $\$1$ c
2.10. c $\$1$ c $\$2$ a

Aufgabe 3 (20 Punkte): Betrachten Sie folgende Vereinbarung eines (Gentle-) Typs:

```
'type' LISTE
  leer
  list(Elem: INT, Rest: LISTE)
```

3.1. Schreiben Sie ein Prädikat entsprechend der folgenden Spezifikation:

```
'condition' keineZwillinge(L: LISTE)
  -- Gelingt, wenn L keinen Zwilling (2 gleiche, benachbarte Elemente) enthält.
  -- Beispiele:
  -- Die Liste list(5, list(6, list(6, list(3, leer)))) enthält einen Zwilling
  -- Die Liste list(6, list(5, list(6, list(3, leer)))) enthält keinen Zwilling
```

3.2. Schreiben Sie ein Prädikat entsprechend der folgenden Spezifikation:

```
'action' mod(Dend: INT, Dor: INT -> Rest: INT)
  -- Berechnet den Rest, der sich nach der Ganzzahldivision von
  -- Dend durch Dor ergibt. Beispiel: Nach dem Aufruf
  -- mod(13, 5 -> R) hat R den Wert 3 (denn 13/5 ist gleich 2 Rest 3)
```

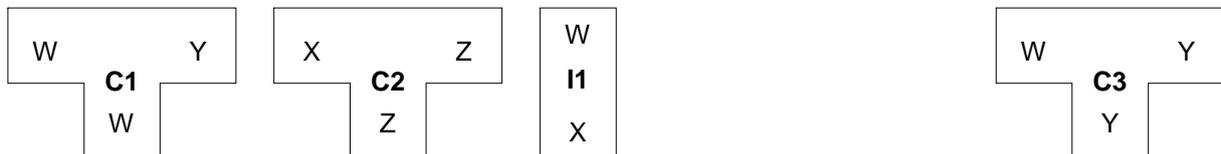
3.3. Schreiben Sie ein Prädikat entsprechend der folgenden Spezifikation:

```
'action' dieUngeraden(Rein: LISTE -> Raus: LISTE)
  -- Raus enthaelt alle ungeraden Zahlen aus Rein. Beispiel:
  -- Nach dem Aufruf dieUngeraden(list(5, list(2, list(7, leer)))-> L)
  -- hat L den Wert list(7, list(5, leer)).
```

Aufgabe 4 (20 Punkte)

Angenommen, Sie haben (nur) eine **Z-Maschine** und die drei Programme C1, C2 und I1:

Sie hätten gerne das Programm C3:



Wie kann man C3 mit Hilfe von C1, C2 und I1 erzeugen? Beschreiben Sie die nötigen (und möglichen) Übersetzungsschritte durch geeignete Diagramme.

Aufgabe 5 (20 Punkte): Beantworten Sie die folgenden Fragen möglichst *kurz*, aber *präzise* und verwenden Sie dabei möglichst die in der Lehrveranstaltung eingeführten *Fachbegriffe*.

5.1. Was ist eine *Satzform* (im Zusammenhang mit einer Grammatik)?

5.2. Welche Bedingungen (oder Einschränkungen) muss eine Chomsky-Regel $LS : RS$ erfüllen, damit sie *rechtslinear* ist?

5.3. Welche Bedingung (oder Einschränkung) muss eine Chomsky-Regel $LS : RS$ erfüllen, damit sie *in einer Typ-1-Grammatik erlaubt* ist?

5.5. Die kontextfreie Grammatik der Programmiersprache Java beschreibt eine bestimmte formale Sprache S . Was hat diese Sprache S mit der Sprache Java zu tun? Ist S gleich Java? Oder ist S eine Teilmenge von Java? Oder ist Java eine Teilmenge von S ? Oder ist keine dieser beiden Sprachen eine Teilmenge der anderen?

5.5. Welchen Wert hat der Ausdruck $2 ** 2 ** 3$ wenn der Potenzierungs-Operator $**$ als *rechtsassoziativer* Operator bzw. als *linksassoziativer* Operator definiert wurde. Geben Sie als Antwort *zwei konkrete Zahlen* (keine "Rechenaufgaben") an.

Der Gentle-Typ BAUM sei wie folgt definiert:

```
'type' BAUM
  leer
  b(INT, BAUM, BAUM)
```

5.6. Geben Sie drei *Grundterme* des Typs BAUM an. In einem dieser Grundterme soll das INT-Literal 17 vorkommen und in einem anderen Grundterm sollen die INT-Literale 25 und 33 vorkommen.

5.7. Geben Sie einen *Term* des Typs BAUM an, der als Muster auf *alle nicht-leeren Bäume* passt.

5.8. Geben Sie einen *Term* des Typs BAUM an, der als Muster auf *alle Bäume* passt.

5.9. Wie heißt der Jasmin-Befehl (Java-Assembler-Befehl), mit dem man eine 8-Bit-Ganzzahl auf den Stapel laden kann? Ebenso für 16-Bit-Ganzzahlen.

5.10. Wie viele Bytes belegt der Befehl `aload_3` und was macht oder bewirkt er?

Beurteilung der Klausur:

Punkte	
Aufgabe 1:	Note:
Aufgabe 2:	Datum:
Aufgabe 3:	
Aufgabe 4:	
Aufgabe 5:	
Summe:	

Korrigierte Beurteilung der Klausur:

Punkte	
Aufgabe 1:	Note:
Aufgabe 2:	Datum:
Aufgabe 3:	
Aufgabe 4:	
Aufgabe 5:	
Summe:	

Lösung 1 (20 Punkte): Geben Sie eine (kontextfreie, Typ-2-) Grammatik an für die Menge aller Zahlen im **4-er-System**, die (ohne Rest) **durch 5 teilbar** sind. Startsymbol RK0 (wie "Restklasse 0").

R01: RK0 : "0"	R09: RK4 : RK1 "0"	R17: RK2 : RK3 "0"
R02: RK1 : "1"	R10: RK0 : RK1 "1"	R18: RK3 : RK3 "1"
R03: RK2 : "2"	R11: RK1 : RK1 "2"	R19: RK4 : RK3 "2"
R04: RK3 : "3"	R12: RK2 : RK1 "3"	R20: RK0 : RK3 "3"
R05: RK0 : RK0 "0"	R13: RK3 : RK2 "0"	R21: RK1 : RK4 "0"
R06: RK1 : RK0 "1"	R14: RK4 : RK2 "1"	R22: RK2 : RK4 "1"
R07: RK2 : RK0 "2"	R15: RK0 : RK2 "2"	R23: RK3 : RK4 "2"
R08: RK3 : RK0 "3"	R16: RK1 : RK2 "3"	R24: RK4 : RK4 "3"

Lösung 2 (30 Punkte): Betrachten Sie die folgende Grammatik für Ausdrücke (Endsymbole sind in doppelte Anführungszeichen eingefaßt, z.B. "\$1" oder "c". AUS1 ist das Startsymbol):

AUS1 : AUS2 "\$1" AUS1
AUS1 : AUS2

AUS2 : AUS2 "\$2" AUS3
AUS2 : AUS3

AUS3 : "\$3" AUS3
AUS3 : AUS4

AUS4 : "\$4" AUS5
AUS4 : AUS5

AUS5 : LIT
AUS5 : "(" AUS1 ")"

LIT : "a"
LIT : "b"
LIT : "c"

In dieser Grammatik kommen die Operatoren \$1, \$2, \$3, \$4 vor. Geben Sie als Antworten auf die folgenden Fragen die zutreffenden Operatoren an:

- | | |
|---|-----------------|
| 2.01. Alle zweistelligen Operatoren? | \$1, \$2 |
| 2.02. Alle einstelligen Operatoren? | \$3, \$4 |
| 2.03. Alle linksassoziativen Operatoren? | \$2 |
| 2.04. Alle rechtsassoziativen Operatoren? | \$1 |
| 2.05. Welcher zweistellige Operator bindet am schwächstens? | \$1 |
| 2.06. Welcher einstellige Operator bindet am stärksten? | \$4 |

Welche der folgenden Worte kan man aus obiger Grammatik ableiten (JA) und welche nicht (NEIN)?

- | | |
|---------------------|-------------|
| 2.07. \$3 \$3 c | JA |
| 2.08. \$4 \$4 a | NEIN |
| 2.09. a \$2 b \$1 c | JA |
| 2.10. c \$1 c \$2 a | JA |

Lösung 3 (20 Punkte): Betrachten Sie folgende Vereinbarung eines (Gentle-) Typs:

```
'type' LISTE
  leer
  list(Elem: INT, Rest: LISTE)
```

3.1. Schreiben Sie ein Prädikat entsprechend der folgenden Spezifikation:

```
'condition' keineZwillinge(L: LISTE)
  -- Gelingt, wenn L keinen Zwilling (2 gleiche, benachbarte Elemente) enthält.
  -- Beispiele:
  -- Die Liste list(5, list(6, list(6, list(3, leer)))) enthält einen Zwilling
  -- Die Liste list(6, list(5, list(6, list(3, leer)))) enthält keinen Zwilling

  'rule' keineZwillinge(leer): .
  'rule' keineZwillinge(list(N, leer)): .
  'rule' keineZwillinge(list(N1, list(N2, L2))):
    ne(N1, N2)
    keineZwillinge(list(N2, L2))
```

3.2. Schreiben Sie ein Prädikat entsprechend der folgenden Spezifikation:

```
'action' mod(Dend: INT, Dor: INT -> Rest: INT)
  -- Berechnet den Rest, der sich nach der Ganzzahldivision von
  -- Dend durch Dor ergibt. Beispiel: Nach dem Aufruf
  -- mod(13, 5 -> R) hat R den Wert 3 (denn 13/5 ist gleich 2 Rest 3)

  'rule' mod(DEND, DOR -> DEND - (DEND / DOR) * DOR): .
```

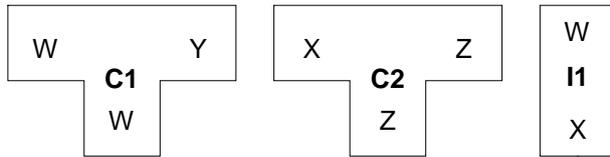
3.3. Schreiben Sie ein Prädikat entsprechend der folgenden Spezifikation:

```
'action' dieUngeraden(Rein: LISTE -> Raus: LISTE)
  -- Raus enthaelt alle ungeraden Zahlen aus Rein. Beispiel:
  -- Nach dem Aufruf dieUngeraden(list(5, list(2, list(7, leer)))-> L)
  -- hat L den Wert list(7, list(5, leer)).

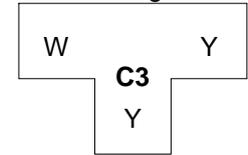
  'rule' dieUngeraden(leer -> leer): .
  'rule' dieUngeraden(list(N, L) -> list(N, DUL))):
    mod(N, 2 -> REST)
    ne(REST, 0)
    dieUngeraden(L -> DUL)
  'rule' dieUngeraden(list(N, L) -> DUL):
    dieUngeraden(L -> DUL)
```

Lösung 4 (20 Punkte)

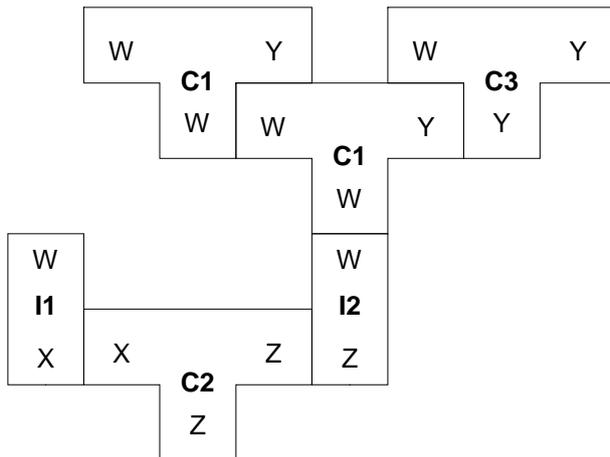
Wir haben C1, C2, I1:



Wir hätten gern C3:



So kann man C3 mit Hilfe von C1, C2 und I1 erzeugen:



Lösung 5 (20 Punkte): Beantworten Sie die folgenden Fragen möglichst kurz, aber präzise und verwenden Sie dabei möglichst die in der Lehrveranstaltung eingeführten Fachbegriffe.

5.1. Was ist eine *Satzform* (im Zusammenhang mit einer Grammatik)?

Eine (endliche, leere oder nicht-leere) Folge von Zwischensymbolen und/oder Endsymbolen.

5.2. Welche Bedingungen (oder Einschränkungen) muss eine Chomsky-Regel $LS \xrightarrow{c} RS$ erfüllen, damit sie *rechtslinear* ist?

LS muss aus genau einem Zwischensymbol bestehen.

RS muss mit einem Zwischensymbol enden und darf ansonsten nur Endsymbole enthalten.

5.3. Welche Bedingung (oder Einschränkung) muss eine Chomsky-Regel $LS \xrightarrow{c} RS$ erfüllen, damit sie *in einer Typ-1-Grammatik erlaubt* ist?

RS darf nicht kürzer sein als LS.

5.5. Die kontextfreie Grammatik der Programmiersprache Java beschreibt eine bestimmte formale Sprache S. Was hat diese Sprache S mit der Sprache Java zu tun? Ist S gleich Java? Oder ist S eine Teilmenge von Java? Oder ist Java eine Teilmenge von S? Oder ist keine dieser beiden Sprachen eine Teilmenge der anderen?

Java ist eine Teilmenge von S.

5.5. Welchen Wert hat der Ausdruck $2^{**} 2^{**} 3$ wenn der Potenzierungs-Operator $**$ als *rechtsassoziativer* Operator bzw.

als *linksassoziativer* Operator

definiert wurde. Geben Sie als Antwort *zwei konkrete Zahlen* (keine "Rechenaufgaben") an.

Rechtsassoziativ: (2 ** 8 gleich) 256

Linksassoziativ: (4 ** 3 gleich) 64

Der Gentle-Typ BAUM sei wie folgt definiert:

```
'type' BAUM
  leer
  b(INT, BAUM, BAUM)
```

5.6. Geben Sie drei *Grundterme* des Typs BAUM an. In einem dieser Grundterme soll das INT-Literal 17 vorkommen und in einem anderen Grundterm sollen die INT-Literale 25 und 33 vorkommen.

z.B. leer, b(17, leer, leer), b(25, b(33, leer, leer), leer)

5.7. Geben Sie einen *Term* des Typs BAUM an, der als Muster auf alle nicht-leeren Bäume passt.

b(N, B1, B2)

5.8. Geben Sie einen *Term* des Typs BAUM an, der als Muster auf alle Bäume passt.

B

5.9. Wie heißt der Jasmin-Befehl (Java-Assembler-Befehl), mit dem man eine 8-Bit-Ganzzahl auf den Stapel laden kann? Ebenso für 16-Bit-Ganzzahlen.

bipush, sipush

5.10. Wie viele Bytes belegt der Befehl `aload_3` und was macht oder bewirkt er?

Er belegt 1 Byte. Er lädt den Wert der Variablen 3 auf den Stapel. Dieser Wert muss eine Referenz sein.