

Klausur „Compilerbau“ – 29. September 2014

Bitte bearbeiten Sie die Aufgaben möglichst auf den Aufgabenblättern. Die Benutzung von Unterlagen ist nicht erlaubt. Die Bearbeitungszeit beträgt 90 Minuten.

Nachname: _____

Vorname: _____

Matrikelnummer: _____

Klausurvoraussetzung erbracht im Semester (Zutreffendes ankreuzen):

SS 14 WS 13/14 SS 13 Sonstiges

Unterschrift: _____

Aufgabe	Punktzahl maximal	Punktzahl erreicht
1	12	
2	6	
3	9	
4	6	
5	10	
6	10	
Summe	53	

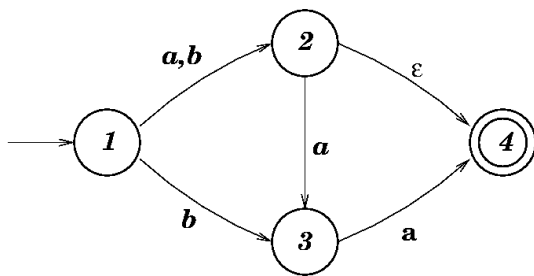
Aufgabe 1 (3+3+3+3)

- a) Was versteht man im Compilerbau unter einem „Parser“? Welche Funktion hat er? Wie arbeitet er mit anderen Compilerkomponenten zusammen?
- b) Was bedeutet im Hinblick auf typisierte Programmiersprachen der Begriff „Überladung“? Geben Sie ein Beispiel an. Wie geht ein Compiler damit um?
- c) Was versteht man unter einer regulären Sprache? Welche Bedeutung haben reguläre Sprachen im Compilerbau?
- d) Ein Ausdruck ist ein syntaktisches Element, das im Quelltext einen Wert repräsentiert. Wie verarbeitet ein Compiler typischerweise einen Ausdruck (Frontend und Backend)?

Aufgabe 2 (3+3 Punkte)

a) Geben Sie das Zustandsdiagramm eines endlichen Automaten für die Sprache an, die durch den regulären Ausdruck $(a|bb|\epsilon)^*c$ repräsentiert wird. Der Akzeptor kann nichtdeterministisch sein.

b) Geben Sie zu dem nachfolgenden nichtdeterministischen endlichen Automaten einen äquivalenten deterministischen endlichen Automaten (Zustandsdiagramm) an.



Aufgabe 3 (3+3+3 Punkte)

Gegeben sei folgende kontextfreie Grammatik G

$$S \rightarrow AB \mid CC$$

$$A \rightarrow AB \mid B$$

$$B \rightarrow bb$$

$$C \rightarrow A \mid c$$

a) Geben Sie eine Linksableitung und den Ableitungsbaum zu $bbbcb$ an.

b) Geben Sie alle Wörter an, die sich zusätzlich ableiten lassen, wenn man die Regel $C \rightarrow \epsilon$ hinzufügt.

c) Beweisen Sie, dass die Grammatik durch die zusätzliche Regel $C \rightarrow \epsilon$ mehrdeutig wird.

Aufgabe 4 (2+2+2 Punkte)

Betrachten Sie folgende Grammatik

$$\begin{aligned} S &\rightarrow ABC \\ A &\rightarrow aabA \mid Cc \mid \epsilon \\ B &\rightarrow bcBd \mid \epsilon \\ C &\rightarrow dC \mid \epsilon \end{aligned}$$

- a) Bestimmen Sie $FIRST(A)$:
- b) Bestimmen Sie $FOLLOW(B)$:
- c) Was steht in der LL(1)-Parsertabelle in dem Eintrag zu B und d ?

Aufgabe 5 (4+3+3 Punkte)

- a) Bestimmen Sie zur nachfolgenden Grammatik die LR(0)-Elemente und die Übergänge im zugehörigen DEA.
- b) Geben Sie die SLR(1)-Parsertabelle dazu an.
- c) Geben Sie die Berechnung des SLR(1)-Parsers für die Eingabe b an.

- (1) $S \rightarrow SA$
- (2) $S \rightarrow \epsilon$
- (3) $A \rightarrow aa$
- (4) $A \rightarrow b$

Aufgabe 6 (3 + 7 Punkte)

- a) Bestimmen Sie zur SPL-Prozedur `p` das Frame-Layout für den Aktivierungsrahmen: Bestandteile in der richtigen Reihenfolge mit Offsets zum Framepointer und Größen in Bytes
- b) Bestimmen Sie den ECO32-Assemblercode zu `p`. Die Prozedur `printi` erwartet einen Wertparameter vom Typ `int`. (SP=\$29, FP=\$25, RET=\$31, verfügbare Register: \$8-\$15).

```
type paartyp = array [2] of int;
```

```
proc p (ref i:int, j:int) {  
  var paar: paartyp;  
  paar[j] := i;  
  printi(i);  
}
```