

Aufgaben zur Klausur **Compilerbau** im SS 2010 (BInf 251, BInf 252, II h769)

Zeit: 75 Minuten

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den gekennzeichneten Stellen in das Aufgabenblatt ein. Ist ihre Lösung wesentlich umfangreicher, so überprüfen Sie bitte nochmals Ihren Lösungsweg.

Sollten Unklarheiten oder Mehrdeutigkeiten bei der Aufgabenstellung auftreten, so notieren Sie bitte, wie Sie die Aufgabe interpretiert haben.

Viel Erfolg!

Diese Klausur besteht einschließlich dieses Deckblattes aus 6 Seiten.

---

**Aufgabe 1:**

Gegeben sei eine kontextfreie Grammatik  $G = (T, N, P, S)$

1. Definieren Sie  $nullable(X)$  für  $x \in (N \cup T)$ .

$nullable(X) = \dots\dots\dots$

2. Definieren Sie  $FIRST(w)$  mit  $w \in (N \cup T)^*$ .

$FIRST(w) = \dots\dots\dots$

3. Definieren Sie  $FOLLOW(X)$  mit  $x \in (N \cup T)$ .

$FOLLOW(X) = \dots\dots\dots$

$\dots\dots\dots$

4. In welche Zellen einer LL(1) Parsertabelle wird eine Regel  $X ::= w$  aus  $P$  auf Grund der  $FIRST$ -Mengen eingetragen?

$\dots\dots\dots$

5. In welche Zellen einer LL(1) Parsertabelle wird eine Regel  $X ::= w$  aus  $P$  auf Grund der  $FOLLOW$ -Mengen eingetragen?

$\dots\dots\dots$

$\dots\dots\dots$

6. Welche Eigenschaft muss eine LL-Parsertabelle besitzen, damit sie überhaupt in einem Parser genutzt werden kann?

$\dots\dots\dots$

7. Nennen Sie drei Grammatik-Eigenschaften, die die LL(1)-Eigenschaft einer kontextfreien Grammatik zerstören.

1)  $\dots\dots\dots$

2)  $\dots\dots\dots$

3)  $\dots\dots\dots$

**Aufgabe 2:**

Die Implementierung einer Programmiersprache mittels eines Interpreters für eine virtuelle Maschine gelten allgemein als wenig laufzeiteffizient. Perl, Tcl, Javascript und andere Skriptsprachen sind aber gerade mit dieser Technik realisiert und zwar auf äußerst effiziente Art. Warum trifft das Vorurteil, Interpretierer sind ineffizient, bei diesen Sprachen nicht zu.

.....

.....

.....

.....

.....



**Aufgabe 3:**

Konstruieren Sie einen regulären Ausdruck zum Erkennen von Kommentaren in C (`/*...*/`).

.....

.....

.....

Konstruieren Sie einen deterministischen endlichen Automaten, der C-Kommentare akzeptiert.

**Das Zustands-Übergangs-Diagramm:**



**Aufgabe 5:**

Transformieren sie den regulären Ausdruck  $((a * |b)c?)^+$  **gemäß dem Transformationsschema aus der Vorlesung** in einen nichtdeterministischen endlichen Automaten. Das zu Grunde liegende Alphabet sei dabei  $I = \{a, b, c\}$ .

Die Transformaton erfolgt schrittweise. Nutzen Sie für die Zwischenschritte die Rückseiten der Klausur.

Der fertige Automat als Zustandsübergangsdiagramm:

Konstruieren Sie den minimalen deterministischen Automaten für den obigen Ausdruck als Zustandsübergangsdiagramm: