

7. Übung „Übersetzerbau“

Abgabe am 6. Juni in der Vorlesung

Aufgabe 25

7 Punkte

Ein Satz der Vorlesung stellt die unterschiedlichen Grammatikklassen in Beziehung. Beweisen Sie folgende Teilaspekte dieses Satzes:

- Zeigen Sie, dass jede $SLR(1)$ -Grammatik auch eine $LALR(1)$ -Grammatik ist. Damit ist mit Aufgabe 23 die echte Inklusion der Grammatikklassen (Punkt 3 des Satzes) gezeigt.
- Zeigen Sie, dass $LALR(1) \subsetneq LR(1)$ gilt.

Aufgabe 26

13 Punkte

In dieser Aufgabe sollen Sie einen Interpreter für die Programmiersprache MPS implementieren. Hierbei sei MPS wie in Aufgabe 17 (also mit Deklarationen getrennt durch Kommas) definiert. Außerdem seien die Statements um eine Print-Anweisung erweitert: $Stm \rightarrow \underline{\text{print}}(\underline{\text{Expr}})$ wobei $\underline{\text{print}},(\underline{\quad})$ Terminale seien und $\underline{\text{Expr}}$ das Nichtterminal zur Erzeugung von Ausdrücken sei.

Gehen Sie in folgenden Arbeitsschritten vor:

- Überarbeiten Sie den Scanner für MPS, so dass er auch die neuen Sprachkonstrukte von MPS abdeckt.
- Implementieren Sie aufbauend auf dem Scanner einen Happy-Parser für MPS. Das Happy-Tool ist auf den Suns unter dem Nutzer `haskell` eingerichtet und kann mit `/home/haskell/bin/happy datei.y` gestartet werden. Die Happy-Datei aus der Vorlesung finden Sie auf der Web-Seite zur Übung.
- Erweitern Sie Ihren Parser um semantische Aktionen zum Interpretieren von MPS. Gehen Sie hierbei analog zu dem in der Vorlesung vorgestellten Interpreter für SIMPLE mit print-Anweisungen vor. Alle Ausgaben des Programms sollen in einer Zahlenliste (`[Int]`) gespeichert werden und schließlich der Rückgabewert des Programms sein.