

3.1 Java-Statement-Kombinatorparser

Sei folgende Grammatik für einen Ausschnitt der Java-Statements gegeben:

statement	→	ifthenstatement ifthenelsestatement whilestatement block emptystatement returnstatement
ifthenstatement	→	IF LBRACE bexpression RBRACE statement
ifthenelsestatement	→	IF LBRACE bexpression RBRACE statement ELSE statement
whilestatement	→	WHILE LBRACE bexpression RBRACE statement
block	→	LBRACKET RBRACKET LBRACKET statements RBRACKET
statements	→	statement statements statement
emptystatement	→	SEMICOLON
returnstatement	→	RETURN SEMICOLON RETURN expression SEMICOLON
bexpression	→	TRUE
expression	→	1

- Definieren Sie eine abstrakte Syntax für die Menge von Java-Statements
- Schreiben Sie einen Kombinatorparser für die Menge Java-Statements, der die Eingabe in abstrakte Syntax umwandelt. Nutzen Sie den Scanner aus Aufgabe 2.4

3.2 Happy-Parser

Schreiben Sie für die Statements aus Aufgabe 3.1 einen Happy-Parser.

3.3 Abstrakte Syntax für Statement

Ergänzen Sie das Happy-File aus Aufgabe 3.2 um Anweisungen, die sie abstrakte Syntax für Statemnts aufbaut.