

## Compilerbau - Wintersemester 2020/21

### Praktisches Übungsblatt 4

Besprechung der Aufgaben am 19.11.21 um 16:30 Uhr in 25.12.02.55 und gleichzeitig online per BBB  
(evtl. früher)

Fragen an Lukas.Lang@hhu.de

Die Bearbeitung ist freiwillig.

Bringen Sie Ihre Lösung mit zum Übungstermin

#### Aufgabe 4.1

Implementieren Sie in Java eine LL1 Parsingtable sowie einen LL1 Parser. Hierzu wird Ihnen ein JUnit Testcase gestellt.

1. Implementieren Sie dazu zuerst folgendes Interface:

```
import java.util.Map.Entry;
import java.util.List;
import java.util.Map;

public interface ILL1ParsingTable {
    public List<String> get(String nonterminal, String terminal);

    public String getStartSymbol();

    public List<String> getNonterminals();

    public List<String> getTerminals();

    public String getEpsilon();
}
```

2. Schreiben Sie eine Klasse LL1Parser welche im Konstruktor eine Instanz von ILL1ParsingTable übergeben bekommt. Diese Klasse hat eine Methode *parse(List<String> token) throws MyParseException* welche eine Liste von Token (als String) übergeben bekommt. Die Methode gibt true zurück wenn die Eingabe korrekt war oder wirft eine ParseException.

3. public interface IToken {  
 public String toString();  
 public Object getValue();  
}

```
public interface ILexer {  
    public boolean hasMore();  
    public void advance() throws LexerException;  
    public boolean eat(String token);  
    public IToken getNext();  
}
```

Die Eingabe als Liste von Strings darzustellen ist problematisch. Diese Darstellung wurde gewählt um die Aufgabe zu vereinfachen. Verbessern Sie ihren LL1 Parser indem Sie Eingaben aus einer Datei an einen Lexer weiter geben. Die Signatur der `parse` Methode ändert sich damit zu *`parse(ILexer l)`*. Implementieren Sie dazu obige Interfaces. Hinweis: Jede Grammatik benötigt ihre eigene Lexerimplementierung.