

---

**Compilerbau**


---

Blatt 4

Abgabe: 14.11.2001

1. [2 Punkte] Welche Sprachen beschreiben LL(0)-Grammatiken? Warum?
2. [5 Punkte] Beweisen Sie Lemma 2.4, also folgende Fixpunktgleichung:

$$\text{follow}_k(A) = \{\xi \in \text{first}_k(\beta \text{follow}_k(B)) \mid B \rightarrow \alpha A \beta\}$$

Hinweis: Verwenden Sie im Beweis zum Abbruch der Iteration die Definition von  $\text{follow}_k$ .

3. [15 Punkte] Implementieren Sie in der Struktur `Grammar` Funktionen für `first` und `follow` mit der folgenden Signatur:

```
first : ('n, 't, 'attrib) grammar -> int
      -> ('n, 't) symbol list -> 't list list
follow : ('n, 't, 'attrib) grammar -> int -> 'n -> 't list list
```

Legen Sie die Implementation so an, daß `first g k` eine Funktion liefert, die effizient arbeitet.

Hinweis: Benutzen Sie Fixpunktiteration und Lemma 2.4.

4. [4 Punkte] Überführen Sie folgende Grammatik in eine äquivalente LL(1)-Grammatik:

```
<type-exp>  -> ' <ident>
           | ( <type-exp> )
           | <type-exp> -> <type-exp>
           | <type-exp> <product-type>+
           | <type-const>
           | <type-exp> <type-const>
           | ( <type-exp> <type-param>+ ) <type-const>
           | <type-exp> as ' <ident>
<product-type> -> * <type-exp>
<type-param>  -> , <type-exp>
```

Beweisen Sie die LL(1)-Eigenschaft Ihres Ergebnisses.

5. [4 Punkte] Implementieren Sie einen Akzeptor für die Grammatik aus Aufgabe 4 mit Hilfe des Codes in L1. Sollten Sie Aufgabe 3 nicht gelöst haben, wenden Sie sich an Martin Gasbichler.