

Lösungshinweise/-vorschläge zum Übungsblatt 2: Software-Entwicklung 1 (WS 2017/18)

Die Hinweise und Vorschläge in diesem Dokument sollen der Lösungsfindung dienen und erheben demnach weder Anspruch auf Vollständigkeit noch Korrektheit. Sollten Sie Fehler finden würden wir uns freuen wenn Sie uns diese mitteilen. (Kontaktinformationen finden Sie auf unserer Webpräsenz.)

Aufgabe 1 Kontextfreie Grammatiken (9 Punkte)

- a) Welches ist der kürzeste Satz der Sprache L? Geben Sie alle Sätze der Sprache (mit Ableitung) an, die bis zu fünf Buchstaben besitzen.

Kürzeste: cb, bis zu fünf: {dcb, ddc b, dddcb, cbacb}

Ableitungen:

$U \Rightarrow VU \Rightarrow dU \Rightarrow dSb \Rightarrow dcb$

$U \Rightarrow VU \Rightarrow dU \Rightarrow dVU \Rightarrow ddU \Rightarrow ddSb \Rightarrow ddc b$

$U \Rightarrow VU \Rightarrow dU \Rightarrow dVU \Rightarrow ddU \Rightarrow ddVU \Rightarrow dddU \Rightarrow dddSb \Rightarrow dddcb$

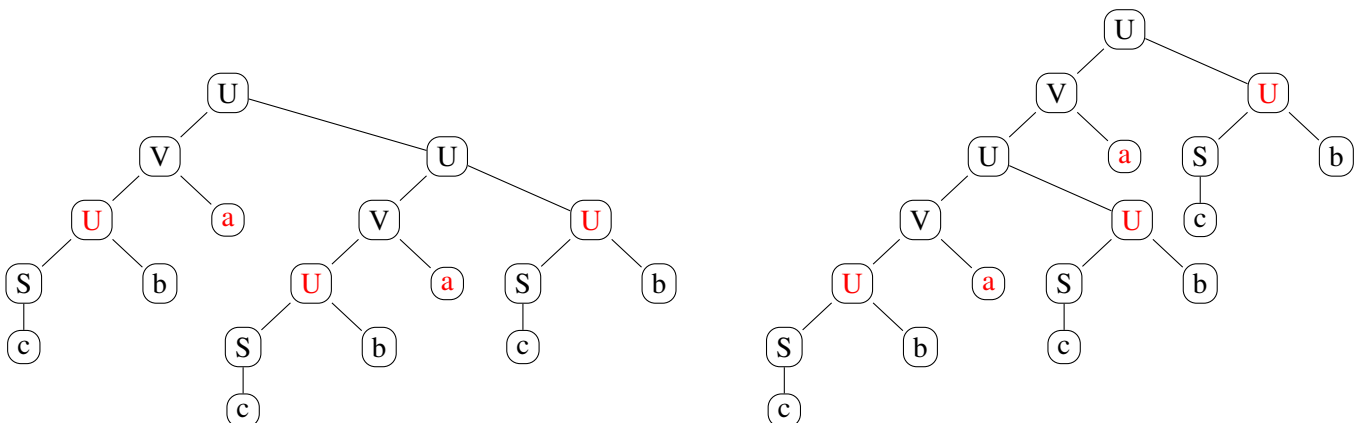
$U \Rightarrow VU \Rightarrow UaU \Rightarrow SbaU \Rightarrow cbaU \Rightarrow cbaSb \Rightarrow cbacb$

- b) Überprüfen Sie, ob Ihre Ableitungen aus Aufgabenteil a *Linksableitungen* sind. Falls nicht, geben Sie zu den entsprechenden Sätzen auch eine Linksableitung an. Können Sie für einen der Sätze auch eine andere Linksableitung angeben?

Die Ableitungen in Teil a) sind Linksableitungen und es gibt keine anderen Linksableitungen.

- c) Zeichnen Sie einen Syntax-Baum¹ für den Satz “cbacbcb”.

Hier ist es hilfreich den Baum “bottom-up”, also ausgehend vom Wort, aufzubauen. Da jedes Terminal-Symbol nur von einem Nichtterminal-Symbol erzeugt werden kann, ist die Vorgehensweise relativ eindeutig. Es gibt aber zwei Möglichkeiten, wie man zu der Satzform “UaUaU” kommen kann, wie folgende Bäume auch zeigen:



- d) Ist die Grammatik Γ eindeutig? Wenn nicht, geben Sie ein Gegenbeispiel an.

¹Syntaxbäume werden erst in der Vorlesung am Donnerstag behandelt.

Wie man an den zwei Ableitungsbäumen aus Teil c) sieht ist die Grammatik nicht eindeutig. Es reicht aber auch, zwei verschiedene Ableitungsbäume für UaUaU zu erstellen, das ist schließlich auch der springende Punkt.

Aufgabe 2 Schreiben von Grammatiken (7 Punkte)

- a) Geben Sie eine kontextfreie Grammatik für E-Mail Adressen an. Dabei sollen folgende Punkte beachtet werden: ...

$\Gamma = (N, T, \Pi, S)$ mit

$$\begin{aligned} N &= \{S, D, W, L\} \\ T &= \{ @, a, b, c, \dots, z, 0, 1, \dots, 9, _ , - , . \} \\ \Pi &= \left\{ \begin{array}{l} S \rightarrow D@D.W \\ D \rightarrow W.D \mid W \\ W \rightarrow LW \mid L \\ L \rightarrow a \mid b \mid c \mid \dots \mid z \mid 0 \mid 1 \mid \dots \mid 9 \mid _ \mid - \end{array} \right\} \end{aligned}$$

- b) (*Zusatzaufgabe*) Sind alle oben genannten Punkte korrekt, das heißt gelten sie für alle gültigen E-Mail Adressen?

Die Syntax für Email-Adressen ist in RFC5322 (<http://tools.ietf.org/pdf/rfc5322.pdf>) definiert und kann zum Beispiel unter <http://isemail.info> geprüft werden. Eine gültige Adresse, die fast alle Punkte oben verletzt ist `"..@$"@de`.

Aufgabe 3 Parser (8 Punkte)

```
public class FemtoParser extends Parser {

    void Programm() {
        VereinbarungsListe();
        accept(print);
        Ausdruck();
        accept(semikolon);
    }

    void VereinbarungsListe() {
        if (nextIs(typeInt)) {
            WertVereinbarung();
            VereinbarungsListe();
        } else {
            // empty
        }
    }

    void WertVereinbarung() {
        accept(typeInt);
        accept(bezeichner);
        accept(gleich);
        Ausdruck();
        accept(semikolon);
    }

    void Ausdruck() {
        if (nextIs(bezeichner)) {
            accept(bezeichner);
        } else if (nextIs(zahl)) {
            accept(zahl);
        } else {
            accept(klammerAuf);
            Ausdruck();
            Operator();
            Ausdruck();
            accept(klammerZu);
        }
    }

    void Operator() {
        if (nextIs(plus)) {
            accept(plus);
        } else {
            accept(mult);
        }
    }

    public static void main(String[] args) {
        FemtoParser p = new FemtoParser();
        p.Programm();
        p.ende();
    }
}
```