

HUMBOLDT-UNIVERSITÄT ZU BERLIN



Wissenschaftliches Arbeiten in der Linguistik (Technische Übung)

LaTeX – Teil 5: Pakete für Linguisten II

Antonio Machicao y Priemer

www.linguistik.hu-berlin.de/staff/amyp

Institut für deutsche Sprache und Linguistik

Inhaltsverzeichnis

1 Strukturbäume

- forest laden
- forest-Syntax
- Bäume in Bsp.-Umgebungen
- Abkürzungen in Bäumen
- Glossen oder Übersetzungen
- Hoch- und tiefgestellt
- Pfeile
- Auszeichnung von Knoten
- Weitere Features

2 Venndiagramme

- tikz-Beispiele
- venndiagram laden
- Venndiagramme zeichnen
- Weitere Features

3 Vokalviereck (einfach)

- vowel laden
- vowel-Umgebungen
- Vokale hinzufügen

4 Hausaufgabe

Grundlage & empfohlene Lektüre

...basierend auf Freitag und Machicao y Priemer (2015) und auf Machicao y Priemer und Kerkhof (2016)

→ LINK

1 Strukturbäume

2 Venndiagramme

3 Vokalviereck (einfach)

4 Hausaufgabe

Strukturbäume

Es gibt verschiedene Pakete für um Strukturbäume zu zeichnen:

- `qtree`
- `pstrees` (anspruchsvollere Syntax, aber mächtiger als `qtree`)
- `tikz-qtree`
- **forest** (einfache Syntax, mächtiger als `pstrees` und `qtree`)
- ...

Wir werden hier mit `forest` arbeiten, für eine kurze Beschreibung der anderen Pakete siehe Freitag und Machicao y Priemer (2015) oder Linke und Linke (2005).

forest laden

Laden Sie das Paket:

```
\usepackage{forest}
```

- forest bietet eine Reihe von Features für linguistische Strukturbäume. Man kann diese Features als Paket-Option laden (oder mit einem Extrabefehl, der die Linguistik-Bibliothek lädt).

```
\usepackage[linguistics]{forest}
```

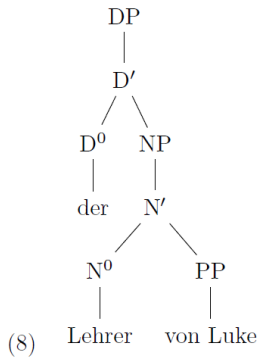


Abb. 1: ohne Option

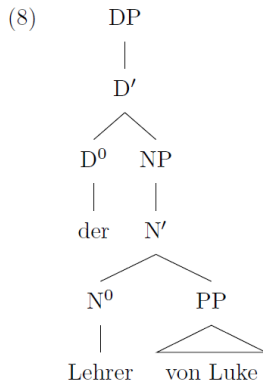


Abb. 2: mit Option

- Wie bereits erwähnt, definiert gb4e bestimmte \LaTeX -Befehle um, die auch für forest benötigt werden. Es ist sehr wichtig, dass Sie **zuerst forest** und **erst dann gb4e** laden (gb4e sollte eines der letzten Pakete sein!)

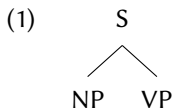
```
\usepackage[linguistics]{forest}
```

```
\usepackage{gb4e}
```


forest-Syntax

- Um Strukturbäume zu zeichnen verwenden Sie die `forest`-Umgebung.
- Die weitere `forest`-Syntax ist wie die bereits bekannte Klammernotation bei Strukturbäumen.

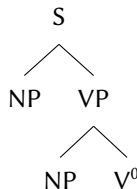
```
\begin{forest}  
  [S [NP] [VP]]  
\end{forest}
```



- Die Klammernotation können Sie hier üben:
<http://ironcreek.net/phpsyntaxtree/>

Bei größeren Bäumen ist es empfehlenswert die Klammernotation **nicht linear** zu verwenden. Lassen Sie aber **keine Leerzeilen** in Ihrem Baum, andernfalls bekommen Sie Fehlermeldungen.

```
\begin{forest}
  [S
    [NP]
    [VP
      [NP]
      [V$^{0}$]
    ]
  ]
\end{forest}
```



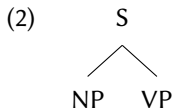
vs.

```
\begin{forest}
[S [NP] [VP [NP] [V$^{0}$]]]
\end{forest}
```

Bäume in Bsp.-Umgebungen

Bei Verwendung der Option `linguistics` können Sie Ihre Bäume **in Beispielumgebungen einbetten**.

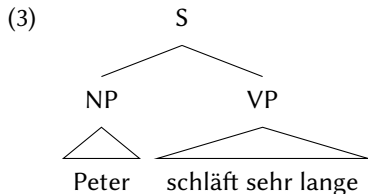
```
\begin{exe}  
\ex  
  \begin{forest}  
    [S [NP] [VP]]  
  \end{forest}  
\end{exe}
```



Abkürzungen in Bäumen

Mit dem Befehl `roof` können Sie **Knoten abkürzen**.

```
\begin{exe}
\ex
  \begin{forest}
    [S [NP [Peter, roof]] [VP [schläft sehr lange, roof]]]
  \end{forest}
\end{exe}
```

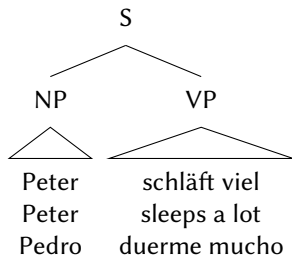


Glossen oder Übersetzungen

Mit `\` können Sie im Baum **Glossen/Übersetzungen** angeben.

```
\begin{forest}
[S
  [NP
    [Peter \
     Peter \
     Pedro, roof]
  ]
  [VP
    [schläft viel \
     sleeps a lot \
     duerme mucho, roof]
  ]
]
\end{forest}
```

(4)



Hoch- und tiefgestellt

- Die Zeichen \wedge und $_$ werden für hoch- und tiefgestellte Elemente verwendet.
- Diese Zeichen können aber **nur im Mathematikmodus** benutzt werden $\$^1\$$
 $\$_1\$$

- Der **Default-Skopus** von \wedge und $_$ ist nur ein Zeichen, siehe (5).
- Um den Skopus zu erweitern, müssen die **geschwungenen Klammern** $\{ \}$ benutzt werden, siehe (6).

```
\ea \label{ex:BspHochTief}
\ea X$^Agens$ Y$^Agens oder Patiens$
\ex X$_Agens$ Y$_Agens oder Patiens$
\z
\ex \label{ex:BspKlammer}
\ea X$^{Agens}$ Y$^{Agens oder Patiens}$
\ex X$_{Agens}$
Y$_{Agens oder Patiens}$
\z
\z
```

- (5) a. $X^A gens Y^A gensoderPatiens$
 b. $X_A gens Y_A gensoderPatiens$
- (6) a. $X^{Agens} Y^{AgensoderPatiens}$
 b. $X_{Agens} Y_{AgensoderPatiens}$

- Wie (5) und (6) zeigen, verhält sich **Text innerhalb vom Mathematikmodus** anders (kursiv ohne Spatien).
- Um Text wiederzugeben verwenden Sie den Befehl `\textrm{ }`, siehe Beispiel (7).

$$X^{\text{\textrm{Agens}}}$$

$$Y^{\text{\textrm{Agens oder Patiens}}}$$

$$X_{\text{\textrm{Agens}}}$$

$$Y_{\text{\textrm{Agens oder Patiens}}}$$

(5) a. $X^{\textit{Agens}} Y^{\textit{AgensoderPatiens}}$

b. $X_{\textit{Agens}} Y_{\textit{AgensoderPatiens}}$

(6) a. $X^{\textit{Agens}} Y^{\textit{AgensoderPatiens}}$

b. $X_{\textit{Agens}} Y_{\textit{AgensoderPatiens}}$

(7) a. $X^{\textit{Agens}} Y^{\textit{Agens oder Patiens}}$

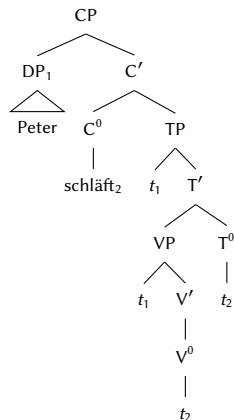
b. $X_{\textit{Agens}} Y_{\textit{Agens oder Patiens}}$

Baum mit hochgestellten und tiefgestellten Zeichen

```

[CP
  [DP$_{1}$ [Peter, roof]]
  [C$^{\backslash\mathbf{prime}}$
    [C$^{0}$ [schläft$_{2}$]]
    [TP
      [$t_{1}$]
      [T$^{\backslash\mathbf{prime}}$
        [VP
          [$t_{1}$]
          [V$^{\backslash\mathbf{prime}}$
            [V$^{0}$ [$t_{2}$]]
          ]
        ]
      ]
    ]
    [T$^{0}$ [$t_{2}$]]
  ]
]
]
]

```



Pfeile

- Um Bewegungen anzugeben, können Pfeile von Knoten zu Knoten gezeichnet werden.
- Dafür werden den Knoten **Namen** (Befehl: `name=`) gegeben, und Pfeile von Knotennamen zu Knotennamen gezeichnet

Befehl: `\draw[X] (Y) to[out=V, in=W] (Z);`

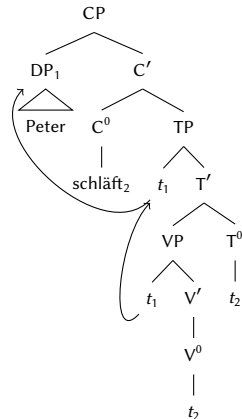
- **X**: Art des Pfeils (`->` `<-` `<->` `-`)
- **Y**: Name des Startknotens
- **Z**: Name des Landeknotens
- **V**: Ausgangsausrichtung im Startknoten
(`south/north` + `east/west`)
- **W**: Ankunftsrichtung im Landeknoten
- `;`: Ende des Befehls

```
\draw[->] (T10) to[out=south west, in=south west](T11);
```

```

[CP
  [DP$_{1}$, name=T12 [Peter, roof]]
  [C$^{\backslash\mathbf{prime}}$
    [C$^{\{0\}}$ [schläft$_{2}$, name=T22]]
    [TP
      [$t_{1}$, name=T11]
      [T$^{\backslash\mathbf{prime}}$
        [VP
          [$t_{1}$, name=T10]
          [V$^{\backslash\mathbf{prime}}$
            [V$^{\{0\}}$ [$t_{2}$, name=T20]]
          ]
        ]
      ]
      [T$^{\{0\}}$ [$t_{2}$, name=T21]]
    ]
  ]
]
]
]
\draw[->] (T10)
  to[out=south west, in=south west](T11);
\draw[->] (T11)
  to[out=south west, in=south west](T12);

```



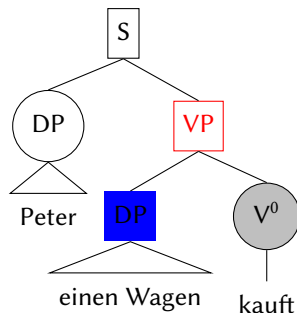
- Auszeichnung von Knoten:

- `draw`: Viereck
- `circle, draw`: Kreis
- `red`: Knoten rot markieren
- `fill=X`: Knoten mit Farbe *X* hinterlegen
- `circle, draw, fill=lightgray`: hellgrau hinterlegter Kreis

```

[S, draw
  [DP, circle, draw
    [Peter, roof]
  ]
  [VP, draw, red
    [DP, fill=blue
      [einen Wagen, roof]
    ]
    [V$^{0}$, circle, draw,
      fill=lightgray
      [kauft]
    ]
  ]
]

```



Weitere Features

- `forest` ist ein sehr mächtiges Paket. Um alle Vorzüge von `forest` zu erfahren, schauen Sie sich die Dokumentation an (Živanović, 2017).
- Eine Anleitung für den schnellen Start finden Sie unter Vanden Wyngaerd (2016).

Venndiagramme

- Venndiagramme können mit Code und dem Paket `tikz` gezeichnet werden. Es ist zwar sehr aufwändig, aber das Resultat ist ziemlich perfekt.
- Eine andere Lösung ist das Paket `venndiagram`. Dieses Paket ist sehr leicht zu bedienen, aber etwas beschränkt in den Möglichkeiten.

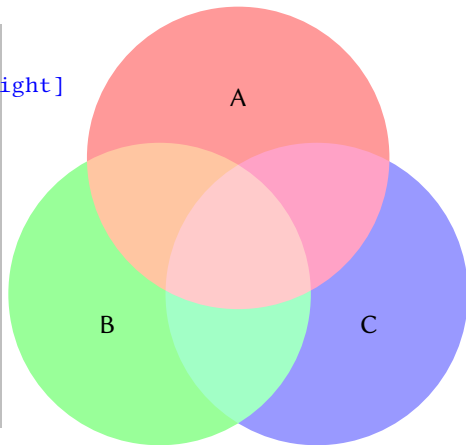
tikz-Beispiele

```
\begin{tikzpicture}

\begin{scope}[blend group=soft light]
  \fill[red!40!white]
    (90:1.2) circle (2);
  \fill[green!40!white]
    (210:1.2) circle (2);
  \fill[blue!40!white]
    (330:1.2) circle (2);
\end{scope}

\node at (90:2) {A};
\node at (210:2) {B};
\node at (330:2) {C};

\end{tikzpicture}
```




```

\begin{tikzpicture}

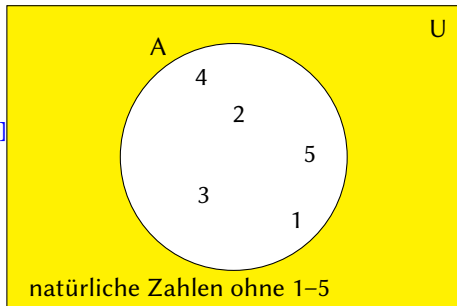
\def\firstrectangle{(0,0) rectangle (6,4)}
\def\firstcircle{(3,2) circle (1.5cm)}
\def\secondcircle{(0:2cm) circle (1.5cm)}

\begin{scope}[shift={(-3cm,2cm)}]
  \clip \firstrectangle;
  \fill[yellow] \firstrectangle;
  \fill[white] \firstcircle;
\end{scope}

\begin{scope}[shift={(-3cm,2cm)}]
  \draw \firstcircle;
  \draw \firstrectangle;

  \node at (33:6.8) {U};
  \node at (60:4) {A};
  \node at (40:4) {2};
  \node at (30:3) {3};
  \node at (17:4) {1};
  \node at (50:4) {4};
  \node at (27:4.5) {5};
  \node at (6.9:2.3) {natürliche Zahlen ohne 1--5};
\end{scope}
\end{tikzpicture}

```



```

\begin{tikzpicture}

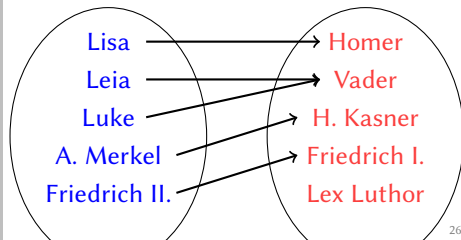
\def\firstellipse{(0,0) ellipse (1.3cm and 1.7cm)}
\def\secondellipse{(3.4,0) ellipse (1.3cm and 1.7cm)}

\begin{scope}
  \draw \firstellipse ;
  \draw \secondellipse ;

  \node at (90:-2.25)
  {\textsc{dom}(f)};
  \node at (90:1.25)
  {\blue{Lisa}};
  \node at (90:.75)
  {\blue{Leia}};
  \node at (90:.25)
  {\blue{Luke}};
  \node at (90:-.25)
  {\blue{A. Merkel}};
  \node at (90:-.75)
  {\blue{Friedrich II.}};

  \node at (3.4,-2.25)
  {\textsc{rng}(f)};
  \node at (3.4,1.25)
  {\color{red}Homer};
  \node at (3.4,.75)
  {\color{red}Vader};
  \node at (3.4,.25)
  {\color{red}H. Kasner};
  \node at (3.4,-.25)
  {\color{red}Friedrich I.};
  \node at (3.4,-.75)
  {\color{red}Lex Luthor};
\end{scope}
\end{tikzpicture}

```



venndiagram laden

- Das Paket `venndiagram` basiert auf `tikz`-Code (das `tikz`-Paket muss jedoch nicht extra geladen werden!)

Laden Sie das Paket:

```
\usepackage{venndiagram}
```

Venndiagramme zeichnen

- Das Paket `venndiagram` definiert zwei Umgebungen:
 - 1 Venndiagramme mit zwei Mengen
 - 2 Venndiagramme mit drei Mengen

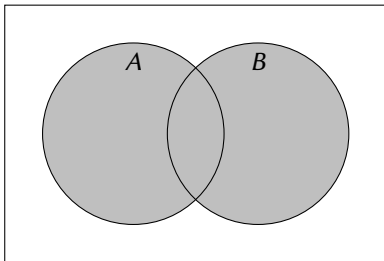
```
\begin{venndiagram2sets}
```

```
\end{venndiagram2sets}
```

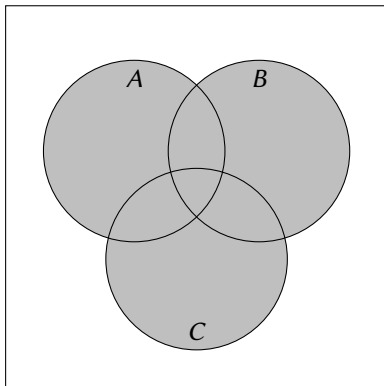
```
\begin{venndiagram3sets}
```

```
\end{venndiagram3sets}
```

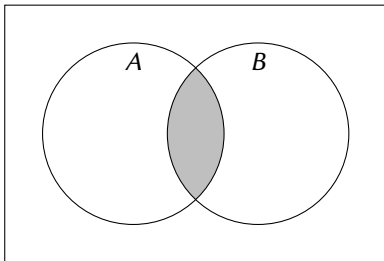
```
\begin{venndiagram2sets}  
\fillA \fillB  
\end{venndiagram2sets}
```



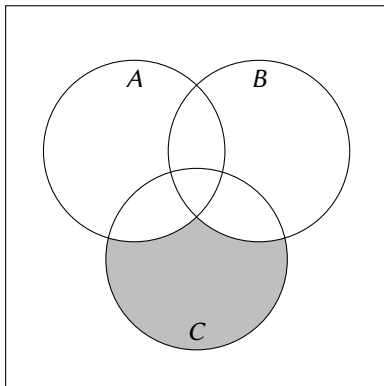
```
\begin{venndiagram3sets}  
\fillA \fillB \fillC  
\end{venndiagram3sets}
```



```
\begin{venndiagram2sets}
\fillACapB
\end{venndiagram2sets}
```



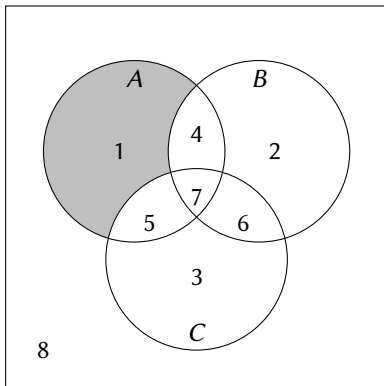
```
\begin{venndiagram3sets}
\fillOnlyC
\end{venndiagram3sets}
```



Um Elemente hinzuzufügen, können diese als Optionen zur Umgebung angegeben werden.

```
\begin{venndiagram3sets}[  
  labelOnlyA={1},  
  labelOnlyB={2},  
  labelOnlyC={3},  
  labelOnlyAB={4},  
  labelOnlyAC={5},  
  labelOnlyBC={6},  
  labelABC={7},  
  labelNotABC={8}  
]
```

```
\fillOnlyA  
\end{venndiagram3sets}
```



Weitere Features

- Für weitere Befehle, schauen Sie sich die Dokumentation an (Talbot, 2016).
- Für komplexe Diagramme ist die Verwendung von `tikz` empfehlenswert.

- 1 Strukturbäume
- 2 Venndiagramme

3 Vokalviereck (einfach)

4 Hausaufgabe

Vokalviereck (einfach)

- Vokalvierecke können mit Code und dem Paket `tikz` gezeichnet werden. Es ist zwar sehr aufwändig, aber das Resultat ist ziemlich perfekt.
- Siehe: <http://userblogs.fu-berlin.de/langsci-press/2016/06/15/drawing-vowel-charts-with-tikz/>
- Eine andere Lösung ist das Paket `vowel`. Dieses Paket ist sehr leicht zu bedienen, aber etwas beschränkt in den Möglichkeiten.

vowel laden

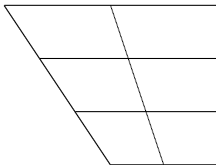
- Zusätzlich zum `vowel`-Paket wird das `tipa`-Paket für die Vokalzeichen benötigt.

Laden Sie das Paket:

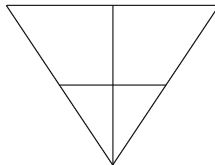
```
\usepackage{vowel}
```

vowel-Umgebungen

```
\begin{vowel}  
\end{vowel}
```



```
\begin{vowel}[triangle,three]  
\end{vowel}
```



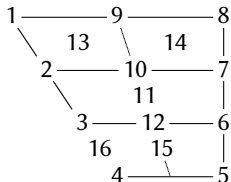
Für weitere Optionen, s. Rei (2001).

Vokale hinzufügen

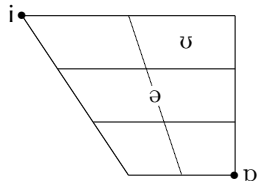
```
\putcvowel[1|r]{x}{y}
```

Bei dem Befehl `putcvowel`:

- Optionen: `l` und `r` → links oder rechts vom Knotenpunkt
- Argumente:
 - `x` → IPA-Zeichen
 - `y` → festgelegte Position im Viereck



```
\begin{vowel}
  \putcvowel[l]{i}{1}
  \putcvowel[l]{\textscripta}{5}
  \putcvowel{\textschwa}{11}
  \putcvowel{\textupsilon}{14}
\end{vowel}
```

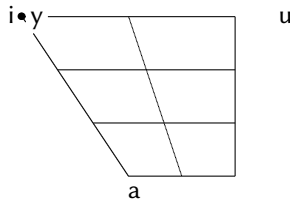


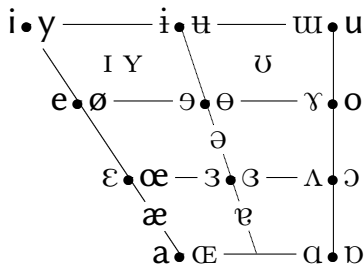
```
\putvowel[l|r]{x}{z}{w}
```

Bei dem Befehl `putvowel`:

- Optionen: `l` und `r` → links oder rechts vom Knotenpunkt
- Argumente:
 - `x` → IPA-Zeichen
 - `z` → Koordinate (x-Achse)
 - `w` → Koordinate (y-Achse)

```
\begin{vowel}
  \putvowel[l]{i}{0pt}{0pt}
  \putvowel[r]{y}{0pt}{0pt}
  \putvowel{a}{42pt}{66pt}
  \putvowel{u}{99pt}{0pt}
\end{vowel}
```





Schauen Sie sich das Handbuch (Rei, 2001) für weitere Features des Pakets an.

- 1 Strukturbäume
- 2 Venndiagramme

- 3 Vokalviereck (einfach)
- 4 Hausaufgabe

Hausaufgabe 1

- Laden Sie die folgende Datei aus dem Moodlekurs herunter und speichern Sie sie in Ihrem Ordner zusammen mit Ihrer `.tex`-Datei:
 - ❶ `test4PDF.pdf`
 - ❷ `lsp-gb4eMyP.sty`
 - ❸ `lsp-cgloss.sty`
- `lsp-gb4eMyP.sty` ist eine leicht veränderte Version von `gb4e`, die weniger instabil ist. `lsp-gb4eMyP.sty` greift auf `lsp-cgloss.sty` (Paket für Glossen), daher benötigen Sie beide Dateien.
- `lsp-gb4eMyP.sty` können Sie mit der gleichen Syntax wie `gb4e.sty` verwenden.

Hausaufgabe 2

- Installieren Sie die folgenden Pakete in Ihrem „myName.tex“-Dokument (mit dem Befehl `usepackage` und den oben besprochenen Optionen).
 - `vowel`
 - `tipa`
 - `forest`
 - `venndiagram`
 - `lsp-gb4eMyP`
- Ergänzen Sie die Option `hidelinks` für das Paket `hyperref`. Hier die Syntax dafür:

```
\usepackage[bookmarksnumbered, hidelinks]{hyperref}
```

NB Bitte beachten Sie, dass `hyperref` als letztes Paket geladen werden sollte.

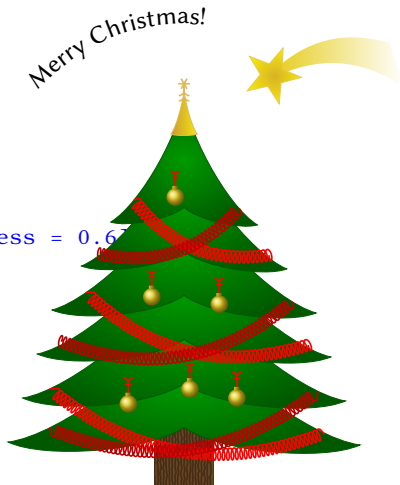
Hausaufgabe 3

- Verwenden Sie Ihre „myName.tex“-Datei vom letzten Mal und
- geben Sie den benötigten Code ein, um das Ergebnis zu erhalten, das Sie in „test4PDF.pdf“ sehen.
- Laden Sie dann Ihre „myName.tex“-Datei und Ihr PDF-Ergebnis bei Moodle hoch.
(Sie müssen nur 2 Dateien hochladen!)

```

\begin{tikzpicture}[
bauble/.pic = {
\shade [ball color = yellow!60!brown]
(0,-0.9) circle [radius = 0.3];
\draw [
ultra thick,
red,
- >[scale=0.6]<[scale=0.9]],
] (0,-0.6) -- (0,0);
\shade [
left color = yellow!40!brown,
right color = yellow!30!black,
]
(-0.1,-0.62) to[bend right, looseness = 0.67]
(0.1,-0.62) -- ++(0,0.1) -| cycle;
},
]
\node (Stern) [
star,
star point height = 6mm,
minimum size = 20mm,
thick,
draw = yellow!60!brown,
inner color = yellow!40!brown,
outer color = yellow!80!brown,
] at (0,0) {}

```



Literatur I

- Freitag, Constantin und Antonio Machicao y Priemer (2015). \LaTeX -Einführung für Linguisten. Manuskript. URL <https://www.linguistik.hu-berlin.de/de/staff/amp/latex-einfuehrung>, Zugriff: 12.04.2015.
- Knuth, Donald E. (1986). *The \TeX book*. Boston, MA: Addison-Wesley.
- Kopka, Helmut (1994). *\LaTeX : Einführung. Band 1*. Bonn: Addison-Wesley.
- Linke, Rona und Alex Linke (2005). LATEX für Linguisten: IPA, Glossing und Baumstrukturen. Manuskript. URL <http://homepage.ruhr-uni-bochum.de/alexander.linke-2/linguistik/LaTeX/>, Zugriff: 07.05.2013.
- Machicao y Priemer, Antonio (2018). Hinweise für Seminararbeiten. Manuskript. URL https://www.linguistik.hu-berlin.de/de/staff/amp/downloads/myp2018-04-06-hinweise_seminararbeit.pdf, Zugriff: 16.10.2018.
- Machicao y Priemer, Antonio und Robyn Kerkhof (2016). \LaTeX -Einführung für Linguisten – Slides. Präsentation beim 7. linguistischen Methodenworkshop an der Humboldt-Universität zu Berlin – 22.–24. Februar 2016. URL <https://www.linguistik.hu-berlin.de/de/staff/amp/latex-einfuehrung>, Zugriff: 23.02.2016.

Literatur II

- Rei, Fukui (2001). vowel – Draw vowel charts for phonetic research. *CTAN: Comprehensive TeX Archive Network*. URL <https://ctan.org/pkg/vowel>, Zugriff: 01.12.2017.
- Talbot, Nicola L. C. (2016). venndiagram v1.1: Drawing Simple Venn Diagrams. *CTAN: Comprehensive TeX Archive Network*. URL <https://ctan.org/pkg/venndiagram>, Zugriff: 01.12.2017.
- Vanden Wyngaerd, Guido (2016). Forest Quickstart Guide for Linguists. Manuskript. URL <https://ling.auf.net/lingbuzz/003391>, Zugriff: 30.11.2017.
- Živanović, Sašo (2017). Forest: a PGF/TikZ-based package for drawing linguistic trees v2.1.5. *CTAN: Comprehensive TeX Archive Network*. URL <https://ctan.org/pkg/forest>, Zugriff: 30.11.2017.