

# L<sup>A</sup>T<sub>E</sub>X para linguistas

Rafael Beraldo  
ForMA — IEL/Unicamp

17, 24 e 31 de julho de 2020

## Resumo

Curso virtual de L<sup>A</sup>T<sub>E</sub>X para linguistas oferecido pelo ForMA — Núcleo de Estudos em Gramática Formal, Mudança e Aquisição do Instituto de Estudos da Linguagem (IEL), Unicamp — nos dias 17, 24 e 31 de julho de 2020. O curso é uma adaptação para o ambiente virtual da versão presencial, que pode ser [encontrada no Github](#). A intenção é que se leia o código-fonte, pois ele contém comentários explicativos importantes.

## Sumário

<b>1</b>	<b>Classes e compilação</b>	<b>2</b>
<b>2</b>	<b>Comandos e espaços em branco</b>	<b>2</b>
2.1	Exercício: espaços em branco . . . . .	3
<b>3</b>	<b>Símbolos especiais</b>	<b>3</b>
3.1	Exercício: caracteres reservados . . . . .	4
<b>4</b>	<b>O preâmbulo do documento</b>	<b>4</b>
<b>5</b>	<b>O corpo do documento</b>	<b>5</b>
5.1	Exercício: completar o artigo . . . . .	6
<b>6</b>	<b>Pacotes</b>	<b>6</b>
6.1	Exercício: carregar pacotes . . . . .	6
6.2	Documentação no CTAN . . . . .	7
<b>7</b>	<b>Fontes</b>	<b>7</b>
7.1	Itálicos, negritos e outros tipos . . . . .	7
7.2	Tamanhos de fontes . . . . .	7
7.3	Exercício: fontes . . . . .	8
<b>8</b>	<b>Layouts de página</b>	<b>8</b>
<b>9</b>	<b>Posicionamento do texto</b>	<b>9</b>
9.1	Ambientes . . . . .	9
9.2	Controlando o espaço na linha . . . . .	9
9.3	Espaço vertical . . . . .	10

<b>10 Listas</b>	<b>10</b>
10.1 Customizando o ambiente <code>enumerate</code>	10
10.2 Exercício: receita usando listas	10
<b>11 Tabulares e tabelas</b>	<b>11</b>
11.1 Exemplos de tabelas	12
11.1.1 Ingredientes básicos para o pão	12
11.1.2 Distâncias cósmicas	13
11.2 Flutuando com <code>table</code>	14
11.3 Ferramentas para tabelas	14
<b>12 Imagens</b>	<b>15</b>
12.1 Exemplo de imagem: nossos vizinhos	15
<b>13 A classe <code>abntex2</code></b>	<b>17</b>
13.1 Exemplo: adaptando este documento	17
<b>14 Bibliografias</b>	<b>17</b>
14.1 Exemplo: bibliografia e citação	18
<b>15 Macros</b>	<b>18</b>
15.1 Macros de substituição	18
15.2 Macros com argumentos	18
15.3 Novos ambientes	19
15.4 Exercício: automatizando com macros	20
<b>16 Linguística</b>	<b>20</b>
16.1 Transcrições fonéticas	20
16.2 Digitar transcrições diretamente	21
16.3 Árvores sintáticas com o <code>forest</code>	21
16.4 Exemplos numerados e glosas com o <code>linguex</code>	24
16.4.1 Referências cruzadas	24

## 1 Classes e compilação

Arquivos `.tex` começam com um cabeçalho (também chamado de *preâmbulo*) que contém informações básicas sobre o documento, como tipo (artigo, livro, carta etc.), metadados e carregamento de pacotes que expandem as funções do `LATEX`.

A classe deste arquivo é `article` e, por isso, conseguimos usar o ambiente `abstract`. Vamos remover esse ambiente e mudar a classe para `book` e ver o que acontece com o número das seções. Vamos observar também como compreender erros na compilação, mantendo o ambiente de resumo.

## 2 Comandos e espaços em branco

Espaços em branco não são tomados literalmente no `LATEX`: esses vários espaços serão condensados em apenas um espaço em branco.

Assim, o uso de espaços no código-fonte separa palavras e comandos e deve seguir melhores práticas que são aprendidas com o tempo, leitura do código de outras pessoas e desenvolvimento de um estilo pessoal.

Certos comandos, como `\tableofcontents` são simples, ou seja, não têm argumentos. Já comandos como `\section{Introdução}` recebem argumentos, assim como verbos têm valência. Para o  $\text{\LaTeX}$ , não há muita diferença entre...

```
\section{Introdução} e  
\section {Introdução}
```

...mas por que a primeira versão é mais adequada do ponto de vista humano?

## 2.1 Exercício: espaços em branco

Acima de tudo, é fundamental ressaltar que o julgamento imparcial das eventualidades maximiza as possibilidades por conta das direções preferenciais no sentido do progresso. Desta maneira, a complexidade dos estudos efetuados cumpre um papel essencial na formulação das condições financeiras e administrativas exigidas, que são três: 1. O mundo atual. 2. Os amigos de família. 3. A importância dos mercados mundiais. Nunca é demais lembrar o peso e o significado destes problemas, uma vez que o fenômeno da Internet exige a precisão e a definição do processo de comunicação como um todo. No entanto, não podemos esquecer que a consulta aos diversos militantes agrega valor ao estabelecimento das regras de conduta normativas. Do mesmo modo, o acompanhamento das preferências de consumo garante a contribuição de um grupo importante na determinação das novas proposições. O empenho em analisar o desenvolvimento contínuo de distintas formas de atuação aponta para a melhoria do sistema de participação geral.

## 3 Símbolos especiais

**Aspas** Não usamos as chamadas “aspas burras” (“”):

“Abrimos aspas com dois acentos graves e fechamos com duas aspas simples.”

**Hífen, travessão e meia-risca** Existe uma diferença entre o hífen, o travessão e a meia-risca:

Leve um guarda-chuva — ouvi na rádio que pode chover entre 10h–13h.

**Espaços não quebráveis** Às vezes, é necessário que um espaço não se quebre ao fim de uma linha, por exemplo:

Às 10 horas de ontem...  
Fui à casa do Sr. Silva...  
Veja mais na página 40.

**Caracteres reservados** Os símbolos a seguir estão reservados:

# \$ % ^ & \_ { } ~ \

Devemos escapá-los, ou seja, precedê-los por uma barra `\`, para que sejam impressos da maneira correta:

# \$ % ^ & \_ { } ~ \

### 3.1 Exercício: caracteres reservados

(Descomente as linhas a seguir:)

## 4 O preâmbulo do documento

Documentos  $\text{\LaTeX}$  são divididos em duas partes: o preâmbulo e o documento em si, que fica entre `\begin{document}` e `\end{document}`. No preâmbulo deste documento, a primeira linha de código é:

```
\documentclass[10pt,a4paper,oneside]{article}
```

Anteriormente discutimos duas classes: `article` e `book`. Existem mais classes padrão, por exemplo:

**article** Para escrever artigos

**letter** Para redigir cartas

**report** Para escrever relatórios

**memoir** Baseada na classe `book`, traz vários comandos úteis

**book** Para livros

**beamer** Para apresentações de slides

O que são essas palavras entre os dois colchetes? São algumas das opções que a classe `article` nos oferece. Aqui estão as opções de classe mais comuns:

**10pt, 11pt, 12pt**

**a4paper, letterpaper, ...**

**fleqn** Equações são alinhadas à esquerda ao invés de serem centralizadas.

**leqno** A numeração das equações fica à esquerda ao invés da direita.

**titlepage, notitlepage**

**twocolumn**

**twoside, oneside** Arruma as margens para a impressão nos dois lados do papel ou apenas um.

**landscape** O documento é impresso em formato paisagem.

**openright, openany** Não funciona com a classe `article`, pois ela não fornece o comando `\chapter`.

**draft** Indica problemas de hifenização e justificação imprimindo um pequeno quadrado na margem direita. Também suprime a colocação das imagens, colocando um quadro em branco em seu lugar. O tempo de compilação é menor.

As classes padrão (`article`, `report`, `book` e `letter`) foram escritas para servir de base para resultados mais profissionais e por isso precisam de ajustes. Muitas vezes, as revistas científicas e universidades fornecem classes para que o texto fique automaticamente dentro do esperado pela instituição.

## 5 O corpo do documento

Enquanto que o preâmbulo contém a declaração do tipo de documento, carrega os pacotes e configura algumas opções (como veremos a seguir), o corpo do documento contém o texto, dividido ou não em partes, capítulos, seções e parágrafos.

O corpo do documento é delimitado por:

```
\begin{document}
...
\end{document}
```

No ambiente `document`, estruturamos nosso documento usando os comandos `\part`, `\chapter`, `\section`, `\subsection`, `\subsubsection`, `\paragraph` e `\subparagraph`.

Alternativamente, os comandos acima possuem uma versão estrelada (`\section*`), que produz uma versão não numerada e que não aparece no sumário.

Às vezes, o título de uma seção ou capítulo pode ser longo demais para o sumário. Por isso, é possível usar a seguinte sintaxe para controlar o nome que aparecerá no sumário:

```
\section[Seção muito longa]{Seção muito longa:
provavelmente não ficará muito boa no sumário}
```

Parágrafos são criados deixando uma linha em branca entre eles. Entretanto, essa linha em branca apenas indica ao  $\text{\LaTeX}$  o começo de um parágrafo novo e *não significa que uma linha em branco será impressa no documento*.

Finalmente, parágrafos que ocorrem imediatamente após uma subdivisão do documento não são indentados, por motivos de tradição tipográfica. No entanto, esse comportamento pode ser modificado carregando o pacote `indentfirst`.

## 5.1 Exercício: completar o artigo

De posse desses conhecimentos sobre como documentos  $\text{\LaTeX}$  são estruturados e o que deve ir no preâmbulo e no corpo do documento, vamos resolver `exercicios/meu-artigo.tex`.

# 6 Pacotes

No exercício anterior, note que a hifenização está errada. Por padrão, o  $\text{\LaTeX}$  é configurado para hifenizar de acordo com a língua inglesa. Para resolver esse problema, devemos carregar nosso primeiro pacote.

Existem várias coisas que não são possíveis com o  $\text{\LaTeX}$  básico — ao menos não trivialmente — mas durante sua vida como usuário desse sistema você descobrirá dezenas de pacotes muito úteis, que tornam tarefas tediosas e difíceis muito mais agradáveis de resolver. Para carregar um pacote, usamos a seguinte sintaxe no preâmbulo do nosso arquivo:

```
\usepackage[opções]{pacote}
```

Para resolver o problema da localização do nosso arquivo, utilizaremos o pacote `polyglossia`:

```
\usepackage{polyglossia}
\setdefaultlanguage[variant=brazilian]{portuguese}
\setotherlanguage{english}
```

Algumas das capacidades do `polyglossia` são:

- Ajustar datas de acordo com a língua
- Ajustar convenções tipográficas para a língua escolhida
- Hifenização
- Strings do documento (como em `\today`)

## 6.1 Exercício: carregar pacotes

Carregue o pacote `polyglossia` e defina o português brasileiro como padrão no arquivo `exercicios/meu-artigo.tex`.

## 6.2 Documentação no CTAN

O [CTAN](#) é o repositório de pacotes e documentação do  $\text{\LaTeX}$ . Antes de resolver alguma tarefa manualmente, é uma boa ideia conferir se alguém já não resolveu o problema com um pacote. Além disso, é possível encontrar a documentação de todos os pacotes lá. Vejamos a [documentação do polyglossia](#), por exemplo.

## 7 Fontes

Antigamente, arquivos  $\text{\LaTeX}$  compilados usando o programa `pdflatex` não podiam usar qualquer fonte. Existem catálogos de fontes suportadas por esse programa, como por exemplo [The LaTeX Font Catalogue](#). Atualmente, no entanto, é possível usar o `lualatex` ou ainda o `xelatex`, que oferecem suporte aos formatos de fonte mais comuns.

Para isso, devemos carregar o pacote `fontspec`:

```
\usepackage{fontspec}
\setmainfont{Times New Roman}
```

### 7.1 Itálicos, negritos e outros tipos

Fontes geralmente vêm em famílias que contêm diversos tipos: romanas maiúsculas e minúsculas, itálicos, negritos e versaletes, além dos algorismos de título e texto. A fonte usada por padrão no  $\text{\LaTeX}$ , chamada de Computer Modern e projetada pelo próprio Knuth, é bastante completa nesse respeito.

No passado, fontes não costumavam ter *tantos* estilos. Tipógrafos compunham livros inteiros com apenas um tipo e um tamanho. Hoje, temos uma miríade de possibilidades. Empregá-las com sabedoria e, talvez, um pouco de parcimônia não são más ideias.

As fontes que usamos comumente contam com tipos como:

- O *itálico*, geralmente usado para enfatizar ideias.
- O **negrito**, ou **bold**, muitas vezes usado para chamar a atenção do leitor.
- Os **VERSALETES**, que são letras em estilo de maiúscula, mas com a mesma altura do corpo da fonte. Uma boa ideia é usá-los em siglas, como IBGE, BC, 3 AM etc. Em nomes próprios e acrônimos geográficos, geralmente usamos maiúsculas como JRR Tolkien.
- Os **tipos monoespaçados** são ótimos para dar exemplos de código-fonte ou nomes de arquivos, como `fontes.tex`.

### 7.2 Tamanhos de fontes

Como já vimos, certos comandos como `\section` e `\chapter` escolhem o tamanho e espaçamento adequados para que nosso texto pareça organizado e fluido. Essas decisões são tomadas pelos designers das classes que usamos e baseadas em **escalas** tipográficas. Nas raras ocasiões em que precisamos diminuir **OU** **aumentar** nosso texto, temos os seguintes comandos à nossa disposição:

- `\tiny`
- `\scriptsize`
- `\footnotesize`
- `\small`
- `\normalsize`
- `\large`
- `\Large`
- `\LARGE`
- `\huge`
- `\Huge`

É possível conter nosso texto entre duas chaves para que apenas uma seção seja afetada pelo comando de tamanho.

### 7.3 Exercício: fontes

Vamos resolver o problema no arquivo `exercicios/sonhos-noite-verao.tex`.

## 8 Layouts de página

Você deve ter notado que este documento usa uma página A4 e as margens estão enormes. Vamos carregar o pacote `showframe` para deixar isso mais claro. Há um motivo para tanta margem: quando lemos uma linha longa demais, perdemos a noção de onde ela havia começado. O tamanho de linha ideal fica por volta de 66 caracteres, incluindo espaços. Esse é o mesmo motivo pelo qual jornais são divididos em diversas colunas. Para resolver esse problema das margens, existem algumas soluções:

- Dividir o texto em duas colunas (opção de classe `twocolumn`)
- Carregar o pacote `fullpage`
- Carregar o pacote `fullpage` com espaçamento grande entre as linhas

Para arrumar o espaçamento entre as linhas, devemos utilizar o pacote `setspace`. Ele vem com os seguintes comandos:

- `\singlespacing`
- `\onehalfspacing`
- `\doublespacing`

Ao usar o comando `\onehalfspacing`, por exemplo, o documento seguirá esse espaçamento até que outro espaçamento seja especificado.

Outro fator que influencia o layout da página é seu estilo. O `LATEX` vem com dois comandos, `\pagestyle{}` e `\thispagestyle{}`, que aceitam os seguintes argumentos:

**empty** Sem texto no cabeçalho e no rodapé

**plain** Cabeçalho limpo, mas o número da página aparece centralizado no rodapé

**headings** Rodapé limpo, informações como o nome da seção e número da página aparecem no cabeçalho



## 9 Posicionamento do texto

Para determinar a posição horizontal do texto, precisamos encontrar nosso primeiro ambiente. Na verdade, um dos primeiros construtos que encontramos foi o ambiente `document`, delimitado por dois comandos: `\begin` e `\end`.

### 9.1 Ambientes

O ambiente `center`, com o nome sugere, se encarrega de centralizar texto na página:

Todo este texto será centralizado na página.

De maneira similar, os ambientes `flushleft` e `flushright` alinham texto ao lado esquerdo e direito do papel, respectivamente:

Esse parágrafo será alinhado à esquerda e não será justificado. Mais abaixo, veremos como ajustar a posição de um texto dentro da própria linha.

Também podemos alinhar texto à direita. Além desses ambientes, podemos mudar o alinhamento do texto usando os comandos `\centering`, `\flushleft` e `\flushright`.

### 9.2 Controlando o espaço na linha

Os comandos `\hspace{comprimento}` e `\hfill` nos permitem controlar o espaço dentro de uma linha:

Aqui haverá                      um espaço.

Algumas unidades que o  $\text{\LaTeX}$  reconhece são:

- `mm`
- `cm`
- `in`
- `pt`
- `em` (comprimento da letra “m”)
- `ex` (altura da letra “x”)
- `\textheight` e `\textwidth` (altura e comprimento da corpo do texto)
- `\pageheight` e `\pagewidth` (altura e comprimento da página toda)

Ainda é possível utilizar o comando `\hfill`, que preenche todo o espaço disponível na linha:

Começo    meio    fim

### 9.3 Espaço vertical

Os comandos para controle do espaço vertical, isto é, o espaço entre os parágrafos, são muito parecidos com os comandos de espaço horizontal. Eles são `\vspace{comprimento}` e `\vfill`.

Espaços entre parágrafos.

## 10 Listas

O  $\text{\LaTeX}$  vem com três ambientes para criar listas: `itemize`, `enumerate` e `description`.

- O ambiente `itemize` é geralmente usado para listas cuja ordem não é importante.
- A numeração que listas do tipo `enumerate` trazem pode indicar os passos necessários para completar uma tarefa, ou sua ordem de importância.
- A lista do tipo `description` é excelente para explicar conceitos relacionados. Que oportunidade perdida de usá-la!

Listas de descrição têm uma sintaxe um pouco diferente:

**Bit** Abreviação de *binary digit*, um bit pode tomar o valor de 1 ou 2, apenas.

**Byte** Uma unidade de informação digital que geralmente tem oito bits. O “y” foi escolhido de propósito para evitar que fosse acidentalmente confundido com “bit”.

### 10.1 Customizando o ambiente `enumerate`

Uma maneira muito elegante de customizar listas ordenadas é o pacote `enumitem`. Ele modifica o ambiente `enumerate` para aceitar um argumento adicional — uma string que contenha os valores `\Alph*`, `\alph*`, `\Roman*`, `\roman*` ou `\arabic*` — permitindo customizar a lista facilmente:

- A) Tales de Mileto
- B) Pitágoras
- C) Xenófanes
- D) Empédocles
- E) Aristóteles

### 10.2 Exercício: receita usando listas

### Receita de Panqueca

Receita de Panqueca do livro *Cooking for geeks*, de Jeff Potter

Em uma tigela, pese e misture:

Em uma tigela separada e que possa ir ao micro-ondas, derreta:

Adicione à manteiga e bata para incorporar completamente:

Derrame os ingredientes sólidos sobre os líquidos e bata até que os ingredientes estejam quase que completamente incorporados. Bolinhas de farinha não são um problema: é melhor evitar bater a massa demais, para minimizar a formação de glúten a partir de duas proteínas, a glutenina e a gliadina, que estão presentes na farinha (elas se ligam formando uma matriz parecida com uma rede).

Coloque uma frigideira não-aderente no fogo médio-alto. Espere até que a panela esteja quente. O teste padrão é jogar algumas gotas de água na panela e observar se chium; o teste geek é usar um termômetro infravermelho e checar se a panela chegou a 200 ° C. Usando uma concha, copo de medida ou colher de sorvete, despeje cerca de meio copo de massa na panela. Enquanto o primeiro lado cozinha, é possível observar bolhas se formando na superfície de cima da panqueca. Vire a panqueca após que as bolhas comecem a se formar, mas antes que estourem (cerca de dois minutos).

## 11 Tabulares e tabelas

Tabelas são difíceis de escrever em qualquer tipo de editor de texto e, embora o  $\text{\LaTeX}$  tenha ferramentas para lidar com esse tipo de construto textual, não devemos tentar imitar a abordagem de programas WYSIWYG.

No  $\text{\LaTeX}$ , assim como na tradição tipográfica, há uma distinção entre uma tabela “formal” (`table`), que é legendada e numerada e uma tabulação “informal” (`tabular`), que é apenas uma disposição de texto alinhado em linhas e colunas. Por exemplo:

1	2	3
4	5	6
7	8	9

O ambiente `tabular` não deve ser encarado simplesmente como uma maneira de fazer tabelas, mas primeiramente de alinhar textos na horizontal e na vertical. Os valores de alinhamento básicos são `c` (centro), `l` (esquerda) e `r` (direita). É possível, também, especificar linhas verticais com `|` e parágrafos com tamanhos definidos com `p{comprimento}` (alinhado ao topo), `m{comprimento}` (alinhado no meio) e `b{comprimento}` (alinhado em baixo). Linhas horizontais podem ser especificadas com `\hline`:

1	2	3
4	5	6
7	8	9

Linhas, sejam elas horizontais ou verticais, devem ser usadas com moderação. O objetivo da tabela é passar informação, portanto o texto deve ser o enfoque central. É melhor deixar que a informação respire, do que cercá-la. Nas palavras de Robert Bringhurst, em *Elementos do Estilo Tipográfico*:

Assim como o texto, as tabelas ficam canhestras quando abordadas de forma puramente técnica. Boas soluções tipográficas não costumam surgir em resposta a perguntas do tipo “Como posso enfiar essa quantidade de caracteres naquele tanto de espaço?”. (p. 81)

## 11.1 Exemplos de tabelas

### 11.1.1 Ingredientes básicos para o pão

Apenas quatro ingredientes são necessários para fazer pão. Vejamos uma receita genérica:

Ingrediente	Quantidade
Trigo	400g
Água	280g
Sal	8g
Fermento	2g

Antes de correr para a cozinha, é fundamental entender para que servem e como interagem esses ingredientes.

Ingrediente	Função
Trigo	A base do pão. Sem trigo, sem pão.
Sal	Retarda o fermento e controla o processo de fermentação. Também adiciona o sabor que todo esperamos do pão.
Fermento	É comumente vendido seco e precisa ser ativado com água morna. Faz o pão crescer e quanto maior a quantidade, mais rápido é o crescimento. Fermento demais deixa o pão com um gosto de cerveja.
Água	Dissolve os ingredientes (solvente universal, alguém?) e ativa o fermento. Ao adicionar mais água, o resultado é um pão mais pegajoso e com buracos menos regulares. Se houver muito pouca água, a expansão da massa é restrita e o resultado é um pão mais firme, seco e duro.

Imprima o resumo abaixo e cole na parede da cozinha, para nunca esquecer esses princípios básicos:

Ingrediente	Função
Trigo	Base do pão
Sal	Retarda o fermento Controla o processo de fermentação Dá sabor ao pão
Fermento	Vendido seco Ativado com água morna Faz o pão crescer Em abundância, o crescimento é mais rápido
Água	Dissolve os ingredientes Ativa o fermento Em abundância, pão mais pegajoso e com buracos menos regulares Em falta, pão mais firme, duro e seco

### 11.1.2 Distâncias cósmicas

Com uma nave que viaje à velocidade da luz, seriam necessários 4,2 anos para chegar à estrela mais próxima. Abaixo, uma lista de destinos para quem dispõe de tempo livre.

Estrelas na Via Láctea	
Estrela	Distância (anos-luz)
Proxima Centauri	4,2
Rigil Kentaurus	4,3
Alpha Cen B	4,3
Estrela de Barnard	6,0
Wolf 359	7,7
BD +36 2147	8,2
Luyten 726-8A	8,4
Luyten 726-8B	8,4
Sirius A	8,6
Sirius B	8,6
Ross 154	9,4
Ross 248	10,4
Epsilon Eri	10,8
Ross 128	10,9
61 Cyg A	11,1
61 Cyg B	11,1
Epsilon Ind	11,2
BD +43 44 A	11,2
BD +43 44 B	11,2
Luyten 789-6	11,2
Procyon A	11,4
Procyon B	11,4
BD +59 1915 A	11,6
BD +59 1915 B	11,6
CoD -36 15693	11,7

## 11.2 Flutuando com table

Até agora, temos colocado nossas tabelas em meio ao texto usando o ambiente `tabular`. É muito comum, no entanto, colocar tabelas em páginas dedicadas, para que não atrapalhem o fluxo do texto. O  $\text{\LaTeX}$  é capaz de fazer isso usando uma abstração conhecida como `float`.

Em  $\text{\LaTeX}$ , os dois ambientes do tipo `float` mais comuns são `table` e `figure`:

```
\begin{table}[posição]
...
\end{table}
```

As posições possíveis são (o valor padrão é `tbp`):

- h** Aqui (here)
- t** Topo da página
- b** Base da página
- p** Página dedicada a floats
- !** Sobrescreva as restrições de float

Nossa primeira tabela poderia ser reescrita desta maneira (Tabela 2):

1	2	3
4	5	6
7	8	9

Tabela 2: Números de 1 a 9

Três comandos a notar: `\centering` pode ser usado ao invés do ambiente `center`, pois seu escopo estará limitado; `\caption{legenda}` pode ser usado para adicionar uma legenda à tabela e `\label{referencia}` permite que referenciemos a tabela usando `\ref{referencia}`.

## 11.3 Ferramentas para tabelas

As tabelas que discutimos durante esta introdução não precisaram de muito espaço horizontal. No entanto, há ocasiões nas quais desejamos escrever uma tabela com tamanho dinâmico e que tome a página toda. Para isso, existe o pacote `tabularx`, que define um novo ambiente que aceita o alinhamento `X`, que é flexível.

Como em tudo em  $\text{\LaTeX}$ , quantidade de pacotes e detalhes que podemos discutir é grande demais para este pequeno workshop. Deixo aqui alguns links úteis:

- [Tutorial do Wikibooks](#), do qual tiramos muitas ideias

- [Descrição de vários pacotes para tabelas, seus usos e conflitos](#)
- [Lista de ferramentas para ajudar a criação de tabelas](#)
- [Table Generator](#)
- [Table Editor](#)

## 12 Imagens

É possível importar gráficos em formatos como `png` e `jpg` usando o pacote `graphicx`. Ele inclui comandos para redimensionar e rotacionar textos e gráficos, bem como o comando `\includegraphics`.

```
\includegraphics[opções]{imagem}
```

O comando aceita uma série de opções. Os mais comuns são:

**width e height** Para controlar o comprimento e altura da imagem. Aceita valores como `\textwidth` e `\pageheight`

**keepaspectratio** Um valor booleano (`true` ou `false`)

**scale** Por exemplo, o valor de 0.5 reduz a imagem pela metade

Geralmente, usamos o ambiente `figure`, um float como o `table`:

```
\begin{figure}
  \centering
  \includegraphics{imagem}
  \caption{Uma imagem de exemplo}
  \label{fig:imagem}
\end{figure}
```

### 12.1 Exemplo de imagem: nossos vizinhos

Na Figura 1, podemos ver nossos vizinhos mais próximos, por assim dizer. A estrela circulada é Proxima Centauri, uma pequena anã vermelha, que pode estar ligada gravitacionalmente às outras duas estrelas. A olho nu, as duas estrelas parecem uma só.



Figura 1: Alpha Centauri e Beta Centauri, com Proxima circulada



## 13 A classe `abntex2`

A descrição oficial do `abnTeX2` é:

O `abnTeX2`, evolução do `abnTeX` (ABsurd Norms for  $\text{T}_{\text{E}}\text{X}$ ), é uma suíte para  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  que atende os requisitos das normas da ABNT (Associação Brasileira de Normas Técnicas) para elaboração de documentos técnicos e científicos brasileiros, como artigos científicos, relatórios técnicos, trabalhos acadêmicos como teses, dissertações, projetos de pesquisa e outros documentos do gênero.

A suíte `abnTeX2` é composta por uma classe, por pacotes de citação e de formatação de estilos bibliográficos, por exemplos, modelos de documentos e por uma ampla documentação.

Para utilizar o `abnTeX2`, devemos utilizar a classe `abntex2`. Ela implementa comandos e ambientes novos, como por exemplo:

- `\titulo`
- `\autor`
- `\orientador`
- `\instituicao`
- `\imprimircapa`
- `citacao` (ambiente)
- `resumo` (ambiente)

### 13.1 Exemplo: adaptando este documento

Vamos adaptar este documento para a classe `abntex2`. Para isso, precisaremos:

- Mudar a classe de `article` para `abntex2` e adicionar a opção `article`
- Alterar os comandos de metadado usando os comandos apropriados em português
- Substituir `\maketitle` por `\imprimircapa` ou algo parecido.

## 14 Bibliografias

Bibliografias em  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  são organizadas em um arquivo `.bib` externo ao arquivo que contém o texto. Esse arquivo `.bib` será alimentado (manual ou automaticamente) com entradas bibliográficas, como por exemplo:

```
@book{chomsky1995,  
  title={The Minimalist Program},  
  author={Chomsky, N.},  
  series={Current studies in linguistics series},  
  publisher={MIT Press},  
  address={Cambridge},  
  year=1995  
}
```

No arquivo principal, no local em que desejamos incluir a bibliografia, usamos o comando `\bibliography{arquivo}`. No decorrer do texto, podemos utilizar os comandos `\cite[p.~20]{greenwade93}` e `\citeonline`, por exemplo, para fazer referência à entrada bibliográfica desejada. Um arquivo `bst` fica responsável pelo estilo correto da citação e da bibliografia. O pacote `abntex2cite` implementa um estilo `bst` que segue as determinações da ABNT.

## 14.1 Exemplo: bibliografia e citação

Para incluir uma bibliografia a este arquivo, basta descomentar o pacote `abntex2cite` no preâmbulo, o comando `\bibliography` e o parágrafo a seguir.

# 15 Macros

Uma das maiores vantagens do  $\text{\LaTeX}$  em relação aos outros editores de texto é a sua extensibilidade. É possível adicionar funcionalidades ao sistema por meio de macros. O próprio  $\text{\LaTeX}$  não passa de um conjunto (bastante complexo) de macros do  $\text{\TeX}$ .

Essas macros são programas que automatizam certas funções e permitem que o autor foque em escrever, ao invés de realizar tarefas tediosas repetidamente.

## 15.1 Macros de substituição

O tipo mais simples de macro é o de substituição:

```
\newcommand{\forma}{\textsf{ForMA}}
```

Declarações do tipo `\newcommand` costumam ficar no preâmbulo do documento, para estarem disponíveis por todo o código. Após declarar o comando anterior, podemos usar `\forma` com a garantia de que o texto nunca terá erros de digitação.

Ao utilizar esta macro, é necessário incluir um espaço antes do resto do texto, ou o espaço em branco será engolido: `\forma{} texto`. Isso ocorre, pois o  $\text{\LaTeX}$  está esperando um argumento. No entanto, é possível carregar o pacote `xspace`, que adiciona esse espaço de maneira automática:

```
\usepackage{xspace}
\newcommand{\forma}{\textsf{ForMA}\xspace}
```

## 15.2 Macros com argumentos

Já encontramos muitos comandos que levam argumentos, como por exemplo `\textbf{}`. Digamos, por exemplo, que nosso texto é repleto de palavras em inglês, que desejamos diferenciar do resto do texto usando itálico (ou romanas, quando o texto ao redor for itálico) e, além disso, garantir que serão hifenizadas de acordo com as regras da língua inglesa, quando ocorrerem ao fim de uma linha. Podemos declarar um comando `\eng{}` facilmente, da seguinte maneira:

```
\newcommand{\eng}[1]{%
  \emph{\foreignlanguage{english}{#1}}%
}
```

Onde 1 indica a quantidade de argumentos que nosso comando irá receber e #1 será substituído pelo argumento provido pelo usuário:

Por exemplo, este texto contém `\eng{some words in English that might be hyphenated correctly}` caso o texto quebre naquela parte.

Por exemplo, este texto contém *some words in English that might be hyphenated correctly* caso o texto quebre naquela parte.

### 15.3 Novos ambientes

Também é possível definir um ambiente novo:

```
\newenvironment{nome}[num]{antes}{depois}
```

Por exemplo:

```
\newenvironment{spotlight}
{
  \vspace{1em}
  \noindent\rule{\linewidth}{.5pt}\par\vspace{5pt}
  \noindent\ignorespaces
}
{
  \par
  \noindent\rule{\linewidth}{.5pt}
  \vspace{1em}\ignorespacesafterend
}
```

Veja o resultado do ambiente `spotlight`:

---

O cuidado em identificar pontos críticos no entendimento das metas propostas pode nos levar a considerar a reestruturação dos conhecimentos estratégicos para atingir a excelência. Todavia, a valorização de fatores subjetivos talvez venha a ressaltar a relatividade das formas de ação.

---

## 15.4 Exercício: automatizando com macros

(Crie os comandos necessários e então descomente o texto abaixo.)

## 16 Linguística

Os linguistas usuários de L<sup>A</sup>T<sub>E</sub>X já escreveram uma série de pacotes que ajudam em tarefas de tipografia comuns à nossa área. Veremos abaixo como escrever transcrições fonéticas com o IPA de maneira fácil; como inserir árvores sintáticas que não são imagens e por isso não perdem resolução ao serem impressas; e como inserir exemplos, glosas e fazer referências a esses elementos no texto.

### 16.1 Transcrições fonéticas

O **pacote `tipa`** permite transcrever usando o Alfabeto Fonético Internacional (IPA) sem ter que procurar e inserir símbolos a partir de uma lista. Tudo o que for escrito dentro do comando `\textipa{}` será convertido automaticamente, seguindo uma tabela de referência intuitiva (veja o documento `tipachart.pdf`). Por exemplo, para produzir `lɪŋ.ˈɡʷis.təs`, basta digitar:

$$\text{\textit{N.}}^{\text{\textit{g}}}\text{\textit{wis.t5s}}$$

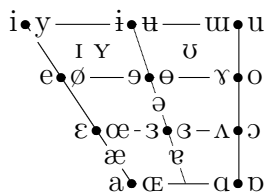
Alguns exemplos de vogais orais e nasais:

[i e ε a ɔ o u]      [ɪ ʊ ʌ]  
[õ ã ẽ]

E também algumas consoantes:

$$\int_3^{\infty} \frac{1}{t^3} dt$$

Ainda é possível inserir diagramas de vogais de maneira bastante simples, usando o pacote `vowel`. O diagrama a seguir é idêntico ao que pode ser encontrado na última versão do IPA:



## 16.2 Digitar transcrições diretamente

Uma alternativa é utilizar a fonte [Doulos SIL](#), que foi especialmente concebida para representar corretamente todos os glifos usados na transcrição. Basta configurá-la no preâmbulo como a fonte para entrada de texto em IPA e digitar a transcrição diretamente no código-fonte:

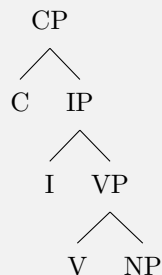
```
% No preâmbulo do documento:  
\newfontfamily\ipafont{Doulos SIL}  
% ...e no corpo do documento (veja o código-fonte, pois esta  
% fonte não é capaz de exibir todos os caracteres):  
\ipafont [ø ã          ħ ]}
```

Resultado: [ʃøõã<sup>yamA</sup>ɪrɿʏ[kʃzɹj]ħɦʒ]

## 16.3 Árvores sintáticas com o forest

Existem muitos pacotes para a inserção de árvores sintáticas no  $\text{\LaTeX}$ , mas provavelmente o melhor e mais poderoso é o [forest](#). Para inserir uma árvore, basta iniciar o ambiente `forest` e inserir uma sentença com os constituintes delimitados por chaves. Ela será convertida automaticamente para uma árvore.

```
\begin{forest}  
  [CP [C] [IP [I] [VP [V] [NP] ] ] ]  
\end{forest}
```



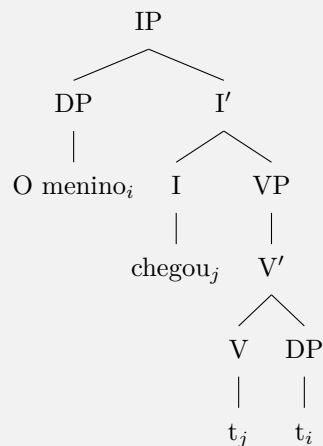
Por motivos de legibilidade do código, devemos pular linhas e indentar os constituintes que estão em níveis mais baixos da árvore (`□` representa um espaço em branco):

```

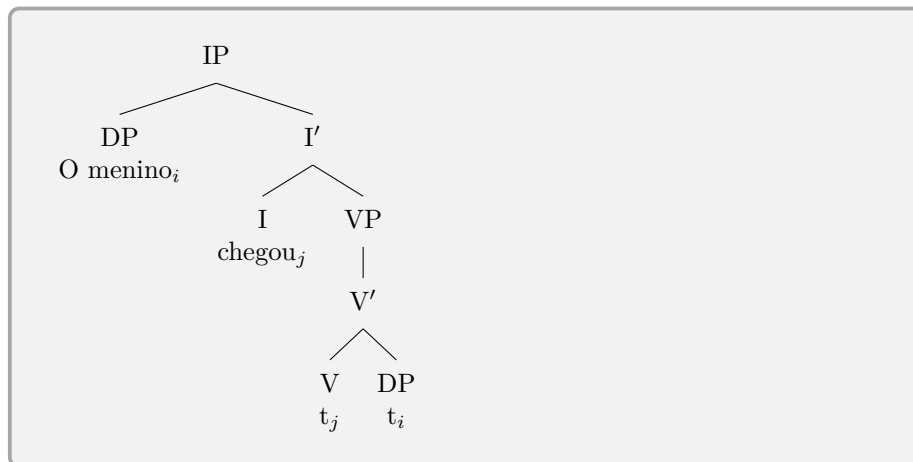
\begin{forest}
  \uu[CP
    \uuuu[C]
      \uuuuuu[IP
        \uuuuuuuu[I]
          \uuuuuuuuu[VP
            \uuuuuuuuuu[V]
              \uuuuuuuuuuu[NP]
                \uuuuuuuuuuu]
              \uuuuuuuuuuu]
            \uuuuuuuuuuu]
          \uuuuuuuuuuu]
        \uuuuuuuuuuu]
      \uuuuuuuuuuu]
    \uuuuuuuuuuu]
  \end{forest}

```

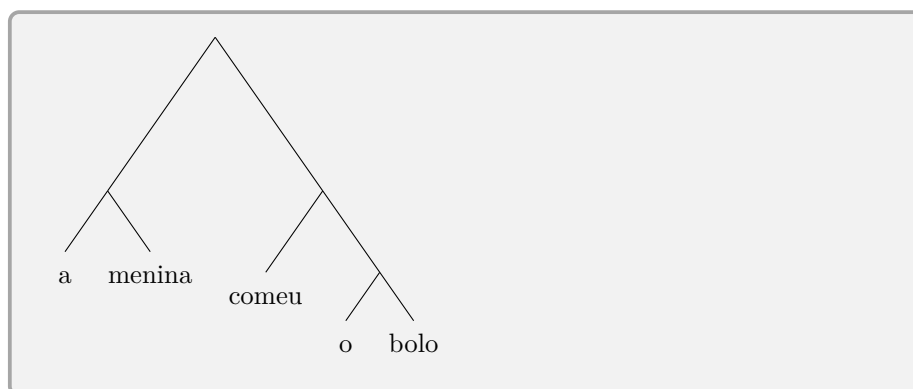
A árvore a seguir (veja o código-fonte) é um exemplo com todos os nós rotulados e com os nós terminais precedidos de linhas. Note como os nós terminais estão marcados como constituintes dentro de outros constituintes, por exemplo em [DP [O menino]<sub>*i*</sub>]. Note, ainda, que  $\$_i\$_$  significa que estamos entrando no modo de matemática ( $\$_\dots\$_$ ) e inserindo um *i* subscrito ( $\$_i\$_$ ):



É bastante usual omitirmos as linhas que levam aos nós terminais, pois algumas pessoas as consideram redundantes. Isso é possível usando uma sintaxe ligeiramente diferente. Note que, agora, há uma quebra de linha e apenas um constituinte em  $\text{[DP}\backslash\backslash\text{ O menino]}$ , por exemplo:

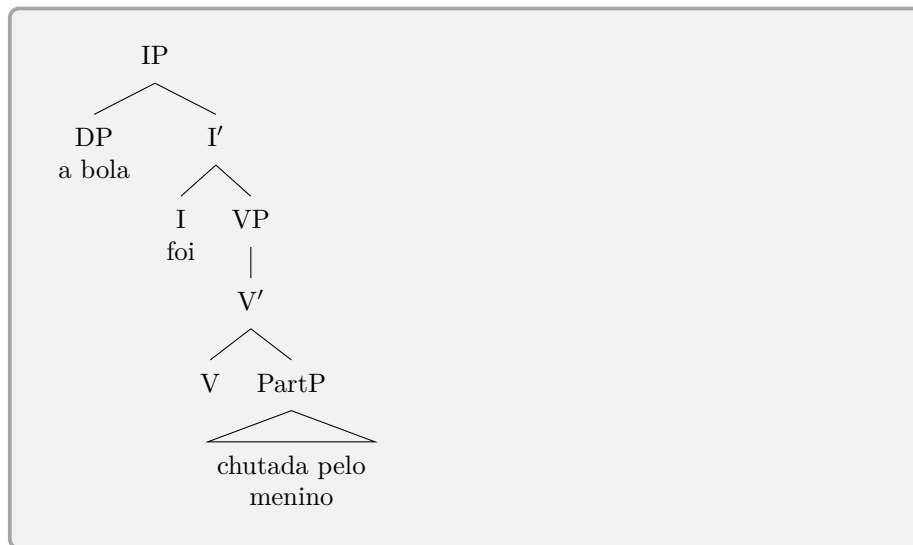


O pacote **forest** é baseado no [PGF/TikZ](#), um conjunto de bibliotecas gráficas para o  $\text{\LaTeX}$ . Como consequência, as árvores aceitam uma série de customizações que não discutiremos no momento. Porém, vejamos um exemplo para ilustrar o que se pode esperar encontrar no manual do **forest**. A árvore a seguir não tem todos os nós rotulados e isso causa um comportamento estranho quando a árvore é disposta na página. No entanto, podemos usar a opção `for tree = nice empty nodes` para remediar a situação:



Essa opção, porque se encontra dentro do ambiente **forest**, irá se aplicar somente para a árvore em questão. Veja o manual do pacote para mais opções.

Às vezes, desejamos omitir informações irrelevantes em alguma árvore. Explicitamos nossa decisão desenhando um triângulo sobre o constituinte que não foi esmiuçado. Para isso, usamos a opção **roof**, que se aplica a um constituinte em nossa árvore. Veja no exemplo a seguir, onde o constituinte [chutada pelo menino] está simplificado:



## 16.4 Exemplos numerados e glosas com o `linguex`

Outra tarefa especialmente comum para o linguista é apresentar seus dados em formato de exemplos de sentenças, que são numeradas à esquerda. Não existe uma implementação perfeita em  $\text{\LaTeX}$  para isso, pois cada uma tem suas vantagens e defeitos. O `linguex` é um desses pacotes com suporte a exemplos e glosas. Você pode aprender sobre outros pacotes para linguistas [nesse artigo da WikiBooks](#), caso o `linguex` não tenha as funcionalidades de que precisa.

Escolhemos o `linguex` pois sua sintaxe é relativamente simples e poderosa. Vejamos alguns exemplos básicos:

- (1) Chegou o menino.
- (2) Chegou a carta.
- (3) \*A chegou carta.

Sempre deixe uma linha em branco para indicar o fim de um exemplo. É possível mostrar também subexemplos:

- (4)
  - a. Chegou o menino.
  - b. O menino chegou.

Novamente, certifique-se de colocar uma linha em branco ao final dos exemplos.

Também é possível escrever glosas usando o comando `\exg`. As palavras serão alinhadas automaticamente:

- (5)
 

Gila abur-u-n	ferma hamišaluğ	güğüna amuq'-da-č.
now they-OBL-GEN farm	forever	behind stay-FUT-NEG
“Now their farm will not stay behind forever.”		

### 16.4.1 Referências cruzadas

Já vimos que os comandos `\label` e `\ref` permitem criar referências que são automaticamente numeradas e atualizadas pelo  $\text{\LaTeX}$ . Esses comandos também



funcionam nos exemplos e glosas do linguex:

(6) Chegou o menino.

(7) \* A chegou carta.

Em (6), temos uma sentença bem formada, ao contrário de (7), que é uma sentença agramatical.

Isso é tudo,  
pessoal! 😁