

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

Giovane de Oliveira Torres

Instituto Federal Sul-Riograndense  
Câmpus Pelotas  
Tecnologia em Sistemas para Internet

*ggiovaneotorres@gmail.com*

7 de Novembro de 2018

# Motivação

- Alternativa para escrita de relatórios, artigos, TCCs, dentre outros documentos acadêmicos
- Abstrai a formatação para quem escreve o texto
- Referenciação de seções, figuras e bibliografia

# Como começamos?

- *Software* essencial para construção de documentos em LaTeX
  - Miktex<sup>1</sup>
    - Comumente utilizado no Windows
  - Texlive<sup>2</sup>
    - Comumente utilizado em sistemas baseados em Linux

---

<sup>1</sup><https://miktex.org/>

<sup>2</sup><https://www.tug.org/texlive/>

# Como começamos?

- Tendo um dos programas citados, podemos utilizar **qualquer** editor de texto
  - Basta saber como compilar o arquivo fonte
- Porém, existem alternativas

- Podemos utilizar IDEs que facilitam um pouco o trabalho
- IDEs contam com *syntax highlighting*, métodos de compilação em um clique, facilidades para utilizar os comandos a serem vistos, dentre outros
  - **Texmaker** <sup>3</sup> (recomendado)
    - Funciona em Windows, Linux e MacOSX
    - *Cross-platform*
  - Texworks <sup>4</sup>
  - Kile <sup>5</sup>

---

<sup>3</sup><http://www.xmlmath.net/texmaker/>

<sup>4</sup><http://www.tug.org/texworks/>

<sup>5</sup><https://kile.sourceforge.io/>

- Podemos usar editores web
- Apresentam suporte à escrita colaborativa
  - Overleaf <sup>6</sup>
  - ShareLaTeX <sup>7</sup>

---

<sup>6</sup><https://www.overleaf.com/>

<sup>7</sup><https://www.sharelatex.com/>

# Iniciando um documento no formato .tex

- Criar um arquivo .tex
- Deve ser necessário alguns comandos

# Iniciando um documento no formato .tex

```
\documentclass{article}  
  
\begin{document}  
    Ola mundo  
\end{document}
```



# Iniciando um documento no formato .tex

```
\documentclass{article}

\begin{document}
    Ola mundo
\end{document}
```

- **\documentclass{article}**
  - Define a classe de documento que será utilizada
  - `article`: Artigo
  - Existem diversas opções que permite diferentes formatos de documentos

# Iniciando um documento no formato .tex

```
\documentclass{article}

\begin{document}
  Ola mundo
\end{document}
```

- `\begin{document}` e `\end{document}`
  - Delimita o escopo de conteúdo do documento
    - Também chamado de corpo do documento
  - O que vem antes de `\begin{document}` é chamado de **preâmbulo** do documento

# Preâmbulo do documento

- Primeira metade de um arquivo .tex
  - **Sempre** antes do texto que irá compor o documento
- Múltiplas funcionalidades
  - Tipo de documento
  - Informações para formatar o documento corretamente
  - Carregamento de pacotes que auxiliam para algumas especificidades

# Corpo do documento

- Segunda metade do arquivo .tex
- Contém todas as informações referentes a conteúdo do documento
  - Texto bruto
  - Comandos para formatação de texto
  - Inserção de elementos adicionais
    - Figuras, tabelas, fórmulas matemáticas, ...

# Localidade

- Por padrão, o LaTeX por si só não é suficiente
- Precisamos estabelecer a localidade para a escrita correta de textos
  - Textos em português: Apresetam caracteres com acentuação (ã, ê, í, etc.)

# Localidade

- Adaptar o LaTeX para a localidade que desejamos
  - Suporte a diversas linguagens espalhadas pelo globo
- Modificar para o LaTeX conseguir interpretar os acentos corretamente
  - além de alterar a localidade para português brasileiro.
  - **Preâmbulo** do documento

```
\documentclass{article}

\usepackage[brazil]{babel}           % Comando para pôr a
    localidade português brasileiro
\usepackage[utf8]{inputenc}         % Comando para reconhecer
    entradas utilizando a codificação UTF-8

\begin{document}
(...)
```

- **\usepackage[brazil]{babel}**
  - Datas e palavras fornecidas pela formatação do LaTeX são traduzidas para português brasileiro
  - Se desejável, pode-se trocar a opção entre colchetes para a linguagem que quiser
    - english, spanish, ...

```
\documentclass{article}

\usepackage[brazil]{babel}           % Comando para pôr a
    localidade português brasileiro
\usepackage[utf8]{inputenc}         % Comando para reconhecer
    entradas utilizando a codificação UTF-8

\begin{document}
(...)
```

- **\usepackage[utf8]{inputenc}**
  - Mostra corretamente caracteres com acento, cedilha, dentre outros
  - Tecnicamente, faz com que a codificação lida como entrada pelo LaTeX seja UTF-8, a qual tem suporte a vários caracteres (inclusos os utilizados no português)



```
\documentclass{article}

\usepackage[brazil]{babel}           % Comando para pôr a
    localidade português brasileiro
\usepackage[utf8]{inputenc}          % Comando para reconhecer
    entradas utilizando a codificação UTF-8

\begin{document}
(...)
```

- **\usepackage[utf8]{inputenc}**
  - Podem ser usados outros tipos de codificação, como `latin1` em vez de `utf8`, a qual corresponde à codificação ISO 8859-1
  - **Recomendação pessoal:** Usar UTF-8

# Classes

- Para um arquivo ser formatado corretamente, faz-se uso de classes
- Algumas classes pré-definidas
  - `article`, `book`, ...
- Classes construídas por usuários
  - Arquivo `.cls`

- Define uma classe de arquivo a qual representa uma formatação definida
- Utiliza a linguagem TeX para desenvolver
  - Linguagem complexa e de difícil entendimento

# Classe do TCC do TSI-IFSul

- **Caso de exemplo:** Classe do documento de TCC do TSI-IFSul Campus Pelotas
  - Segue a mesma formatação do modelo de documento fornecido em .doc
- Disponível em um repositório do GitHub:
  - <https://github.com/gdotorres/tsi-ifsul-tcc>

- Arquivo **textsi.cls**: Define a formatação geral do documento
  - Margens, fonte, espaçamento, ...
  - Define usos de alguns pacotes
    - Requeridos para o desenvolvimento desta classe
  - Desenvolvido na linguagem TeX
- Arquivo **exemplo-tcc.tex**: Pacotes adicionais e corpo do documento
  - Focaremos principalmente neste arquivo
- Arquivo **exemplo-tcc.bib**: Referências bibliográficas
  - Será visto com mais detalhes adiante

- **exemplo-tcc.tex** é composto de conteúdo com comentários indicando para que serve cada linha ou conjunto de linhas de comando

# Comentários?

```
% Começo de um documento .tex  
\documentclass{article}  
(...)
```

- Estamos utilizando uma linguagem de programação para descrever nosso documento
- Como linguagem de programação, ela fornece meio para comentários
  - Utilizamos o caractere %
  - Somente uma nova linha no código para desfazer o comentário
  - **Não** existe comentário em bloco

# Inserção de texto

- Para inserir texto, é extremamente simples
- Basta digitá-lo dentro do corpo de documento
- Entre `\begin{document}` e `\end{document}`



# Inserção de texto

```
(...)  
\begin{document}  
(...)  
    Escrever um texto em LaTeX é simples como testar  
        todos os caracteres como "The quick brown fox  
        jumps over the lazy dog".  
(...)  
\end{document}
```

# Inserção de texto

- O próprio LaTeX acha o melhor meio de fazer o texto caber dentro das margens
  - Utiliza o espaçamento adequadamente
  - Consegue fazer a quebra de palavras utilizando separação de sílabas para melhor disposição do texto em uma linha
- Porém um texto pode apresentar outras características
  - Como fazer um novo parágrafo?
  - Uma nova linha?

# Parágrafos

- Diferentes métodos para adição de parágrafos
  - ① Inserção de uma linha em branco entre dois parágrafos.
  - ② Comando `\par`

# Parágrafos

(...)

`\begin{document}`

(...)

Escrever um texto em LaTeX é simples como testar todos os caracteres como "The quick brown fox jumps over the lazy dog".

Existe uma linha em branco acima da primeira linha, logo esta linha será um novo parágrafo.

Note que esta frase não aparece em um novo parágrafo, já que não existe uma linha em branco entre a frase anterior e esta. `\par` Já esta frase irá aparecer como um novo parágrafo, dado que estamos inserindo o comando para fazê-lo.

(...)

`\end{document}`

# Nova linha

- Diferentes métodos para adição de uma nova linha
  - ① Método usual: comando `\\`
  - ② Método alternativo: comando `\\newline`

# Inserção de texto

```
(...)  
\begin{document}  
(...)  
    Escrever um texto em LaTeX é simples como testar  
    todos os caracteres como "The quick brown fox  
    jumps over the lazy dog". \\ Existem duas  
    contrabarras logo após a primeira frase, logo  
    esta frase aparece em uma nova linha.  
(...)  
\end{document}
```

- Nova linha **não** é novo parágrafo
  - Parágrafo tem o recuo à direita em relação a uma nova linha

# Formatação básica

- Somente texto cru não faz muita coisa
- Vários elementos de formatação também se fazem presentes no LaTeX

# Negrito, itálico e sublinhado

Eis que você está digitando um texto e deseja fazer um `\textbf{destaque}`. A palavra "destaque" da frase anterior irá aparecer em negrito.

- `\textbf{texto}`
  - Formata texto em negrito



# Negrito, itálico e sublinhado

Usualmente, utiliza-se texto em itálico para destacar palavras estrangeiras, como `\textit{smartphone}`.

- `\textit{texto}`
  - Formata texto em itálico

# Negrito, itálico e sublinhado

E dá pra `\underline{deixar elementos sublinhados}`.

- `\underline{texto}`
  - Formata texto em sublinhado

# Negrito, itálico e sublinhado

E o que acontece se fizermos uma `\textbf{\textit{salada}}`  
`\textit{\underline{mista}}`?

- Quando fazemos `\comando`, estamos invocando um comando.
- Podemos invocar comandos aninhados
  - `\comando1{\comando2{texto}}`
- Primeiro destaque: Negrito e itálico
- Segundo destaque: Itálico e sublinhado

- A fim de promover uma organização melhor no texto, utilizamos seções
- Estas seções são **usualmente** numeradas
- Caso de exemplo: **Template do TCC-TSI**
  - Cinco níveis de seção
  - Seção primária (1), seção secundária (1.1), . . . , seção quinária (1.1.1.1.1)
- Numeração das seções: feita **automaticamente**

Nível da seção	Comando
Primária	<code>\chapter{nome}</code>
Secundária	<code>\section{nome}</code>
Terciária	<code>\subsection{nome}</code>
Quaternária	<code>\subsubsection{nome}</code>
Quinária	<code>\paragraph{nome}</code>

- As normas do IFSul nos restringem a no máximo seções quinárias
- Usualmente estes comandos são redefinidos nas classes utilizadas

# Secionamento não numerado

- Eventualmente, pode-se haver a necessidade de estruturar uma seção no documento não numerada
- Para isto, fazemos o uso do caractere asterisco (\*) para tal fim
  - Exemplo: `\section*{seção não numerada}`

## Secionamento (outras informações)

- Outras classes de documentos podem definir diferentes estilos de secionamento
- Dentro da linguagem LaTeX são fornecidos até 7 níveis de secionamento

# Secionamento (outras informações)

Nível da seção	Comando
Primária	<code>\part{nome}</code>
Secundária	<code>\chapter{nome}</code>
Terciária	<code>\section{nome}</code>
Quaternária	<code>\subsection{nome}</code>
Quinária	<code>\subsubsection{nome}</code>
Senária	<code>\paragraph{nome}</code>
Setenária	<code>\subparagraph{nome}</code>

- Se realmente quiséssemos ir a fundo, podíamos definir outros comandos para estabelecer **mais níveis de secionamento**



# Referência à seções

- Eventualmente, podemos chegar a um ponto que precisamos fazer referência à uma determinada seção
- Exemplo clássico: Dentro da introdução de um TCC, falamos sobre como as demais seções estão organizadas no documento.
- Esta referenciação pode ser feita de maneira **automática**
- Utilizaremos um par de comandos para tal fim: `label` e `ref`

```
\section{Laranjas} \label{s:laranjas}
```

Laranja é laranja por causa da cor laranja ou por causa da fruta laranja?

```
\section{Maracujás} \label{s:maracujas}
```

Uma questão foi feita na Seção~\ref{s:laranjas}. Os maracujás tem a cor amarela.

- `\label{nome do rótulo}`

- Rotula um elemento em LaTeX. Neste caso, estamos utilizando para uma `section`

# Referência à seções

```
\section{Laranjas} \label{s:laranjas}
```

Laranja é laranja por causa da cor laranja ou por causa da fruta laranja?

```
\section{Maracujás} \label{s:maracujas}
```

Uma questão foi feita na Seção~\ref{s:laranjas}. Os maracujás tem a cor amarela.

- **\ref{nome do rótulo}**

- Referencia um elemento em LaTeX previamente rotulado. O LaTeX encarrega-se de mostrar a numeração (neste caso) da seção corretamente

# Estruturas de Listas

- Em alguns casos podemos fazer uso de estruturas de listas para enumerar uma sequência de itens
- Algo como está sendo visualizado nesta apresentação
- Existem três tipos de estruturas de listas
  - Não numeradas
  - Enumeradas
  - Descritivas

# Listas não numeradas

```
\begin{itemize}  
  \item Um item  
  \item Outro item  
\end{itemize}
```

- `\begin{itemize}` e `\end{itemize}`
  - Denomina um **ambiente** de nome `itemize`, o qual permite construir listas não numeradas
  - Todo tipo de comando onde for utilizado neste estilo (`begin` e `end`) é chamado de ambiente em LaTeX

# Listas não numeradas

```
\begin{itemize}  
  \item Um item  
  \item Outro item  
\end{itemize}
```

- **\item**

- Determina um item de uma lista não numerada
- Este comando **deve** estar dentro de um ambiente que permite construir listas! (Ex: `itemize`)

# Listas enumeradas

```
\begin{enumerate}  
  \item Primeiro item enumerado  
  \item Segundo item enumerado  
\end{enumerate}
```

- **\begin{enumerate}** e **\end{enumerate}**
  - Denomina um ambiente de nome `enumerate`, o qual permite construir listas enumeradas

# Listas enumeradas

```
\begin{enumerate}  
  \item Um item  
  \item Outro item  
\end{enumerate}
```

- **\item**

- Determina um item de uma lista enumerada
  - Neste caso é enumerada porque está **dentro do ambiente** `enumerate`
- O LaTeX efetua a numeração da lista automaticamente



# Listas enumeradas manipuladas

- A priori, temos listas com números
- Podemos ir além
- Uso de um pacote `enumitem`
  - `\usepackage{enumitem}`
  - Permite fazer manipulação nas listas de modo a permitir listas ordenadas por caractere (a, b, ...), números romanos (i, ii, ...), dentre outros

# Listas enumeradas manipuladas

```
(...)  
\usepackage{enumitem}  
(...)  
\begin{document}  
(...)  
\begin{enumerate}[label=\roman*]  
  \item Um item  
  \item Outro item  
\end{enumerate}
```

- **[label=formato]**

- Determina que uma lista enumerada utilize um determinado formato
- Neste caso, estamos utilizando `\roman*`, o qual formata os itens de uma lista do tipo `enumerate` para números romanos

# Listas enumeradas manipuladas

- Argumento para formatação de lista enumerada é opcional
  - Qualquer argumento que vier dentro de colchetes é considerado pela linguagem do LaTeX como argumento opcional
- Se nada é passado para `\begin{enumerate}` o padrão é utilizado (lista enumerada numericamente)

# Formatos de listas enumeradas

Comando	Formato
<code>\alph</code>	a. b. c.
<code>\Alph</code>	A. B. C
<code>\arabic</code>	1. 2. 3.
<code>\roman</code>	i. ii. iii.
<code>\Roman</code>	I. II. III

# Listas descritivas

```
\begin{description}  
  \item [Banana] Uma fruta  
  \item [Maçã] Outra fruta  
\end{description}
```

- **\begin{description}** e **\end{description}**
  - Denomina um ambiente de nome `description`, o qual permite construir listas descritivas

# Listas descritivas

```
\begin{description}  
  \item [Banana] Uma fruta  
  \item [Maçã] Outra fruta  
\end{description}
```

- **\item**

- Determina um item de uma lista descritiva
  - Dentro dos colchetes vai a descrição de um item
  - Comumente formatado em negrito

## Mix de listas

```
\begin{itemize}
  \item Um item não numerado
  \begin{enumerate}
    \item Item enumerado-x
    \begin{enumerate}
      \item Item enumerado-y
    \end{enumerate}
  \end{enumerate}
\end{itemize}
```

- Podemos inserir listas dentro de listas
- Somente deve-se tomar cuidado em que nível estamos ao adicionar um item

# Tabelas

- Comumente utilizamos uma tabela para fazer a sumarização de algumas informações
  - Comparar informações
  - Mostra de dados
- Um elemento não tão simples de ser inserido em LaTeX



# Tabelas

```
\begin{table} [!h]
\centering
\caption{legenda}
\begin{tabular}{|c|c|} \hline
a & b \\ \hline
c & d \\ \hline
\end{tabular}
\source{fonte}
\end{table}
```

a	b
c	d

# O ambiente `table`

```
\begin{table}[!h]
\centering
\caption{legenda}
\begin{tabular}{|c|c|} \hline
a & b \\ \hline
c & d \\ \hline
\end{tabular}
\source{fonte}
\end{table}
```

- `\begin{table}` e `\end{table}`
  - Define um ambiente de tabela
    - Em especial, elementos como tabela e figura são chamados no LaTeX de *floats*
  - Pode ter como argumento opcional o **posicionamento** do *float* dentro do documento

# Posicionamento de *floats*

- Definido logo após um comando `\begin{"float"}`
  - Exemplo: `\begin{table}[!h]`
- Indica um ou mais possíveis posicionamentos do *float* no LaTeX

Caractere	Significado
h	Posiciona "aqui", sendo colocado aproximadamente no mesmo ponto que ocorre no texto.
t	Posiciona ao topo da página
b	Posiciona na base da página
p	Insere em uma página especial somente para floats
!	Sobrepõe os parâmetros internos que o LaTeX utiliza para determinar "boas" posições para floats

# Centralização de *floats*

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|} \hline  
    a & b \\\hline  
    c & d \\\hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

- **\centering**

- Denota que o *float* utilizado seja centralizado no documento

# Legenda e fonte de *floats*

```
\begin{table}
\centering
\caption{legenda}
  \begin{tabular}{|c|c|} \hline
    a & b \\ \hline
    c & d \\ \hline
  \end{tabular}
\source{fonte}
\end{table}
```

- **\caption{legenda}**
  - Legenda do *float*
- **\source{fonte}**
  - Fonte do *float*
  - Este comando é definido **somente** na classe do TCC do TSI

- No caso da classe do TCC do TSI, o comando `\source{fonte}` pode ser utilizado devido à uma linha de código na classe do documento:
  - `\newcommand{\source}[1]{\\ Fonte: #1}`
- Se adicionarmos tal linha de comando no preâmbulo de nosso documento, poderemos utilizar `\source`
- Mas o que é esta linha de comando ...?
  - De maneira sucinta, estamos criando um novo comando chamado `\source`, o qual requer um argumento. Este comando faz uma quebra de linha, adiciona o texto "Fonte: " e utiliza o argumento fornecido.
  - Ao chamarmos `\source{Autoria própria}`, seria a mesma coisa que inserir `\\ Fonte: Autoria própria`

# O ambiente tabular

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|} \hline  
    a & b \\ \hline  
    c & d \\ \hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

- **\begin{tabular}** e **\end{tabular}**
  - Define um ambiente do tipo *tabular*
  - Permite o desenho/formatação de tabelas

# Parâmetros de tabular

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|} \hline  
    a & b \\ \hline  
    c & d \\ \hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

- O ambiente `tabular` requer um argumento obrigatório, o qual indica vários parâmetros da tabela
  - ① Número de colunas da tabela
  - ② Alinhamento do texto nas colunas da tabela
  - ③ Se a tabela apresenta linhas verticais



# Parâmetros de tabular

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|}\hline  
    a & b \\\hline  
    c & d \\\hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

## ① Número de colunas da tabela

- Caso de exemplo: Temos **dois** caracteres alfabéticos. Assim, nossa tabela irá conter **duas** colunas
- Outro exemplo: `{|l|c|c|r|}`. Neste caso, há quatro caracteres. Logo, quatro colunas.

# Parâmetros de tabular

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|} \hline  
    a & b \\ \hline  
    c & d \\ \hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

## ② Alinhamento do texto nas colunas da tabela

- Dependendo do caractere utilizado, este irá definir qual o alinhamento do texto dentro da tabela de cada coluna
  - 1 caractere = 1 coluna

# Alinhamento de texto nas colunas das tabelas

<code>l</code>	Coluna com texto justificado à esquerda
<code>c</code>	Coluna com texto centralizado
<code>r</code>	Coluna com texto justificado à direita
<code>p{tam}</code>	Coluna com tamanho fixo definido em <code>tam</code> . Se o texto ultrapassar o tamanho definido, há uma quebra de linha no texto. O texto é verticalmente alinhado ao topo

# Parâmetros de tabular

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|} \hline  
    a & b \\ \hline  
    c & d \\ \hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

- ③ Presença ou ausência de linhas verticais entre as colunas
- Dois tipos de linhas verticais:
    - |: linha vertical simples
    - ||: linha vertical dupla
  - Se não há especificação de linhas, elas não serão desenhadas na tabela

# Conteúdo da tabela

```
\begin{table}
\centering
\caption{legenda}
  \begin{tabular}{|c|c|} \hline
    a & b \\ \hline
    c & d \\ \hline
  \end{tabular}
\source{fonte}
\end{table}
```

- Dentro do ambiente tabular definimos o conteúdo das tabelas
- Segue um formato:
  - Texto da 1ª coluna & Texto da 2ª coluna & ... & Texto da última coluna

# Conteúdo da tabela

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|} \hline  
    a & b \\ \hline  
    c & d \\ \hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

- Quando chegamos no texto da última coluna dentro de uma linha e desejamos inicializar uma nova linha, utilizamos o comando de nova linha (\\)

# Conteúdo da tabela

```
\begin{table}  
\centering  
\caption{legenda}  
  \begin{tabular}{|c|c|}\hline  
    a & b \\\hline  
    c & d \\\hline  
  \end{tabular}  
\source{fonte}  
\end{table}
```

- Da mesma maneira que podemos inserir linhas verticais ou não, podemos fazê-la com linhas horizontais
- Utilizamos o comando `\hline`

# Como fazer uma tabela destas?

<b>Escalação da Seleção Brasileira Brasil 1x2 Bélgica (06/07/2018)</b>			
Goleiro	GOL	1	Alisson
Defesa	LAD	22	Fagner
	ZAG	2	Thiago Silva
	ZAG	3	Miranda
	LAE	12	Marcelo
Meio-campo	VOL	17	Fernandinho
	VOL	15	Paulinho
	MEI	11	Philippe Coutinho
Ataque	ATA	19	Willian
	ATA	10	Neymar
	CTA	9	Gabriel Jesus



# Como fazer uma tabela destas?

```
\begin{table}[!t]
\centering
\caption{A escalação brasileira naquele dia}
\begin{tabular}{|c|l|l|l|} \hline
\multicolumn{4}{|c|}{\textbf{\begin{tabular}{c}{@{}c@{}}
    Escalação da Seleção Brasileira\\ Brasil 1x2 Bélgica
    (06/07/2018)\end{tabular}}}} \\ \hline
Goleiro & GOL & 1 & Alisson \\ \hline
\multirow{4}{*}{Defesa} & LAD & 22 & Fagner \\ \cline
    {2-4}
    & ZAG & 2 & Thiago Silva \\ \cline{2-4}
    & ZAG & 3 & Miranda \\ \cline{2-4}
    & LAE & 12 & Marcelo \\ \hline
(...)
```

# Como fazer uma tabela destas?

```
(...)  
\multicolumn{1}{|l|}{\multirow{3}{*}{Meio-campo}} & VOL &  
17 & Fernandinho \\ \cline{2-4}  
\multicolumn{1}{|l|}{} & VOL & 15 & Paulinho \\ \cline  
{2-4}  
\multicolumn{1}{|l|}{} & MEI & 11 & Philippe Coutinho \\ \hline  
\multirow{3}{*}{Ataque} & ATA & 19 & Willian \\ \cline  
{2-4}  
& ATA & 10 & Neymar \\ \cline{2-4}  
& CTA & 9 & Gabriel Jesus \\ \hline  
\end{tabular}  
\\ \vspace{0.2cm} \small{Fonte: Autoria própria}  
\end{table}
```

# Como fazer uma tabela destas?

- Complexidade de código aumenta substancialmente conforme mais recursos de tabela são utilizados
  - Colunas que ocupam várias colunas, linhas que ocupam várias linhas, diferentes alinhamentos na mesma coluna em linhas diferentes, ...
- Existem facilidades: Você constrói a tabela de maneira intuitiva e o código é gerado automaticamente
  - <https://www.tablesgenerator.com/>
  - <https://truben.no/table/>

# Refência a tabelas

- Vimos como fazer referência à seções, e como o LaTeX consegue lidar com a numeração automaticamente
- Eventualmente um trabalho conterá tabelas as quais precisam ser explicadas durante o texto
- Tabelas são usualmente numeradas, e assim podemos referenciá-las

# Refência a tabelas

```
\begin{table}  
\centering  
\caption{legenda}  
\label{minhatabela}  
    \begin{tabular} (...) \end{tabular}  
\source{fonte}  
\end{table}
```

Na Tabela~\ref{minhatabela}, são mostrados (...)

- Fazemos de maneira parecida com o que ocorre na referência à seções
- O comando `\label` deve ir logo abaixo de `\caption`
- Usamos `\ref` para referenciar a tabela

# Figuras

- Podemos inserir figuras, ilustrações, gráficos, dentre outros no nosso documento
- O ambiente de inserção de figuras é do tipo *float*
  - Tal qual o ambiente `table`, que permite a colocação de tabelas
- Comandos simples, bem diferente do visto em tabelas

```
\begin{figure}[!h]  
\centering  
\caption{Uma bola}  
\includegraphics[scale=0.09]{ibagens/ball.jpg}  
\source{A internet}  
\end{figure}
```

Figura 1: Uma bola



Fonte: A internet

# O ambiente figure

```
\begin{figure} [!h]
\centering
\caption{Uma bola}
\includegraphics[scale=0.09]{ibagens/ball.jpg}
\source{A internet}
\end{figure} [!h]
```

- **\begin{figure}** e **\end{figure}**
  - Define um ambiente de tabela
  - Pode ter como argumento opcional o **posicionamento** do *float* dentro do documento<sup>8</sup>

---

<sup>8</sup>Ver Tabela do *slide* 59



# O ambiente figure

```
\begin{figure}[!h]
\centering
\caption{Uma bola}
\includegraphics[scale=0.09]{ibagens/ball.jpg}
\source{A internet}
\end{figure}
```

- `\centering`<sup>9</sup>, `\caption{legenda}`<sup>10</sup> e `\source{fonte}`<sup>11</sup>
  - Elementos comuns a *floats*, como visto nas tabelas

---

<sup>9</sup>Ver slide 60

<sup>10</sup>Ver slide 61

<sup>11</sup>Ver slides 61 e 62

# Inclusão de imagens

```
\begin{figure}[!h]
\centering
\caption{Uma bola}
\includegraphics[scale=0.09]{ibagens/ball.jpg}
\source{A internet}
\end{figure}
```

- **\includegraphics[opcional]{arquivo}**
  - Faz a inclusão de um arquivo de imagem externo ao LaTeX
  - Em `arquivo` é passado o caminho onde está a imagem que deseja-se mostrar no documento
    - Formatos aceitos: pdf, png e jpg
    - Para aceitar outros formatos, somente com pacotes adicionais

# Inclusão de imagens

```
\begin{figure}[!h]
\centering
\caption{Uma bola}
\includegraphics[scale=0.09]{ibagens/ball.jpg}
\source{A internet}
\end{figure}
```

- **\includegraphics[opcional]{arquivo}**
  - Em `opcional` podem ser passados alguns argumentos optativos, sendo os mais utilizados:
    - `scale=escala`, onde fazemos o redimensionamento da imagem
    - `width=valor`, onde limitamos a largura da imagem
    - Dentre outros

# Inclusão de imagens

```
\usepackage{graphicx}
(...)
\begin{document}
(...)
\begin{figure}[!h]
\centering
\caption{Uma bola}
\includegraphics[scale=0.09]{imagens/ball.jpg}
\source{A internet}
\end{figure}
```

- `\includegraphics[opcional]{arquivo}`
  - Por padrão, o LaTeX **não** tem suporte a este comando
  - Necessariedade de um pacote
    - `graphicx`
    - Incluímos no preâmbulo do documento `\usepackage{graphicx}`

# Refência a figuras

- Vimos como fazer referência à seções e tabelas
- Referencia de figuras = referência de tabelas

# Refência a figuras

```
\begin{figure}[!h]
\centering
\caption{Uma bola}
\label{minhaimagem}
\includegraphics[scale=0.09]{ibagens/ball.jpg}
\source{A internet}
\end{figure}
```

Na Figura~\ref{minhaimagem}, são mostrados (...)

- Utilizamos o par de comandos `\label` e `\ref`, tal qual tabelas<sup>12</sup>

---

<sup>12</sup>Ver slide 77

- Desenvolvimento de bons trabalhos acadêmicos implicam em boas referências bibliográficas
- Latex fornece um esquema de citações e referências bibliográficas
  - Nomeia uma referência para depois citá-la

# Bibliografia do zero (Estilo)

```
\bibliographystyle{acm}  
\bibliography{bibliografia.bib}
```

- **\bibliographystyle{acm}**
  - Define o estilo de bibliografia a ser utilizado.
  - O Latex já fornece alguns como padrão
    - plain, unsrt, abbrv, alpha, ...



# Bibliografia do zero (Estilo)

```
\bibliographystyle{acm}  
\bibliography{bibliografia.bib}
```

- **\bibliographystyle{acm}**
  - Podemos utilizar padrões personalizados
  - Tal qual existem as classes de documentos, também existem classes para bibliografia
    - Comumente fornecidas em arquivos .bst
    - Utiliza uma linguagem específica para formatação de bibliografias, a qual tem uma certa complexidade
  - Classe do TCC do TSI: abnt.bst

```
\bibliographystyle{acm}  
\bibliography{bibliografia.bib}
```

- **\bibliography{arquivo}**
  - Informa qual o arquivo externo que fornece a bibliografia
  - Geralmente é atribuído a extensão .bib
  - Este arquivo segue um formato diferente
    - **Bibtex**

```
@article{rodriguez1985consideraciones,  
  title={Consideraciones relativas a la actuaci{\o}n y l  
    {\i}mites de las oposiciones fonol{\o}gicas  
    interrupto/continuo y tenso/flojo en espa{\~n}ol},  
  author={Rodr{\i}guez, Alexandre Veiga},  
  journal={Verba: Anuario galego de filoloxia},  
  number={12},  
  pages={253--286},  
  year={1985},  
  publisher={Servicio de Publicaciones}  
}
```

- **@tipo**

- Define o tipo de bibliografía. Se é artigo, libro, *site*, dentre outros

# Arquivo .bib (Tipos)

- Existem diversos tipos para uma entrada de bibliografia
- Formatos aceitos:
  - @article ,@book ,@collectedbook ,@conference ,@electronic ,@ieeetranbstctl ,@inbook ,@incollectedbook ,@incollection ,@injournal ,@inproceedings ,@manual ,@mastersthesis ,@misc ,@patent ,@periodical ,@phdthesis ,@preamble ,@proceedings ,@standard ,@string ,@techreport e @unpublished

```
@article{rodriguez1985consideraciones,  
  title={Consideraciones relativas a la actuación y límites de las oposiciones fonológicas interrumpido/continuo y tenso/flojo en español},  
  author={Rodríguez, Alexandre Veiga},  
  journal={Verba: Anuario galego de filoloxia},  
  number={12},  
  pages={253--286},  
  year={1985},  
  publisher={Servicio de Publicaciones}  
}
```

## ● rótulo da referência

- Define o rótulo da bibliografia
- Importante pois usaremos ela para fazer a referência no texto
- Neste caso, o nome é rodriguez1985consideraciones

```
@article{rodriguez1985consideraciones,  
  title={Consideraciones relativas a la actuación y lí  
    mites de las oposiciones fonológicas interrupto/  
    continuo y tenso/flojo en español},  
  author={Rodríguez, Alexandre Veiga},  
  journal={Verba: Anuario galego de filoloxia},  
  number={12},  
  pages={253--286},  
  year={1985},  
  publisher={Servicio de Publicaciones}  
}
```

- **campo da referênci**a={valor}

- Estabelece um campo para a referência, o qual representa alguma informação
  - Título, ano de publicação, revista, ...

# Arquivo .bib (Campos)

- Existem vários campos para uma entrada bibliográfica
- Campos aceitos:
  - address, annote, author, booktitle, chapter, crossref, edition, editor, howpublished, institution, journal, key, month, note, number, organization, pages, publisher, school, series, title, type, volume, year
- Dependendo do formato de bibliografia utilizado, cada **tipo** necessita de determinados **campos**
  - O formato fornecido por abnt.bst insere alguns indicadores para caso de falta de informação

# Ainda sobre o arquivo .bib

- Caracteres especiais no arquivo .bib
  - ã, â, á, à, ç, ...
- Salvar o arquivo .bib **no mesmo formato de codificação que o .tex**
  - UTF-8 sendo preferível



# Ainda sobre o arquivo .bib

- Precisamos construir toda a entrada formatada para todas as referências?
  - Não necessariamente
- Grande parte das bases de dados de livros, artigos, dentre outros, **já fornecem a bibliografia em formato Bibtex**
  - Google Scholar
  - IEEE
  - ACM
- Diminui os esforços para a bibliografia

# Citação bibliográfica

- Tendo uma entrada no arquivo .bib, podemos fazer a citação da mesma
- Voltaremos ao nosso arquivo .tex

Como dito em `\cite{rodriguez1985consideraciones}`,  
existem alguns fatores (...)

- `\cite{rótulo da citação}`
  - Faz a citação de uma bibliografia, utilizando um rótulo já definido
    - O Latex já se encarrega de referenciar corretamente
    - Se for o caso, a numeração também fica a cargo do Latex

# Particularidades

- Adendo à codificação de caracteres
- Fontes TrueType
- Modularização

# Um adendo à codificação de caracteres

- Em alguns casos específicos, não será possível inserir caracteres especiais
  - Quando editamos documentos com classes que fornecem diferentes codificações
- É possível fazer a inserção de caracteres especiais **mesmo** em diferentes codificações
- Existe comandos para isto
  - Em algumas referências prontas do Bibtex é possível verificar tais comandos

# Um adendo à codificação de caracteres

```
@article{rodriguez1985consideraciones},  
  title={Consideraciones relativas a la actuaci\u{a}n y l  
        \'{i}mites de las oposiciones fonol\u{o}gicas  
        interrumpido/continuo y tenso/flojo en espa\~{n}ol},  
  (...)  
}
```

## ● \adicional{caractere}

- Permite a inserção de um caractere especial
- Em **caractere**, descrevemos qual o caractere que levará um "adendo"
- Em **adicional**, inserimos o que vai no caractere
  - \'{o}: Estamos colocando ' (que acento agudo) no caractere o
  - O resultado é o caractere especial ó

# Um adendo à codificação de caracteres

Tabela 1: Comandos para inserção de caracteres especiais utilizados em português

Comando	Saída	Descrição
<code>\`{o}</code>	ò	Acento grave
<code>\'{o}</code>	ó	Acento agudo
<code>\^{o}</code>	ô	Circunflexo
<code>\~{o}</code>	õ	Til
<code>\c{c}</code>	ç	Cedilha

- Lembrete: Embora os exemplos acima na maioria das vezes utilizam `o` como exemplo, basta trocar o caractere para descrever outros caracteres especiais
  - `\'{a}`: á

# Fontes TrueType

- Ao longo destes *slides* foram vistos diversas vezes o uso de fontes TrueType
  - Fontes as quais todos os caracteres ocupam o mesmo tamanho!
  - Fonte ideal para a mostra de **código**
- Simples: `\texttt{texto em truetype}`



# Modularização

- Se assim desejarmos, podemos dividir nosso projeto em Latex em vários arquivos
  - Interessante quando temos um projeto grande
  - Duas formas

# input vs. include

- `\input{nome do arquivo}`
  - Inclui um arquivo externo do tipo .tex
  - Funcionamento: Tudo que está dentro do arquivo externo é passado para onde houve a chamada do comando `\input`
- `\include{nome do arquivo}`
  - Inclui um arquivo externo do tipo .tex, porém em uma **nova página**
  - Funcionamento: É feita uma quebra de página, e então o conteúdo do arquivo externo é passado para onde houve a chamada do comando `\include`

# Outras Possibilidades

- Existem **muitas** possibilidades em Latex
  - Comunidade do Latex bem ativa
  - Desenvolvimento de diversos pacotes para múltiplos fins

- Mudanças na **classe de documentos**
- Alguns comandos adicionais
- De restante, **grande parte** dos comandos em Latex são aplicáveis
- Tal qual o *template* do TSI para TCC, existe um formato voltado para apresentações
  - Também disponível no github
  - `https://github.com/gdotorres/apresentacao-tsi-pelotas`

# Ambiente matemático

- Permite uma ampla variedade de comandos os quais fornecem meios para formatação de conceitos matemáticos
  - Equações, fórmulas, ...
- Utiliza uma série de caracteres especiais, os quais são definidos em uma extensa lista de comandos<sup>13</sup>
- Possível construir equação em um ambiente online, permitindo mais rapidez e geração automática do código
  - Online LaTeX Equation Editor:  
<https://www.codecogs.com/eqnedit.php>

---

<sup>13</sup>[https://en.wikibooks.org/wiki/LaTeX/Mathematics#List\\_of\\_mathematical\\_symbols](https://en.wikibooks.org/wiki/LaTeX/Mathematics#List_of_mathematical_symbols)

- Exemplos de uso do ambiente matemático

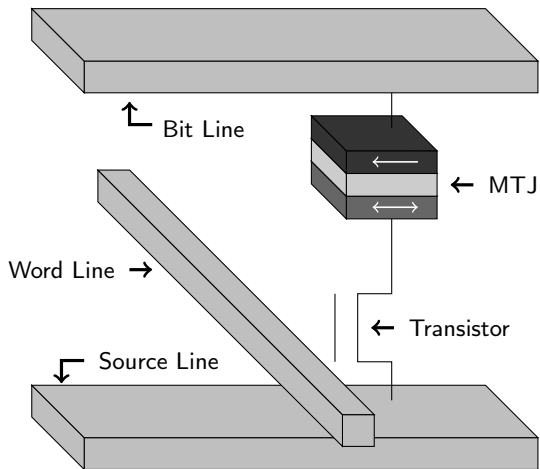
$$\frac{n!}{k!(n-k)!} = \binom{n}{k}$$

$$\int_0^{\infty} e^{-x} dx$$

$$f(n) = \begin{cases} n/2 & \text{if } n \text{ is even} \\ -(n+1)/2 & \text{if } n \text{ is odd} \end{cases}$$

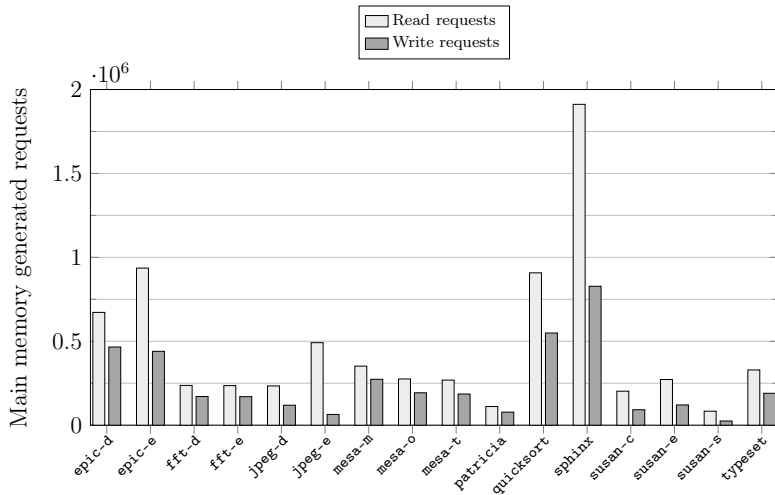
# Outras funcionalidades

- Desenho de figuras
  - Pacote TikZ



# Outras funcionalidades

- Gráficos
  - Pacote pgfplots





# Outras funcionalidades

- Códigos
  - Pacote listings

```
class Coord {  
private:  
    int x;  
    int y;  
  
public:  
    Coord(int x, int y) {  
        this.x = x;  
        this.y = y;  
    }  
  
    int getX() { return x; }  
    int getY() { return y; }  
}
```