A classe `ist-thesis`

Classe LaTeX de tese não oficial do Instituto Superior Técnico

Daniel de Schiffart

com apoio de

João Gonçalves

João Lourenço

https://github.com/ekspek/ist-thesis

Resumo

Esta classe tem como objetivo preparar e formatar qualquer documento LaTeX numa tese de mestrado de acordo com os regulamentos do Instituto Superior Técnico.

Índice

Índice **2**

1 *Quick Start* – Os Básicos **5**

2 Argumentos Opcionais **6**

3 Comandos e Ambientes **7**

3.1 Capa e Dados de Tese 7

3.1.1 Exemplo de Utilização 8

3.2 Secções Especiais 8

3.2.1 Exemplo de Utilização 9

3.3 Comandos Internos 10

3.4 Lista Completa de Comandos e Ambientes 10

4 Implementação da Classe **12**

4.1 Estilo de Página 12

4.2 Cores e Hiperligações 12

4.3 Tipos de Letra 13

4.3.1 Tipo de Letra Principal 13

4.3.2 *Monoespaço* 13

4.3.3 Tipografia Matemática 13

4.3.4 Tipos de Letra *Serif* 14

4.3.5 Demonstrações 14

4.4 *Packages* Utilizadas 15

Introdução

Mal de mim se não deixasse umas palavras minhas aqui mesmo.

Eu aprendi a utilizar LaTeX à dois anos, no início quarto ano, e como toda a gente deve esperar, foi a escrever relatórios para as cadeiras do Técnico. Depois de ter sido feito refém de algumas horas a escrever relatórios de Telecomunicações e Projeto de Sistemas Digitais, peguei-lhe o gosto e tenho aprendido a pouco e pouco os limites do que se pode fazer com LaTeX (que são muitos, *spoiler alert*). Vendo os semestres a passar e a preparar-me para ter o Técnico nas costas, queria deixar qualquer coisa atrás escrita em LaTeX para a malta mais nova de aeroespacial. Uma sebenta, uns apontamentos, umas resoluções, enfim, qualquer coisa. O tempo passou e com a preguiça lá ia adiando qualquer motivação que tivesse para isso, especialmente considerando que a última coisa que alguém quer ouvir falar nas férias é de cadeiras acabadas. Portanto peguei na única coisa que fazia sentido nesta reta final: gastar tempo com o modelo da tese meses antes sequer de começar.

Este projeto começou no final do ano passado como uma brincadeira. Estava a começar a apanhar o ritmo do LaTeX e começava a reparar o quanto reutilizava certas *packages*, certas definições, enfim, sempre o mesmo código em cada relatório que fazia. Considerando que as classes de LaTeX definiam já um molhe de coisas quando são carregadas, comecei a ter isso em conta, olhando para as classes de LaTeX online, nem pensava que fosse capaz de escrever uma de raíz. Mas o aspeto mente; é pura e simplesmente código LaTeX, e embora fossem precisos uns trambolhões, lá apareceu o código a pouco e pouco.

Muito deste trabalho transpõe o que aprendi a escrever a classe `ist-relatorio`, um projeto mais experimental e *kaleidoscpópico* que mesmo assim ainda se prova útil quando preciso dele. Embora menos versátil, esta classe aqui ainda colocou vários desafios na interface entre o utilizador e o *output*, mas acho que em geral o produto final está com bom aspeto, e ainda preservando alguma versatilidade. Aliás, para provar essa versatilidade, decidi implementar esta mesma documentação utilizando a classe, só mesmo por curiosidade, para ver se resultava. Eu sei que o abuso do *Arial* em todas as páginas queima os olhos a ver, mas o Técnico é que manda `¯\_('_')_/¯`

A inspiração principal de passar para a tese foi o meu prazo para tal a reduzir mais e mais, mas também observar os inúmeros modelos de tese que estão espalhados pela net, para as mais variadas universidades à volta do mundo. Não fazia sentido ver o Técnico ficar para trás, especialmente considerando a *imensa imensa* progressividade tecnológica deste nosso cantinho da Universidade de Lisboa. Menções especiais para o modelo de tese da TUDelft, criado à imagem do *Huisstijl* deles, que li e reli para aprender a criar classes.

E como é óbvio, também muita inspiração foi retirada do modelo de tese do Professor André Marta, que ainda detém o *de facto* modelo de LaTeX do Técnico. Admiro a disponibilidade dele para partilhar o código dele sem restrições e o quão bem montado está, tanto que pensei em largar a ideia completamente e utilizar o código dele. Eu não quero de modo algum competir com o projeto do professor Marta, apenas considero duas abordagens diferentes ao mesmo problema. E promovo que olhem para o projeto e escolham, vejam na página do Fénix do professor.

Não tenho muito mais a acrescentar. Ficam aqui as palavras típicas, revejam o produto final, contactem-me se necessário, partilhem o projeto, contribuam se puderem, divulguem se gostarem do que está aqui feito ou deixem sugestões para o que podia ser melhor.

Muito boa sorte com a vossa tese!

Daniel Lopes de Schiffart
Aluno 81479
Fevereiro de 2019

Capítulo 1

Quick Start – Os Básicos

Para carregar a classe e criar um documento com o aspeto de tese, é necessário fazer o download do ficheiro da classe, `ist-thesis.cls`. Após isso, num ficheiro `.tex` separado, a primeira linha do documento terá de ser

```
\documentclass{ist-thesis}
```

Isto carrega a classe. Introduza qualquer texto e compile para testar. Um exemplo de um documento `.tex` o mais pequeno possível válido que compila sem problemas encontra-se abaixo.

```
1 \documentclass{ist-thesis}
2
3 \begin{document}
4
5 Hello World!
6
7 \end{document}
```

Mais propriedades que a classe tem encontram-se nos capítulos que se seguem.

Capítulo 2

Argumentos Opcionais

Ao carregar a classe, no comando **\documentclass**, é possível passar argumentos opcionais para definir opções globais da classe. Os argumentos opcionais estão listados aqui.

portuguese Define a língua portuguesa para o documento, principalmente no idioma de macros e expressões como índice e resumo. Esta opção está seleccionada por defeito e não precisa de ser passada.

english Define a língua inglesa para o documento, análogo à opção `portuguese`.

bw Ativa a opção preto e branco. Troca todas as instâncias de ciano no documento, como nas ligações e URLs, por preto.

origmath Retorna o tipo de letra matemático ao carregado originalmente pelo LaTeX. Segue da discussão da secção 4.3.3 e evita o carregamento da *package* `newtxsf`.

Para ativar um ou mais argumentos, insira no início do ficheiro entre parênteses retos os argumentos a utilizar.

```
\documentclass[english]{ist-thesis}
```

Para carregar mais que um argumento, basta separá-los por vírgulas.

```
\documentclass[english,origmath]{ist-thesis}
```

Capítulo 3

Comandos e Ambientes

Este capítulo contém a documentação oficial dos comandos e ambientes implementados na classe utilizáveis pelo utilizador e as suas funções e dependências dentro do documento que as utiliza. Também incluídas estão algumas modificações feitas a documentos base de LaTeX para funcionarem corretamente com as funções implementadas.

Estes comandos podem ser separados em três categorias, relevantes para a capa e informação do documento, na secção 3.1, para a estrutura e secções especiais, na secção 3.2, e comandos internos definidos e utilizados pela classe, na secção 3.3.

3.1 Capa e Dados de Tese

A classe implementa um comando que permite criar a capa automaticamente, pedindo informação ao utilizador sobre os dados do documento e colocando-os no sítio certo. Para criar uma capa, basta utilizar o comando

```
\makecover
```

Como o LaTeX corre o código de cima para baixo, recomenda-se que o utilizador invoque este comando no início do documento, logo a seguir ao comando `\begin{document}`.

```
\begin{document}

\makecover
```

Este comando por si só devolve uma capa com os dados preenchidos com valores pré-definidos. Para alterar estes valores, a classe fornece um conjunto de comandos, um para cada entrada da capa.

`\settitle{<`*título*`>}` Define o título.

`\setsubtitle{<`*subtítulo*`>}` Define o subtítulo.

`\setauthor{<`*autor(es)*`>}` Define o(s) autor(es). Para definir mais que um autor, a classe fornece um comando para separar autores, o comando `\tand`. Consulte a secção 3.1.1 para exemplos de uso.

\setdegree{<*curso*>} Define o nome do curso.

\setinstitution{<*instituição*>} Define a instituição. Comando residual, não é utilizado na capa.

\setdate{<*data*>} Define a data.

\setsupervisor{<*orientador(es)*>} Define o(s) orientador(es). Para definir mais que um orientador, consulte o comando para definir autores.

\setcsupervisor{<*orientador(es)*>} Igual ao comando anterior, mas define o(s) orientador(es) vogais. Permite distinguir os dois grupos caso estes sejam diferentes.

\setchairperson{<*presidente*>} Define o presidente.

\setcommittee{<*vogal(is)*>} Define o(s) vogal(is). Para definir mais que um vogal, consulte o comando para definir autores.

\setcoverimage[<*largura relativa*>]{<*imagem*>} Define a imagem de capa. Neste caso, é necessário dar o nome da imagem que se deve encontrar no mesmo diretório que o documento. A sintaxe é igual ao comando **\includegraphics**, excepto que só aceita um argumento, a largura da imagem em relação à largura de texto (portanto um valor de 1 devolveria uma imagem que cobre a capa de margem a margem).

3.1.1 Exemplo de Utilização

Para exemplificar o uso dos comandos da secção anterior, está aqui incluído um exemplo de sintaxe.

```
\settitle{Tese do Schiffart}
\setsubtitle{Sobre um tema}
\setauthor{Daniel de Schiffart}
\setdegree{Mestrado Integrado em Engenharia Aeroespacial}
\setsupervisor{Prof. Supervisor}
\setcsupervisor{Prof. Supervisor Vogal}
\setchairperson{Presidente}
\setcommittee{Prof. A. \tand Prof. B.}
\setcoverimage[0.5]{example-image-a}
```

3.2 Secções Especiais

Na escrita de uma tese existem certas secções que são escritas por grande parte dos autores. Algumas das secções descritas no guia da Direcção Académica estão por isso implementadas na classe utilizando ambientes[1] LaTeX especiais para lhes atribuir formatação especial e facilitar a escrita ao utilizador. São elas três, a dedicatória, os agradecimentos, e o resumo.

Os ambientes estão defindos em LaTeX como listado a seguir.

[1] Ambientes em LaTeX são macros parecidas com comandos, mas em vez de serem chamados uma vez e o seu conteúdo escrito dentro dos argumentos, os ambientes têm princípio e fim delimitados pelo comando `\begin` e `\end`. Consulte os exemplos da secção 3.2.1.

dedication Dedicatória.

acknowledgements Agradecimentos.

tabstract Resumo. Este ambiente introduz palavras-chave no final da página, que podem ser escritas via um argumento opcional do ambiente. Consulte a secção 3.2.1 para a sintaxe de utilização.

fabstract Resumo na língua alternativa. Este ambiente é idêntico ao `tabstract`, mas deteta a língua utilizada no resto do documento e utiliza a língua contrária (inglês no caso do português, português no caso do inglês).

O texto escrito dentro destes ambientes é colocado numa página em separado para cada um, e aparecem no documento na ordem em que são escritos no texto.

3.2.1 Exemplo de Utilização

Exemplo de utilização dos ambientes fornecidos. Note-se que no ambiente do resumo (`tabstract`) é entregue um argumento extra, dentro de chavetas, que será impresso no final da página.

```
\begin{dedication}
        Para dedicar a quem quiser.
\end{dedication}

\begin{acknowledgements}
        Did you ever hear the tragedy of Darth Plagueis The Wise? I thought not. It's not
        ↪ a story the Jedi would tell you. It's a Sith legend. Darth Plagueis was a
        ↪ Dark Lord of the Sith, so powerful and so wise he could use the Force to
        ↪ influence the midichlorians to create life... He had such a knowledge of the
        ↪ dark side that he could even keep the ones he cared about from dying.
\end{acknowledgements}


\begin{tabstract}{Resumo, Palavras-Chave, Resumo Analítico}
        Resumo e palavras-chave (em português e em inglês). O resumo analítico, também
        ↪ designado por resumo ou abstract, descreve o objectivo, o conteúdo do
        ↪ trabalho e as conclusões. Deve ser escrito em português e inglês, com um
        ↪ máximo de 250 palavras cada e acompanhado de 4 a 6 palavras-chave.
\end{tabstract}


\begin{fabstract}{Abstract}
        The abstract describes the objective, the content of the project and the
        ↪ conclusions. It must be written in both portuguese and english, with a
        ↪ maximum of 250 words, accompanied by 4 to 6 keywords.
\end{fabstract}
```

3.3 Comandos Internos

Estes são utilizados para criar na capa e retornam os valores definidos pelo utilizador nos comandos da secção 3.1. Estes são apenas utilizados internamente pela classe, mas podem ser utilizados pelo utilizador ao longo do documento para colocar o seu valor algures no texto.

```
\ttitle{}
\tsubtitle{}
\tauthor{}
\tsupervisor{}
\tcsupervisor{}
\tdegree{}
\tchairperson{}
\tcommittee{}
\tinstitution{}
\tdate{}
\tcoverimage{}
\tcoverimagewidth{}
```

3.4 Lista Completa de Comandos e Ambientes

Nesta secção estão listados todos os comandos disponíveis ao utilizador ao usar a classe.

Primeiramente temos os comandos para criar a capa.

```
\settitle{}
\setsubtitle{}
\setauthor{}
\setdegree{}
\setinstitution{}
\setdate{}
\setsupervisor{}
\setcsupervisor{}
\setchairperson{}
\setcommittee{}
\setcoverimage[]{}

\tand

\makecover
```

Seguidos estão os ambientes disponíveis.

```
\begin{dedication}
\end{dedication}

\begin{acknowledgements}
\end{acknowledgements}

\begin{tabstract}{}
\end{tabstract}

\begin{fabstract}{}
\end{fabstract}
```

Por fim, os comandos internos.

```
\ttitle{}
\tsubtitle{}
\tauthor{}
\tsupervisor{}
\tdegree{}
\tchairperson{}
\tcommittee{}
\tinstitution{}
\tdate{}
\tcoverimage{}
\tcoverimagewidth{}
```

Capítulo 4

Implementação da Classe

Este capítulo é dedicado à descrição de como a classe e todas as suas funções foram implementadas dentro do contexto de TEX e LATEX, com especial atenção às *packages* utilizadas. Tem por isso um cariz mais técnico e não é necessária a sua leitura para qualquer utilização da classe.

4.1 Estilo de Página

As definições da página estão reduzidas aos tamanhos de margem. Estes estão estabelecidos em 2.5 centímetros para todos os lados, e são definidos utilizando a *package* `geometry`.

Também definido está o espacamento entre linhas. Para este efeito é utilizada a *package* `setspace`, que evita definir o espaçamento de linhas para **todo** o texto, definindo apenas o espaçamento para o corpo do texto, evitando definir por exemplo legendas de imagens ou tabelas. Permite também criar exceções a meio do texto via o comando `\setspace`, mas sugere-se o seu uso apenas em situações especiais tendo em atenção o regulamento da Direcção Académica.

4.2 Cores e Hiperligações

As hiperligações feitas na tese de exemplo e nesta documentação são feitas utilizando a *package* `hyperref`, e estão definidas para utilizar texto colorido (ao invés de links rectangulares a rodear o texto, que é o comportamento por defeito). Esta definição pode ser mudada utilizando os comandos descritos na documentação oficial da *package*.

As cores utilizadas nestas hiperligações (e não só) foram definidas utilizando a *package* `xcolor`. A cor mais utilizada é o ciano do Instituto Superior Técnico, dado pelo Manual de Identidade, e definida como $1, 0, 0, 0$ na escala de CMYK. A segunda definida é um cinzento de $0.2, 0, 0, 0.8$ na escala de CMYK, dada no mesmo documento referido.

Esta cor está definida na classe como `ist-cyan`. Estas cores estão definidas na classe como `ist-cyan` e `ist-gray` e podem ser reutilizadas de acordo com a sintaxe da *package* `xcolor`.

4.3 Tipos de Letra

O tópico de tipos de letra tem vários pormenores a descrever, mas a escolha de cada um deles prende-se à decisão da Direcção Académica em escolher *Arial* como o tipo de letra principal do documento. A discussão de cada tipo de letra implementado inclui a justificação relevante, mas é importante referir que para os tipos de letra não principais que serão descritos podem ser substituídos de acordo com a preferência do utilizador da forma que entender. Em geral, as implementações de tipos de letra num documento feito sem e com a classe `ist-thesis` são idênticas e devem funcionar como suposto.

4.3.1 Tipo de Letra Principal

Em contraste com o tipo de letra de raíz de qualquer documento LaTeX[1], *Computer Modern*, o tipo de letra especificado pelo guia de preparação da dissertação, *Arial*, é muito distinto, mas igualmente (ou mais) reconhecível. No entanto, um dos problemas é que este tipo de letra é muito mais omnipresente em sistemas com *Windows* do que no restantes, como também na internet. É, por isso, um tipo de letra que não encontrou implementação nativa em LaTeX.

No entanto, *Arial* é baseado no tipo de letra *Helvetica*, que é facilmente disponível online em vários formatos e codificações. Os dois tipos de letra são praticamente indistinguíveis, sendo o design de *Arial* baseado fortemente em *Helvetica* na altura da sua criação. Esta segunda opção, pela sua facilidade de obtenção, já encontrou várias implementações em LaTeX.

Para o tipo de letra base deste projeto escolhemos o tipo de letra TeX Gyre Heros, que está incluído em qualquer distribuição de LaTeX. A razão desta escolha foi a sua facilidade de implementação e relativa solidez em várias misturas de propriedades, desde **negrito** a *itálico* à ***mistura dos dois***, como também a caracteres estrangeiros, nomeadamente caracteres utilizados na língua portuguesa. A grande lacuna deste tipo de letra é a falta de tipografia matemática, uma lacuna que será discutida na secção 4.3.3.

4.3.2 *Monoespaço*

O tipo de letra monoespaçado escolhido para o projeto foi `Inconsolata Narrow`, pela aparência relativamente semelhante ao tipo de letra principal, mas principalmente por preferência pessoal. Embora este tipo de letra não seja extensivamente utilizado na maioria das teses, é fulcral na introdução de código de programação a meio do texto onde caracteres monoespaçados mantêm coerência entre cada linha.

4.3.3 Tipografia Matemática

Como anteriormente referido na secção 4.3.1, o tipo de letra utilizado para matemática não está incluído no tipo de letra principal utilizado e teve de ser seleccionado à parte.

Para utilizadores com experiência em LaTeX, o tipo de letra de raíz utilizado pelo LaTeX, *Computer Modern*, já inclui todas as funções necessárias para as mais variadas necessidades de escrita de textos matemáticos. E embora isto seja verdade, a colisão entre *Arial* (ou TeX Gyre Heros, neste caso) e

[1] Neste caso referimo-nos a documentos utilizando as classes base de LaTeX, como `article`, `report`, ou book, entre outras.

Computer Modern não é a mais agradável, o que levou à procura de uma alternativa mais satisfatória. Não obstante, para quem não se quiser preocupar com pormenores ou estiver satisfeito com o tipo de letra de matemática original, é incluída a opção `origmath` na classe para manter este tipo de letra.

Neste projeto foi utilizada a *package* `newtxsf`, que implementa um tipo de letra mais semelhante ao tipo de letra utilizado no resto do documento.

4.3.4 Tipos de Letra *Serif*

Para os leitores mais perspicazes, podem ter reparado que ainda sobra uma família de tipos de letra por definir, os tipos de letra *serif*.

Para os leitores curiosos, passo a explicar. O LaTeX implementa em cada documento duas famílias de tipos de letra principais, *serif* e *sans-serif*. Estes termos são utilizados geralmente (mesmo fora de qualquer contexto de LaTeX) para descrever tipos de letra. O primeiro corresponde a tipos de letra mais pormenorizados, angulares e pontiagudos, sendo mais adequados a ler em papel. O segundo é utilizado para descrever tipos de letra mais simples e redondos, muito mais fáceis de ler em ecrãs ou outros suportes digitais onde o detalhe sacrifica alguma legibilidade. Para os leitores mais interessados deixo um link que pode esclarecer o tópico.

Por defeito, o LaTeX utiliza tipos de letra *serif*, como o já referido *Computer Modern*, por defeito, deixando ao utilizador a opção de definir e utilizar a sua contrapartida *sans-serif* em qualquer local no texto. No entanto, para o projeto em questão, de acordo com os regulamentos do Instituto Superior Técnico, utilizamos um tipo de letra *sans-serif* como base, o que implica que está disponível a utilização de um tipo de letra *serif* quando necessário.

Como este tópico é de relativamente baixa relevância, deixámos este tipo de letra sem modificações, o que implica que quando é chamado, como neste excerto, obtemos o tipo de letra de raíz do LaTeX.

4.3.5 Demonstrações

Os tipos de letra referidos nas secções anteriores estão demonstrados nesta secção. Por ordem, os tipos de letra normal (*sans-serif*), itálico, maiúsculas pequenas, *serif*, e matemático, todas acompanhadas pelo seu equivalente a negrito.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

`Lorem ipsum dolor sit amet, consectetuer adipiscing elit.`

`Lorem ipsum dolor sit amet, consectetuer adipiscing elit.`

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

Loremipsumdolorsitamet, consectetueradipiscingelit.

Loremipsumdolorsitamet, consectetueradipiscingelit.

Também incluído aqui está uma demonstração de uma equação no tipo de letra newtxsf referido na secção 4.3.3.

$$\mathbb{P}\left(\frac{X_1 + \cdots + X_n}{\sqrt{n}} \leq y\right) \to \mathrm{R}(y) = \int_{-\infty}^{y} \frac{e^{-t^2/2}}{\sqrt{2\pi}}\,dt \qquad \text{as} \quad n \to \infty$$

4.4 *Packages* Utilizadas

Fica na tabela 4.1 um resumo de todas as *packages* utilizadas na implementação da classe.

Tabela 4.1: *Packages* carregadas pela classe sem argumentos opcionais, dependendo do compilador utilizado.

pdfLaTeX	XeLaTeX ou LuaLaTeX
etoolbox	
ifluatex	
ifxetex	
ifpdf	
mathtools	
graphicx	
xcolor	
hyperref	
geometry	
inputenc	
fontenc	
	fontspec
babel	
tgheros	
inconsolata	
newtxsf	
microtype	
tocbibind	
setspace	
totpages	