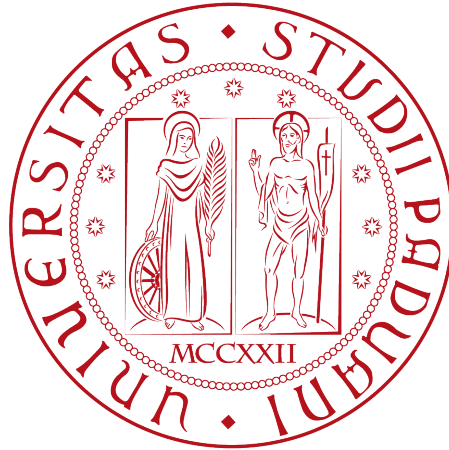


Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA

CORSO DI LAUREA MAGISTRALE IN INFORMATICA



Guida Galattica per \LaTeX

v0.2.0

Nuove release: <https://github.com/R-and-LaTeX/GuidaGalatticaPerLaTeX/releases>

Le [contribuzioni](#) e le [segnalazioni di problemi](#) son ben accette.

Davide Polonio, Marco Zanella: *Guida Galattica per \LaTeX* , © CC-BY-SA-4.0, 2018.

Indice

Non fatevi prendere dal panico!	1
1 Introduzione	3
1.1 Si mangia?	3
1.2 Se usassimo Word?	3
1.3 Da dove abbiamo copiato	4
2 Ambiente di Lavoro	5
2.1 Windows e TeXnicCenter	5
2.1.1 Creare un progetto	6
2.2 Compilare su Linux/MacOS	11
2.3 Suddivisione del progetto in più file	11
3 Primi passi	13
3.1 Come è diviso un documento	13
3.2 Tipi di documento	14
3.2.1 Esempi	14
3.3 Sezionare un documento	15
3.4 Package	16
4 Comandi Base	19
4.1 Liste	19
4.1.1 Esempi	19
4.1.2 Personalizzare gli elenchi puntati	20
4.2 Commenti	21
4.3 Formattazione di base	21
4.4 A spasso nella spaziatura	21
4.4.1 A capo riga	21
4.4.2 Inserire pagine	22
4.4.3 Vuoti cosmici nel testo	22
4.4.4 Riempire i vuoti cosmici	22
4.4.5 Orientare le pagine	23
4.4.6 Ma io voglio tutto il documento orizzontale!	24
4.5 Tabelle	25
4.5.1 Tabelle semplici	25
4.5.2 Tabelle su più pagine	27
4.5.3 Altre informazioni sulle tabelle	30

4.6	Immagini	32
4.6.1	Affiancare le immagini	33
4.6.2	Affiancare testo e immagini	35
4.7	Note a piè di pagina	37
4.8	Collegamenti ipertestuali	37
4.8.1	Collegamenti nel testo	37
4.8.2	Collegamenti al Web	38
5	Formule matematiche	39
5.1	La modalità matematica	39
5.2	Apici e pedici	39
5.3	Ambienti utili	40
5.3.1	Equazioni	40
5.3.2	Definizione per casi	40
6	Bibliografia	43
6.1	Bibliografia manuale	43
6.1.1	Inserire la bibliografia nel documento	43
6.2	Bibliografia automatica	45
6.2.1	Database delle opere	45
6.2.2	Stile di citazione	46
6.2.3	Inserire la bibliografia nel documento	47
7	Creare comandi personalizzati	49
8	Creare presentazioni: Beamer	51
8.1	I comandi di base	51
8.1.1	Creare e dare titoli a frame	51
8.1.2	Creazione slide iniziale	52
8.2	Temi	54
8.2.1	Temi predefiniti	54
8.3	Transizioni ed effetti	55
8.3.1	Transizioni per gli elenchi puntati	55
8.3.2	Effetti	56
9	La Tesi	59
9.1	Template	59
9.2	Copertina	59
9.3	Seconda pagina, ringraziamenti	62
9.4	Si può fare di più?	64

Elenco delle figure

2.1	Schermata principale di TeXnicCenter	6
2.2	Zoom della schermata principale di TeXnicCenter	6
2.3	Finestra per l'inserimento di una nuova immagine nel documento . .	10
2.4	Il risultato che dovremmo ottenere	11
4.1	Didascalia dell'immagine	33
4.2	Didascalia globale delle 4 immagini	34
4.3	Didascalia dell'immagine	36
8.1	Presentazione L ^A T _E X personalizzata	52
9.1	Frontespizio tesi	62

Elenco delle tabelle

1.1	Confronto tra L ^A T _E X e Word	3
4.1	Una tabella d'esempio	26
4.2	Una tabella d'esempio con i bordi	27
4.3	Una tabella multipagina d'esempio	29
4.4	Elenco delle opzioni che si possono mettere nella voce <code>tabular</code> . . .	31
4.5	Una tabella con la larghezza delle colonne personalizzata	31
8.1	Tabella degli effetti Beamer	57

Non fatevi prendere dal panico!

Questo libricino serve da piccola guida per tutti coloro che si vogliono avventurare nel mondo di \LaTeX . Verrà presentata solo una piccola parte del grande ecosistema \LaTeX , con lo scopo di essere un manuale di rapida consultazione rispetto ad una guida completa.

Per tutti coloro che stanno seguendo il corso, sarà anche utile per riprendere argomenti persi o per completare i propri appunti.

Il manuale introdurrà inizialmente cos'è \LaTeX , verrà poi preso in considerazione un IDE¹ apposito che sarà impiegato su Windows, per poi parlare dei vari comandi del linguaggio, la formattazione del testo e l'impaginazione di immagini o tabelle. Si concluderà con un breve accenno ad argomenti più avanzati, come la creazione di slide tramite *Beamer* e la scrittura di pacchetti e stili personalizzati.

Per concludere è raccomandato avere sempre a portata di mano un asciugamano in caso d'emergenza.

¹Che sta per **Integrated Development Environment**, ovvero un programma in cui schiacciando un bottone vi genera il PDF di quello che state facendo.

Introduzione

1.1 Si mangia?

No, non ancora. \LaTeX (pronunciato come /'latek/ e non /'lateks/) fu ideato all'incirca una trentina d'anni fa, nel 1985, da Leslie Lamport (da cui L sta per il suo cognome) ed è un linguaggio di programmazione¹ che permette la stesura di documenti al fine di generare PDF e altri formati leggibili da umani.

\LaTeX non ha un approccio WYSIWYG², ma il codice viene scritto in un *markup* che poi viene compilato. È possibile tuttavia aiutarsi con degli IDE per avere aiuti durante la stesura del testo.

1.2 Se usassimo Word?

È una domanda che sorge spontanea, e per documenti di piccole dimensioni è un'alternativa preferibile. Per documenti grandi come per esempio una tesi o un articolo scientifico invece \LaTeX vince senza dubbi: permette una facile gestione del documento, e permette di gestire cose come la notazione matematica e il versionamento in maniera facile.

Uso	\LaTeX	Word
Piccoli progetti	😊😊	😊😊😊
Grandi progetti	😊😊😊	😊
Facilità d'uso	😊😊	😊😊😊
Curva d'apprendimento	😊	😊😊😊
Qualità layout	😊😊😊	😊😊
Scrittura scientifica	😊😊😊	😊
Totale	14	13

Tabella 1.1: Confronto tra \LaTeX e Word

¹Per maggiori informazioni: <https://stackoverflow.com/a/2968527>.

²Acronimo per What You See Is What You Get, che era evidentemente troppo lungo per essere scritto ogni volta.

1.3 Da dove abbiamo copiato

Questa piccola guida non contiene tutto il sapere su \LaTeX , ma solo un sottoinsieme ristretto: maggiore conoscenza può essere acquisita da:

- [Wikibook su \$\text{\LaTeX}\$](#)
- [L'arte di scrivere in \$\text{\LaTeX}\$](#)

Queste fonti sono le stesse da dove gli autori si sono *ispirati*, rielaborando materiale anche di altri corsi³. Esistono siti come [Stackexchange](#) che possono essere utili per fare domande e ricevere risposte a dubbi che si possono avere durante la stesura di un documento. Infine, per chi non masticasse bene l'inglese, esiste anche il [Gruppo di Utilizzatori Italiani di \$\text{\TeX}\$](#) , che presenta un forum su cui è possibile esprimere dubbi e ottenere aiuto.

³Vedesi le lezioni [1](#) [2](#) e [3](#) a cura di Mirko Polato.

Ambiente di Lavoro

Per far funzionare \LaTeX sul vostro computer dovete scaricare un compilatore, che potete trovare a questo indirizzo <https://miktex.org/download>. L'installazione è facile ma il completamento potrebbe richiedere un po' di tempo. Siate pazienti. Per gli scopi di questo corso utilizzeremo **TeXnicCenter**, un comodo IDE che potete scaricare gratuitamente all'indirizzo <http://www.texniccenter.org/download/>. Ovviamente avrete bisogno anche di un visualizzatore PDF, in questo caso noi consigliamo un lettore PDF leggero e veloce (cosa che Adobe Reader non è), che ci sarà molto utile in quanto dovremo aprirlo ripetutamente: il lettore PDF in questione si chiama SumatraPDF ed è gratuitamente scaricabile al seguente indirizzo <https://www.sumatrapdfreader.org/download-free-pdf-viewer.html>. In ogni caso siete liberi di utilizzare il lettore PDF che più preferite.

2.1 Windows e TeXnicCenter

Durante la prefazione (che nessuno di voi avrà certamente letto) abbiamo parlato di IDE. Per il nostro corso ne utilizzeremo uno (come già detto sopra) e in questo capitolo vedremo un attimo alcune delle sue funzionalità, giusto per prendere mano con l'ambiente che dovremo usare.

Se avete installato TeXnicCenter sul vostro computer per la prima volta e lo state aprendo solo ora vi chiederà alcune informazioni riguardo al lettore DVI e PDF da usare. Riguardo al lettore DVI potete pure lasciare i campi vuoti, mentre per il lettore PDF premete i tre puntini accanto alla prima casella di testo, e selezionate nel vostro hard disk l'eseguibile del lettore PDF che più preferite.

Prima di tutto, alla apertura del programma ci troveremo ad una schermata vuota, in cui possiamo vedere diversi bottoni.

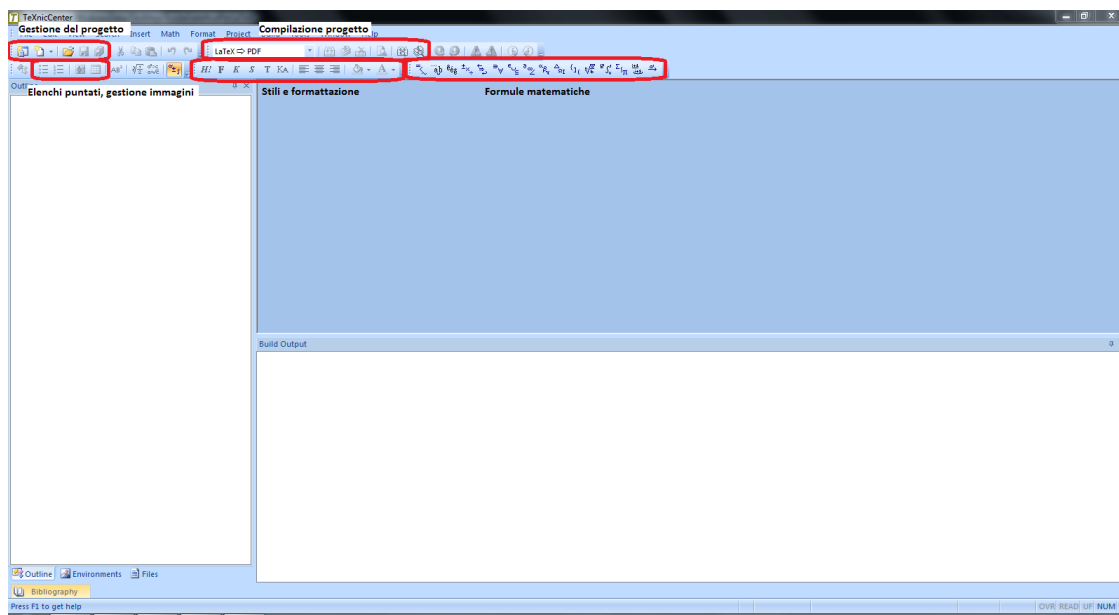


Figura 2.1: Schermata principale di TeXnicCenter

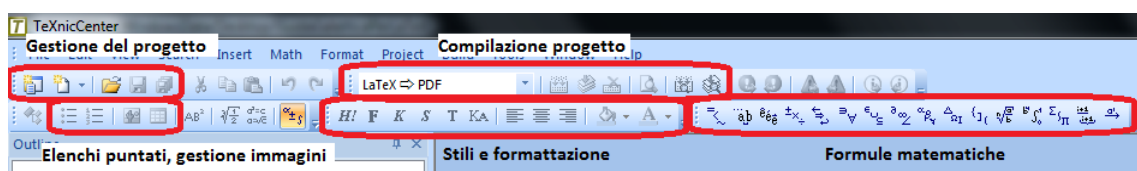


Figura 2.2: Zoom della schermata principale di TeXnicCenter

Come possiamo vedere dall'immagine¹ notiamo partendo da in alto a sinistra e scorrendo verso destra che inizialmente troviamo dei bottoni per gestire il progetto (o per crearne di nuovi). Abbiamo subito dopo le opzioni per la compilazione: il menù a tendina ci permette di selezionare che tipo di output vogliamo (se PDF o altri formati) e i bottoni accanto a questo menù ci permettono di lanciare la compilazione del progetto. Nella seconda riga troviamo le impostazioni per la creazione di elenchi puntati e numerati senza necessità di scrivere codice, seguiti dai bottoni per aggiungere immagini e tabelle. Nel penultimo riquadro ci sono i classici bottoni per impostare gli stili di formattazione (quelli che siamo abituati ad avere in Word per intenderci). Alla fine troviamo i bottoni per l'inserimento di formule matematiche, che tratteremo più avanti in questa guida.

2.1.1 Creare un progetto

Iniziamo ora la creazione di un progetto. Clicchiamo su **New Project** e selezioniamo il modello vuoto. Spuntiamo la casellina **Uses BibTeX** nella sezione **Features**, e diamo un nome al nostro progetto. Fatto ciò il bottone **Ok** dovrebbe essersi sbloccato, e lo clicchiamo. Il programma creerà le basi necessarie per cominciare un nuovo progetto!² Il progetto che avremo davanti è però vuoto, e consisterà di un unico file:

¹Sappiamo che si vede poco ma questo è il meglio che siamo riusciti a fare

²Nota, se volete potete selezionare il percorso che preferite dove salvare il progetto.

per progetti \LaTeX grandi è generalmente buona pratica separare quelle che sono le impostazioni del progetto (come configurazione dei pacchetti e importazione degli stessi) dal contenuto (ovvero quello che è il vero e proprio documento). Questo permette poi di eseguire modifiche più facilmente, e di avere un maggior ordine nel progetto stesso. La struttura consigliata potrebbe essere³:

```
<nomeprogetto>.tex
res/
-- config/
---- config.tex
---- package.tex
-- sections/
-- img/
listOfSections.tex
```

Spiegheremo brevemente perché:

- Il `<nomeprogetto>.tex` è buona pratica che contenga solo il codice necessario per importare i vari pezzi del progetto, e per darne le informazioni di base
- In `res/` ci saranno le risorse vere e proprie del progetto, che sono composte dai file di configurazione, dalle varie immagini e da altri file TeX, che possono rappresentare capitoli o parti (la suddivisione è a proprio piacimento)
 - Abbiamo il file `config/` dove configurare i vari pacchetti (per esempio scriviamo lì se preferiamo avere i link di colore rosso al posto di blu), mentre nell'altro file, `package.tex`, abbiamo tutte le importazioni necessarie per far compilare il documento. In questa maniera, se un pacchetto dovesse a detta del compilatore risultare mancante potremmo subito controllare se avrà ragione o meno⁴
 - In `img` metteremo tutte le immagini, così da avere un unico posto dove salvarle e eviteremo in questa maniera di spargerle su tutto il progetto.
 - `listOfSections` grazie alla *keyword* `\include{ }` possiamo includere diversi file, ed è consigliato includere qui la lista dei sezioni/parti che vogliamo vengano compilati nel documento.

Il funzionamento e l'inclusione dei file in \LaTeX è un argomento importante che verrà meglio spiegato in 2.3.

Ora che abbiamo un progetto vuoto possiamo praticamente fare quello che vogliamo! Siccome è la prima volta, proviamo a fare un progetto che ci stampi due pagine, uno con il titolo e l'altra con una scritta e una figura.

³Consigliata in quanto non ne esiste una unica, questa è quella che negli anni gli autori si sono trovati meglio ad utilizzare.

⁴E l'avrà perché i computer sono macchine, e le macchine se programmate bene hanno la tendenza e non sbagliare.

2.1.1.1 Hello world!

In Informatica è buona pratica iniziare sempre con un programmino semplice che stampi una scritta di saluto. Noi ovviamente non ci sottraeremo dalla tradizione, per cui faremo lo stesso.

File principale Dal nostro progetto vuoto inseriamo il seguente codice nell'unico file che abbiamo:

```

1 \input{res/config/package}
2 \input{res/config/config}
3
4 \author{Davide Polonio}
5 \date{27/07/2017}
6 \title{Primo progetto di Prova}
7
8 \begin{document}
9
10 \maketitle
11
12 \input{res/listOfSections}
13
14 \end{document}
```

Cosa stiamo dicendo qui? Stiamo dicendo prima di tutto che vogliamo includere il file contenente la lista dei pacchetti (che dobbiamo ancora scrivere) e che vogliamo includere il file con le configurazioni dei pacchetti che abbiamo incluso prima. Dopo abbiamo settato l'autore, la data, e il titolo del nostro documento. Finito il preambolo abbiamo iniziato con il contenuto, in cui abbiamo detto a \LaTeX di creare una pagina standard di copertina, e poi gli diciamo di andare ad includere anche i contenuti che sono presenti nel file `listOfSections` che si trova nella sottocartella che andremo a creare `res/`.

File `res/config/package.tex` Dalla interfaccia di `TeXnicCenter` andiamo a cliccare il bottone per creare un nuovo file (che è accanto a quello per creare un nuovo progetto, in alto a sinistra). Si aprirà una scheda con un nuovo file vuoto. Qui scriviamo:

```

1 \documentclass[12pt]{book}
2
3 \usepackage{graphicx}
4 \usepackage{float}
```

Qui gli stiamo specificando come vogliamo creare un documento che sarà un libro, di usare il carattere 12 quando non specificato e di includere due pacchetti: `graphicx` e `float`. Quando andiamo a salvare ora andiamo nel nostro progetto e creiamo due nuove cartelle, `res` e dentro questa cartella creiamo `config`. All'interno di quest'ultima salviamo il nostro file, impostando la voce `Encoding` in `UTF-8`⁵.

Ora passiamo al file successivo.

⁵Questo ci serve per aggiungere compatibilità con gli accenti

File res/config/config.tex Andiamo a creare questo file come al solito. In questo progetto non abbiamo nessuna configurazione particolare, e quindi lo possiamo lasciare vuoto. In caso dovremo inserire qualche configurazione potremmo modificarlo più avanti.

File res/listOfSections.tex Creiamo questo file con il seguente contenuto:

```
1 \input{res/sections/parte1}
```

Anche in questo caso, facciamo attenzione alla voce **Encoding** quando andiamo a salvarlo. Questo file conterrà solamente inclusioni ai veri contenuti di testo, permettendoci di escludere parti o riordinarle senza fare fastidiosi copia-incolla.

Passiamo ora al file nel quale mettere veramente il nostro contenuto.

File res/sections/parte1.tex Per prima cosa, con Esplora Risorse prendiamo l'immagine che più ci piace e copiamola in **res/img/**(creando la cartella **img**). Creando un nuovo file, ora inseriamo questo codice:

```
1 Hello world!
2
3 \begin{figure}[H]
4   \centering
5   \includegraphics[scale=0.5]{res/img/helloWorld}
6   \caption{Immagine bellissima}
7   \label{fig:immagine_saluto}
8 \end{figure}
```

Cosa abbiamo qui? Prima di tutto, abbiamo una normale scritta, che contiene il nostro famoso saluto. Proviamo ora a includere la immagine. Ci sono due modi per farlo:

- Scrivere direttamente il codice (scelta consigliata)
- Includere l'immagine tramite TeXnicCenter

Nel primo caso, dobbiamo solo copiare il codice, mentre nel secondo caso andiamo a cliccare l'icona a forma di immagine che si trova in alto a sinistra (per chi non trovasse l'icona veda l'immagine 2.1, nel quadrato con l'etichetta *Elenchi puntati, gestione immagini*). A questo punto si aprirà una finestrella che ci chiederà di inserire il percorso dell'immagine e altre opzioni. Fatto ciò andiamo a selezionarla premendo sul bottone con i tre puntini.

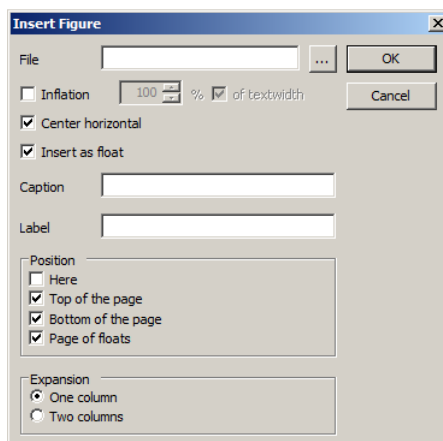


Figura 2.3: Finestra per l’inserimento di una nuova immagine nel documento

Selezioniamo anche la spunta “Here” e inseriamo una *caption* e una *label* che più ci aggradano (non inserite caratteri speciali nella *label* o il documento non compilerà più).

Compilazione Prima di compilare settiamo l’output di compilazione. Nel menù a tendina selezioniamo la voce “LaTeX → PDF”. Ora clicchiamo il quinto bottone a partire dal menù (muovendoci da sinistra verso destra). Se abbiamo seguito le istruzioni correttamente, si aprirà il visualizzatore PDF con quanto abbiamo scritto. Complimenti! Avete creato il vostro primo PDF usando L^AT_EX.

Dove trovare il progetto Se volete vedere il documento di prova che abbiamo fatto fino ad adesso (e per dimostrarvi che non vi abbiamo trollato ma che l’abbiamo fatto anche noi e che funziona) potete recarvi alla seguente pagina web e guardare i singoli file <https://github.com/R-and-LaTeX/GuidaGalatticaPerLaTeX/tree/develop/res/examples/Prova>.

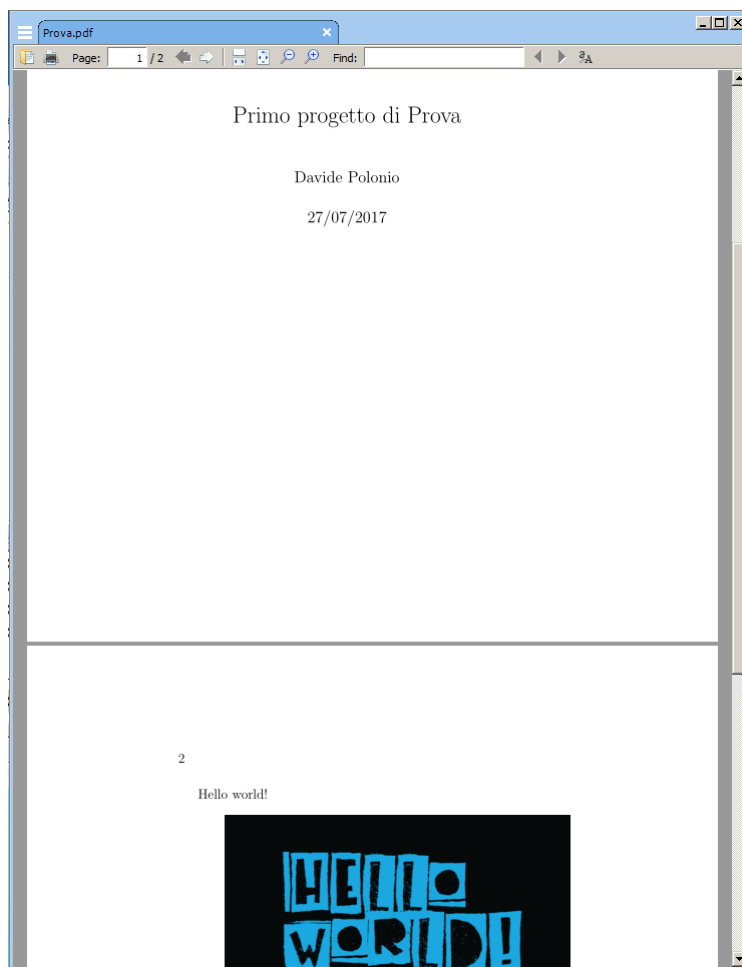


Figura 2.4: Il risultato che dovremmo ottenere

2.2 Compilare su Linux/MacOS

La compilazione di un progetto su Linux/MacOS risulta più semplice e fortunatamente più facile da spiegare. In generale, dopo aver installato tutte le dipendenze necessarie per l'ambiente \LaTeX , basta solitamente aprire un terminale nella `root` del progetto e dare il comando magico `latexmk -pdf` che si occuperà di compilare il tutto il numero giusto di volte.

2.3 Suddivisione del progetto in più file

Come abbiamo visto prima, anche per un semplice progetto come un *Hello World* può risultare nella creazione di 4-5 file. Non preoccupatevi! È un fenomeno normale in \LaTeX . Per includere diversi file in \LaTeX esistono due comandi:

- `\input{ }` questo comando concatena i documenti considerandoli come se fosse uno unico, e permette di concatenare file fino a qualsiasi profondità.
- `\include{ }` la seguente *keyword* inserisce della magia, come un'interruzione di pagina e creare altri file utili a tempi di compilazione. Questo comando

è solitamente utile quando si ha un progetto grande su computer molto lento, in quanto il cambiamento di un file non comporta la ricompilazione completa di tutto il documento, anche se non supporta inclusioni annidate (chiamare `\include{ }` su un altro file che è già stato incluso con `\include{ }` produrrà un errore). Per quello che ci serve a noi, è meglio usare sempre il primo comando.

Suddivisione del contenuto in `res/sections/` In generale, non c'è una regola predefinita per suddividere il proprio contenuto: questo può essere suddiviso per capitoli se per esempio si sta scrivendo una testi, oppure per sezioni se invece si sta scrivendo una pubblicazione scientifica, il tutto dipende dalla granularità che si vuole dare al documento, e alla facilità con cui si vogliono spostare gli argomenti.

Primi passi

Possiamo finalmente passare alla creazione del nostro primo documento in \LaTeX !

```

1 \documentclass{article}
2 \usepackage[T1]{fontenc}
3 \usepackage[utf8]{inputenc}
4 \usepackage[italian]{babel}
5
6 \begin{document}
7     Ecco il mio primo documento con \LaTeX{}.
8 \end{document}
```

Non ci sorprenderà molto che, compilando questo file, ci creerà un file PDF con all'interno scritto “*Ecco il mio primo documento con \LaTeX .*”. Non sempre, ahimè, tutto va come vorremmo. Infatti bisogna scrivere le cose in maniera corretta altrimenti non verrà prodotto il documento che vogliamo. In questo caso per esempio abbiamo un errore.

```

1 \documentclass{article}
2
3 \begin{document}
4     Ecco il mio primo documento con \LaTeX{}.
5 \end{docment} %manca la 'u' in document
```

Sarebbe molto bello che il computer ci dicesse Ehi guarda che manca una “u” ma purtroppo ci dirà ! `LaTeX Error: \begin{document} ended by \end{docment}.`, meno espressivo di certo ma ci aiuta a capire lo stesso dove sbagliamo. Abituatevi! Quando si ha a che fare con i linguaggi di programmazione (anche se magari non è la vostra aspirazione) non viene mai detto effettivamente cosa c’è che non va.

3.1 Come è diviso un documento

Come avete magari già notato nell’esempio, il documento si divide in due parti:

- la prima, dove ci sono cose tipo `\documentclass{...}` e `\usepackage{...}` viene detta **preambolo**. In questa sezione vengono definiti i pacchetti che il documento utilizza (vedremo in seguito) e il tipo di documento (appena dopo, un po’ di pazienza!);

- la seconda, invece, che inizia e finisce rispettivamente con `\begin{document}` e `\end{document}` viene detta **corpo** e contiene tutto il testo del documento.

3.2 Tipi di documento

L^AT_EX dà la possibilità di scrivere differenti tipi di documento a seconda delle necessità degli utenti. Il tipo di documento è specificato tramite il comando `\documentclass` a cui, tra parentesi graffe, viene indicato che tipologia si vuole. Le principali tipologie sono:

- **article**: è la tipologia più comune, serve per scrivere piccoli articoli oppure articoli per una rivista;
- **book**: si utilizza per scrivere documenti lunghi (es. la tesi) e libri;
- **report**: anche questa classe è utilizzata per articoli lunghi;
- **letter**: per le lettere;
- **slide**: serve per fare slide, anche se è poco utilizzato;
- **beamer**: anche questo usato per la creazione di slide.

3.2.1 Esempi

3.2.1.1 Esempio di documento book

```

1 \documentclass{book}
2
3 \begin{document}
4   \chapter{Titolo del capitolo}
5   Questo si pu\`o fare solo nella \verb!\documentclass{book
6     }! e in
7   \verb!\documentclass{report}!
8   \section{Titolo della sezione}
9   Questo invece si pu\`o fare sia nella \verb!\
10  \documentclass{book}! che in
11  \verb!\documentclass{article}!
12 \end{document}
```

3.2.1.2 Esempio di documento report

```

1 \documentclass{report}
2
3 \begin{document}
4   \chapter{Titolo del capitolo}
5   Questo \`e un capitolo
6   \section{Titolo della sezione}
```

```

7      Questo \‘e una sezione
8      \subsection{Titolo della sotto-sezione}
9      Questo \‘e una sotto-sezione
10     \subsubsection{Titolo della sotto-sotto-sezione}
11     Questo \‘e una sotto-sotto-sezione
12 \end{document}

```

3.2.1.3 Esempio di documento beamer

```

1 \documentclass{beamer}
2
3 \begin{document}
4     \begin{frame}{Titolo della slide}
5         Esempio di una slide
6     \end{frame}
7 \end{document}

```

3.3 Sezionare un documento

Per suddividere un documento è possibile utilizzare molti comandi differenti, che seguono le convenzioni presenti nei paesi anglosassoni e sono (ordinati per “grandezza” logica):

`\part` per strutturare il nostro documento in parti. Questo comando non influisce sulla numerazione dei capitoli ed è disponibile per le classi **book** e **report**;

`\chapter` per dividere in capitoli. Anche questo comando è disponibile per le classi **book** e **report**;

`\section` corrisponde più o meno ad un paragrafo;

`\subsection` per dividere in sottoparagrafi;

`\subsubsection` per dividere in sotto-sottoparagrafi;

`\paragraph` è l’equivalente di una sezione;

`\subparagraph` sezionamento di livello più basso possibile.

È spesso consigliato di non sezionare il documento troppo finemente. Le divisioni fino a `subsection` sono aggiunte automaticamente all’indice e vengono automaticamente numerate. Se per qualche motivo si vuole che una di queste divisioni non appaia nell’indice è necessario postporre al nome del comando un asterisco (per esempio `\chapter*Nome capitolo`). Infine per mostrare l’indice delle sezioni è sufficiente dare il comando `\tableofcontents`.

3.4 Package

Nella scrittura di un documento, talvolta abbiamo necessità di aggiungere alcune cose magiche. In italiano, come in altre lingue a dir la verità, abbiamo un sacco di accenti e vorremmo far sì che questi appaiano nel nostro documento. Una possibilità per scrivere, per esempio, la “è” è `\’e` che però è abbastanza scomoda. In nostro aiuto, però, vengono i pacchetti. Possiamo vedere appunto i pacchetti come delle cose magiche scritte da altri che ci aiutano a scrivere in maniera più semplice. Vediamo un esempio.

```
1 \documentclass{article}
2
3 \begin{document}
4   Sarà così difficile mettere gli accenti in \LaTeX{}?
5 \end{document}
```

```
1 \documentclass{article}
2 \usepackage[utf8]{inputenc}
3
4 \begin{document}
5   Sarà così difficile mettere gli accenti in \LaTeX{}?
6 \end{document}
```

Se proviamo a compilare il primo dei due esempi gli accenti non appariranno, mentre nel secondo sì, poichè abbiamo utilizzato il pacchetto `\usepackage[utf8]{inputenc}`. Questi *packages* ci permettono di fare cose ancora più magiche. Provare il seguente documento per credere!

```
1 \documentclass{article}
2 \usepackage{tikz}
3 \usepackage{verbatim}
4 \usepackage[active,tightpage]{preview}
5 \PreviewEnvironment{tikzpicture}
6 \setlength\PreviewBorder{5pt}%
7 \usepackage{tikz-3dplot}
8
9
10 \begin{document}
11
12 \tdplotsetmaincoords{70}{0}
13 \begin{tikzpicture}[tdplot_main_coords]
14 \def\RI{2}
15 \def\RII{1.25}
16
17 \draw[thick] (\RI,0)
18   \foreach \x in {0,300,240,180} { -- (\x:\RI) node at (\x:\RI) (R1-\x) {} };
19 \draw[dashed,thick] (R1-0.center)
20   \foreach \x in {60,120,180} { -- (\x:\RI) node at (\x:\RI) (R1-\x) {} };
```

```

21 \path[fill=gray!30] (\RI,0)
22   \foreach \x in {0,60,120,180,240,300} { -- (\x:\RI)};
23
24 \begin{scope}[yshift=2cm]
25 \draw[thick,fill=gray!30,opacity=0.2] (\RII,0)
26   \foreach \x in {0,60,120,180,240,300,360} { -- (\x:\RII)
27     node at (\x:\RII) (R2-\x) {} };
28 \end{scope}
29 \foreach \x in {0,180,240,300} { \draw (R1-\x.center)--(R2-\x
30   .center); };
31 \foreach \x in {60,120} { \draw[dashed] (R1-\x.center)--(R2-\
32   x.center); };
33 \end{tikzpicture}
\end{document}

```

P.S. Noi non vedremo mai cose così magiche.

Comandi Base

Di seguito verranno illustrati i comandi base di \LaTeX , utili per scrivere qualsiasi tipo di documento.

4.1 Liste

Le liste nei documenti possono risultare molto utili quando si tratta di elencare oggetti in sequenza, in quanto aumentano la leggibilità e scorrevolezza di un testo, non appesantendone la lettura.

La lista è composta fondamentalmente da 3 comandi:

- `\begin{}` per dichiarare che si sta iniziando una lista puntata
- `\item` per indicare un punto della lista
- `\end{}` per segnalare che la lista puntata è terminata.

Esistono due tipologie di liste: elenchi puntati ed elenchi numerati. Per gli elenchi **puntati** si utilizza `itemize` all'interno delle parentesi di `begin` e `end`. Se invece si vogliono utilizzare gli elenchi **numerati** la parola da inserire all'interno di `begin` e `end` è `enumerate`.

4.1.1 Esempi

Per far capire meglio quanto spiegato verranno illustrati un paio di esempi.

Prima di tutto vediamo una lista puntata:

```
1 \begin{itemize}
2   \item Una
3   \item lista
4   \item puntata
5 \end{itemize}
```

Questo è invece il codice per una lista numerata:

```
1 \begin{enumerate}
2   \item Ecco
3   \item una
4   \item lista
5   \item numerata
6 \end{enumerate}
```

4.1.2 Personalizzare gli elenchi puntati

Gli elenchi sono interessanti, ma a volte potremmo volere qualcosa di più. Fortunatamente L^AT_EX è stato progettato anche con questa possibilità, ed è possibile applicare piccoli trucchetti per rendere i propri elenchi puntati “speciali”.

4.1.2.1 Item personalizzati

Gli item possono essere personalizzati e al posto dei punti inserire un simbolo o un nome personalizzato. Vediamo subito un esempio.

Scrivendo a esempio il seguente codice:

```
1 \begin{itemize}
2   \item[Uno] Questo
3   \item[Due] elenco
4   \item[Tre] risulta
5   \item[Quattro] personalizzato
6 \end{itemize}
```

Che produrrà:

Uno Questo

Due elenco

Tre risulta

Quattro personalizzato

Come possiamo notare, non sempre l’aspetto che otteniamo è bello, ma questo è un altro discorso.

4.1.2.2 Personalizzare gli elenchi puntati

A volte il classico pallino degli elenchi puntati è proprio brutto, e vorremmo sostituirlo con qualcosa di diverso. Anche questa operazione è piuttosto semplice, e va eseguita nel preambolo.

Il codice da inserire è il seguente:

```
\renewcommand{\labelitemi}{\ast}
```

Questo ci modificherà tutti gli elenchi puntati, mettendo al posto del classico pallino un asterisco, “*”. Ci sono altre opzioni, come per esempio `\bullet`, `\cdot` e `\diamond`.

4.2 Commenti

A volte è utile inserire nel codice (non mi riferisco solamente a \LaTeX ma a qualsiasi linguaggi di programmazione) dei commenti che per esempio specificano cosa un comando fa o cosa si deve cambiare, senza che questi però vengano visualizzati nel PDF in *output*. Per fare ciò è sufficiente usare il carattere `%`. In questo modo tutto quello che appare dopo tale carattere non verrà visualizzato nel PDF ma rimarrà solamente nel vostro codice.

4.3 Formattazione di base

Non c'è da sorprendersi che su un linguaggio progettato per scrivere documenti esista la formattazione di base.

Grassetto Per scrivere in grassetto il comando da dare è il seguente: `\textbf{testo da scrivere in grassetto}`.

Corsivo Per il corsivo il comando da dare è invece: `\textit{testo in corsivo}`.

Sottolineato Il sottolineato si fa invece con: `\underline{testo sottolineato}`.

Testo barrato Per cancellare il testo: `\sout{Testo da cancellare}`¹.

4.4 A spasso nella spaziatura

È anche utile sapere come andare a capo riga nei momenti giusti e anche come inserire spazi appositamente nel testo. A differenza di Word dove viene commesso il crimine immondo di inserire invii finché non si raggiunge la spaziatura desiderata, in \LaTeX è possibile inserire quantità di spazio precise, compiendo un lavoro più pulito.

4.4.1 A capo riga

Esistono diversi comandi per eseguire la stessa azione. In questo caso, essi sono:

- `\\`
- `*`

¹Nota che per il testo sbarrato è necessario aggiungere un altro pacchetto, scrivendo `\usepackage[normalem]{ulem}`. La parte relativa ai pacchetti verrà trattata meglio più avanti.

- `\newline`

È importante far notare che non è possibile eseguire una cosa del genere in \LaTeX per andare a capo più volte²:

```
1 \newline
2 \newline
3 \newline
4 \newline
5 \newline
6 \newline
```

Esiste un comando specifico per fare ciò, che sarà subito spiegato.

4.4.2 Inserire pagine

L'inserimento di una nuova pagina è molto semplice e il comando per creare una nuova pagina è `\newpage`.

4.4.3 Vuoti cosmici nel testo

Avere delle parti vuote nel testo può essere utile, si pensi solamente per esempio se vogliamo creare una prima pagina diversa, e in generale può servire per personalizzare la pagina.

Spaziatura verticale Per far ciò esiste un comando magico, chiamato `\vspace{ }` che permette di eseguire una spaziatura orizzontale del testo. Quantità dello spazio viene misurata solitamente in pixel, quindi ad esempio per inserire una spaziatura verticale di 25px sul testo dovreste scrivere `\vspace{25px}`. Questo produrrà

che infine è proprio quello che volevamo (si, sopra abbiamo inserito una spaziatura verticale).

Spaziatura orizzontale La spaziatura orizzontale si esegue invece con il comando `\hspace{ }` e senza sorprese esegue una spaziatura di tipo orizzontale.

4.4.4 Riempire i vuoti cosmici

Ripensandoci, alla fine non è proprio così brutto alle volte riempire quel freddo spazio bianco che avvinghia i poveri elementi del nostro amato documento. Siccome siamo a bordo della nostra nave spaziale, è di certo equipaggiata con degli strumenti che ci permettono di fare ciò. Questi strumenti in \LaTeX si chiamano `hfill` e `vfill`, e servono rispettivamente per riempire spazi orizzontali e verticali. Questa descrizione sembra un poco povera ed è normale mettere la loro utilità in discussione, ma vedremo che questi due strumenti si riveleranno utili quando andremo a manipolare immagini ed ad inserirle.

²Si, l'abbiamo messo per aumentare il numero di pagine di questa guida.

4.4.5 Orientare le pagine

No, non stiamo parlando di come far trovare la posizione alle pagine che si sono perse, ma vi spiegheremo come sia possibile avere pagine orientate in maniera orizzontale in un documento con pagine verticali. Vedremo poi come invece applicare un'orientazione orizzontale ad un intero documento.

4.4.5.1 Una pagina orizzontale

Per avere solo una pagina orizzontale i passi da fare non sono molti, e consistono a dire al signor \LaTeX che vogliamo che la prossima pagina sia ruotata di 90 gradi.

Vediamo un esempio completo:

```

1 \documentclass{article} % Non succede nulla cambiandolo
2
3 \usepackage[utf8]{inputenc} % Per i caratteri italiani
4 \usepackage{pdflscape} % Per l'orientazione di singole pagine
5
6 \begin{document}
7
8 Hey, questo è un documento orientato verticalmente, ma la
9 prossima pagina la posizioneremo orizzontalmente.
10 \newpage
11
12 % Inizia la pagina orientata orizzontalmente
13 \begin{landscape}
14     % Ora la pagina è in orizzontale!
15     Questa pagina è invece orizzontale a quanto pare.
16 \end{landscape}
17 \end{document}
```

4.4.6 Ma io voglio tutto il documento orizzontale!

E va bene, non c'è bisogno di scaldarsi tanto! Ora vediamo come creare un documento orientato orizzontalmente. Per farlo ci sono due modi:

1. Impostando la rotazione direttamente in `documentclass`
2. Impostando la rotazione tramite il pacchetto `geometry`

Impostando la rotazione da `geometry` La rotazione del documento da `geometry` è impostabile in questa maniera:

```
1 \documentclass{article} % Può essere di qualsiasi tipo
2
3 % Impostiamo la rotazione del documento
4 \usepackage[a4paper,margin=1in,landscape]{geometry}
5
6 \usepackage{blindtext} % Serve per generare testo-riempitivo
7
8 \begin{document}
9   \blinddocument % Generazione di testo casuale
10 \end{document}
```

4.5 Tabelle

Siamo arrivati ad un argomento spinoso: le tabelle. Come potete immaginare, la rappresentazione di una tabella in un documento di testo non dev'essere nulla di semplice. Immaginate con \LaTeX , dove c'è la possibilità di specificare le linee da visualizzare e di poter dire come specificare la formattazione del testo all'interno di ogni cella: insomma, un vero disastro. A spezzare una lancia a favore delle tabelle in \LaTeX c'è la loro estrema flessibilità e personalizzazione: vedremo com'è possibile creare tabelle molto avanzate, senza rischiare che si disintegrino alla prima modifica.

4.5.1 Tabelle semplici

Iniziamo con una tabella semplice. Realizzare una tabella senza troppe pretese è semplice dopo che si è capito come funziona.

Prima di tutto, per dichiarare che si sta creando una tabella bisogna scrivere `\begin{table}[]`, dove all'interno delle parentesi quadre è possibile specificare la posizione di dove vogliamo la tabella. Se la vogliamo nella posizione di dove la stiamo dichiarando scriveremo `[H]`³.

Dopodiché bisogna dire che vogliamo una tabulazione, e possiamo specificare il numero di colonne che vogliamo avere. Per fare ciò scriveremo `\begin{tabular}{}`, dove nel secondo paio di `{ }` specificheremo il numero di colonne. Vedremo più avanti nel dettaglio questa parte, per il momento fidatevi.

4.5.1.1 Esempietto

Vediamo come creare una semplice tabella:

```

1 \usepackage{eurosym} % Serve solo per il simbolo dell'euro
2 \begin{table}[H]
3 \centering
4 \begin{tabular}{llll}
5 Articoli      & Identificativo & Quantità & Costo \\
6 Matite        & Cancelleria   & 100      & 20 \euro{} \\
7 Penne biro    & Cancelleria   & 120      & 15 \euro{} \\
8 Gomme        & Cancelleria   & 40       & 60 \euro{} \\
9 Raccoglitori  & Ufficio       & 30       & 12 \euro{} \\
10 \end{tabular}
11 \caption{Una tabella d'esempio}
12 \label{tabella:esempio}
13 \end{table}

```

Che risulterà in questa tabella:

³Nota che è necessario importare il pacchetto `float`. Se non volete farlo, potete scrivere `h!` che vi darà un risultato simile

Articoli	Identificativo	Quantità	Costo
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Raccoglitori	Ufficio	30	12 €

Tabella 4.1: Una tabella d'esempio

A questo punto ci piacerebbe però avere dei bordi sulla tabella. Niente di più facile! Per aggiungere i bordi alla tabella dobbiamo usare `\hline` per le linee orizzontali, mentre per avere le linee verticali (ovvero tra colonne) dobbiamo modificare il `\begin{tabular}{llll}` nella seguente maniera⁴:

4.5.1.2 Esempietto

Vediamo come creare una semplice tabella:

```

1 \usepackage{eurosym} % Serve solo per il simbolo dell'euro
2 \begin{table}[H]
3 \centering
4 \begin{tabular}{|l|l|l|l|} % Notare le sbarrette
5 \hline % Notare gli \hline su ogni riga
6 Articoli & Identificativo & Quantità & Costo \\
7 \hline
8 Matite & Cancelleria & 100 & 20 \euro{}\\
9 \hline
10 Penne biro & Cancelleria & 120 & 15 \euro{}\\
11 \hline
12 Gomme & Cancelleria & 40 & 60 \euro{}\\
13 \hline
14 % Attenzione che nella riga successiva abbiamo dovuto
15 % aggiungere un \\ in più, in quanto vogliamo inserire
16 % una linea nella riga successiva.
17 Raccoglitori & Ufficio & 30 & 12 \euro{}\\
18 \hline
19 \end{tabular}
20 \caption{Una tabella d'esempio}
21 \label{tabella:esempio2}
22 \end{table}

```

Che ci darà il seguente risultato:

⁴Nota che in \LaTeX l'utilizzo di linee verticali è a quanto pare sconsigliato in quanto peggiora la leggibilità della tabella.

Articoli	Identificativo	Quantità	Costo
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Raccoglitori	Ufficio	30	12 €

Tabella 4.2: Una tabella d'esempio con i bordi

Una nota importante da considerare è la possibilità di personalizzare dove posizionare i bordi o meno: per la vostra curiosità, provate a rimuovere uno o due `\hline` o qualche sbarretta sull'inizializzazione della tabella, e vedrete come la tabella si “evolve”.

Online esistono tool che permettono la creazione di semplici tabelle in maniera visuale. Sono molto utili e permettono di risparmiare un sacco di tempo, per cui noi li consigliamo. Un buon sito per la generazione di tabelle (che permette cose tra cui l'inserimento di bordi in maniera visuale) è disponibile all'indirizzo <http://www.tablesgenerator.com/>.

4.5.2 Tabelle su più pagine

A volte capita di avere una grande mole di dati da mettere in una tabella, e che questi dati non ci stiano su una pagina. Se avete provato a fare una tabella più grande di una pagina avrete sicuramente notato di come questa continui ad andare anche fuori dal termine della pagina stessa, sbattendosene di qualsiasi bordo presente. Siccome questa *feature* non piaceva a tutti, qualche programmatore brufoloso e annoiato nel seminterrato di sua madre si è inventato una bella sera⁵ di implementare le tabelle su più pagine e di pubblicare ciò su un pacchetto a parte, `longtable`. Per usare `longtable` quindi bisogna includere l'omonimo pacchetto nel preambolo del documento, con il solito comando a cui siamo abituati, ovvero `\usepackage{longtable}`. La creazione di una tabella sarà simile a quella di prima, con qualche piccola differenza che ora vedremo in questo esempio:

```

1 \usepackage{eurosym} % Serve solo per il simbolo dell'euro
2 % Serve per fare le tabelle multi-pagine!
3 \usepackage{longtable}
4 % Posizioniamo la tabella in centro del foglio
5 \begin{center}
6 \begin{longtable}{llll} % Usiamo longtable ora
7 Articoli      & Identificativo & Quantità & Costo \\
8 Matite        & Cancelleria   & 100      & 20 \euro{}\\
9 Penne biro    & Cancelleria   & 120      & 15 \euro{}\\
10 Gomme        & Cancelleria   & 40       & 60 \euro{}\\
11 Matite        & Cancelleria   & 100      & 20 \euro{}\\
12 Penne biro    & Cancelleria   & 120      & 15 \euro{}\\
13 Gomme        & Cancelleria   & 40       & 60 \euro{}\\
14 Matite        & Cancelleria   & 100      & 20 \euro{}\\
15 Penne biro    & Cancelleria   & 120      & 15 \euro{}\\

```

⁵L'habitat preferito per un programmatore.

```

16 Gomme          & Cancelleria    & 40          & 60 \euro{}\
17 Matite         & Cancelleria    & 100         & 20 \euro{}\
18 Penne biro     & Cancelleria    & 120         & 15 \euro{}\
19 Gomme          & Cancelleria    & 40          & 60 \euro{}\
20 Matite         & Cancelleria    & 100         & 20 \euro{}\
21 Penne biro     & Cancelleria    & 120         & 15 \euro{}\
22 Gomme          & Cancelleria    & 40          & 60 \euro{}\
23 Matite         & Cancelleria    & 100         & 20 \euro{}\
24 Penne biro     & Cancelleria    & 120         & 15 \euro{}\
25 Gomme          & Cancelleria    & 40          & 60 \euro{}\
26 Matite         & Cancelleria    & 100         & 20 \euro{}\
27 Penne biro     & Cancelleria    & 120         & 15 \euro{}\
28 Gomme          & Cancelleria    & 40          & 60 \euro{}\
29 Matite         & Cancelleria    & 100         & 20 \euro{}\
30 Penne biro     & Cancelleria    & 120         & 15 \euro{}\
31 Gomme          & Cancelleria    & 40          & 60 \euro{}\
32 Matite         & Cancelleria    & 100         & 20 \euro{}\
33 Penne biro     & Cancelleria    & 120         & 15 \euro{}\
34 Gomme          & Cancelleria    & 40          & 60 \euro{}\
35 Matite         & Cancelleria    & 100         & 20 \euro{}\
36 Penne biro     & Cancelleria    & 120         & 15 \euro{}\
37 Gomme          & Cancelleria    & 40          & 60 \euro{}\
38 Raccoglitori  & Ufficio        & 30          & 12 \euro{}\
39
40 % Notare come la caption è dentro longtable
41 \caption{Una tabella multipagina d'esempio}
42 % Idem per la label
43 \label{tabella:esempio_tabella_lunga}
44 \end{longtable}
45 \end{center}

```

Questa risulta in:

Articoli	Identificativo	Quantità	Costo
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €

Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Matite	Cancelleria	100	20 €
Penne biro	Cancelleria	120	15 €
Gomme	Cancelleria	40	60 €
Raccoglitori	Ufficio	30	12 €

Tabella 4.3: Una tabella multipagina d'esempio

Le tabelle su più pagine possono diventare estremamente complicate, e quello che abbiamo presentato è solamente un accenno, giusto per farvi vedere come funzionano e che esistono. Alleghiamo qui un esempio di come una tabella può evolvere in qualcosa di estremamente complicato ma molto personalizzato. Questa è da considerarsi stregoneria, e per i propositi del nostro corso non verrà spiegata.

```

1 \begin{center}
2 \begin{longtable}{|l|l|l|}
3 \caption[Feasible triples for a highly variable Grid]{
4   Feasible triples for
5   highly variable Grid, MLMMH.} \label{grid_mlmh} \\
6 \hline \multicolumn{1}{|c|}{\textbf{Time (s)}} & \multicolumn{
7   {1}{c|}{\textbf{Triple chosen}} & \multicolumn{1}{c|}{\
8   \textbf{Other feasible triples}} \\ \hline
9 \endfirsthead
10 \multicolumn{3}{c}{\
11 \bfseries \tablename\ \thetable{} -- continued from
12   previous page}} \\
13 \hline \multicolumn{1}{|c|}{\textbf{Time (s)}} & \
14   \multicolumn{1}{c|}{\textbf{Triple chosen}} & \
15   \multicolumn{1}{c|}{\textbf{Other feasible triples}} \\ \hline
16 \endhead
17 \hline \multicolumn{3}{|r|}{Continued on next page} \\ \hline
18 \endfoot

```

```

18
19 \hline \hline
20 \endlastfoot
21
22 2745 & (1, 12, 10980) & (1, 13, 8235), (2, 2, 0), (2, 3, 0),
      (3, 1, 0) \\
23 5490 & (1, 12, 13725) & (2, 2, 2745), (2, 3, 0), (3, 1, 0) \\
24 131760 & (2, 2, 2745) & (2, 3, 0), (3, 1, 0) \\
25 134505 & (1, 13, 16470) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
26 137250 & (1, 13, 13725) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
27 139995 & (2, 2, 2745) & (2, 3, 0), (3, 1, 0) \\
28 142740 & (2, 2, 2745) & (2, 3, 0), (3, 1, 0) \\
29 148230 & (2, 2, 2745) & (2, 3, 0), (3, 1, 0) \\
30 150975 & (1, 13, 16470) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
31 153720 & (1, 12, 13725) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
32 156465 & (1, 13, 13725) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
33 159210 & (1, 13, 13725) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
34 161955 & (1, 13, 16470) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
35 164700 & (1, 13, 13725) & (2, 2, 2745), (2, 3, 0), (3, 1, 0)
      \\
36 \end{longtable}
37 \end{center}

```

4.5.3 Altre informazioni sulle tabelle

Se pensavate fosse tutto finito qui, vi state sbagliando di grosso. Il signor \LaTeX permette di personalizzare ulteriormente le nostre tabelle, e qui verranno spiegati alcuni piccoli trucchi che pensiamo possano essere utili a voi lettori.

4.5.3.1 Posizionamento del testo nelle celle

A volte potrebbe essere utile avere la possibilità di centrare o posizionare il testo di una tabella in maniera particolare. Questo è ovviamente possibile, e per farlo bisogna modificare il modo in cui dichiariamo inizialmente la tabella. Quando mettiamo la dicitura `l` in `\begin{tabular}{lllll}` stiamo dicendo infatti che vogliamo che il testo di tutte le celle nella tabella sia orientato a sinistra. Esistono diverse opzioni per questo comando, che sono `l`, `c` e `r`.

l	colonne aggiustate a sinistra
c	colonne centrate
r	colonne aggiustate a destra

Tabella 4.4: Elenco delle opzioni che si possono mettere nella voce `tabular`.

4.5.3.2 Impostare manualmente la grandezza delle celle

Se non viene specificato nessuna impostazione, in \LaTeX la larghezza delle celle viene impostata automaticamente in maniera tale da ottenere il miglior effetto visibile per il lettore. Sfortunatamente, questo non sempre capita, e a volte impostare manualmente la larghezza delle colonne può risultare utile. L'impostazione della larghezza avviene sempre nella linea `\begin{tabular}{ }`, e al posto di `l`, `c` e `r` andremo a mettere un altro parametro, che si chiama `p`.

Come funziona la keyword `p` La *keyword* `p` ha bisogno di un parametro per funzionare, e questo parametro è la larghezza in `cm` che si vuole della colonna. Vediamo subito un esempio per chiarirci le idee:

```

1 \begin{table}[H]
2 \centering
3 \begin{tabular}{|p{2cm}|p{10cm}|}
4 \hline
5 È questa colonna sporporzionata? & Si \\ \hline
6 \end{tabular}
7 \caption{Una tabella con la larghezza delle colonne
8   personalizzata}
9 \label{tab:es_sporporzionata}
10 \end{table}

```

Questo codice ci darà il seguente risultato:

È questa colonna sproporzio- nata?	Si
---	----

Tabella 4.5: Una tabella con la larghezza delle colonne personalizzata

Come possiamo notare, bisogna prestare attenzione quando si impostano manualmente le dimensioni delle varie colonne, perché l'aspetto potrebbe risentirne (come in questo caso). È buona pratica quindi, se vogliamo impostare la dimensione delle colonne manualmente, eseguire delle prove visuali in modo da vedere se il risultato ottenuto ci soddisfa.

4.6 Immagini

Solitamente in un documento si ha la necessità di inserire delle immagini o per spiegare meglio il contenuto o per abbellire il testo. Per inserire le immagini in \LaTeX solitamente si utilizza il pacchetto `\graphicx` (quindi nel preambolo è necessario inserire `\usepackage{graphicx}`). Il comando da utilizzare è `\includegraphics[opzioni]{img_path}`, dove le opzioni sono opzionali⁶ e le più comuni sono:

- **width**: per la larghezza;
- **height**: per l'altezza;
- **scale**: per scalare l'immagine (1.0 è il 100%);
- **keepaspectratio**: per mantenere lo stesso rapporto larghezza-altezza dell'immagine originale

`img_path` serve per specificare il percorso dell'immagine da includere, che può essere assoluto o relativo (meglio specificare il percorso relativo, specificare il percorso assoluto è un po' come passare per casa se devi andare dall'aula al bagno). È possibile anche specificare nel preambolo, tramite il comando `\graphicspath{{default_path}}` il percorso di default per le immagini. In questo modo al posto di un percorso nel comando `\includegraphic` è possibile passare il nome dell'immagine voluta.

Come per le tabelle abbiamo bisogno di un "ambiente" che contenga le immagini, che in questo caso sarà `\begin{figure}...\end{figure}`.

È buona norma accompagnare le immagini con una didascalia che le spieghi. Per fare ciò ci viene in aiuto il comando `\caption{...}` che accetta tra le parentesi graffe del testo per descrivere le immagini.

Attenzione! Non tutti i formati di immagini sono supportati! I formati supportati sono **pdf**, **png** e **jpg**. Nelle nuove installazioni di \LaTeX è possibile utilizzare però anche il formato **eps**.

Di seguito possiamo vedere un esempio di come inserire un'immagine con didascalia in un documento \LaTeX .

```

1 \begin{figure}[H]
2   \centering
3   \includegraphics[scale=0.15]{immagine}
4   \caption{Didascalia dell'immagine}
5 \end{figure}

```

Il risultato sarà questo:

⁶No, non ci scusiamo per il gioco di parole

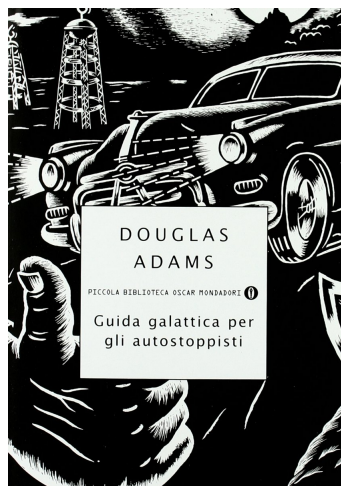


Figura 4.1: Didascalia dell'immagine

4.6.1 Affiancare le immagini

(Fate un respiro profondo che sennò qui si perde il proprio posto in paradiso)
 Per poter affiancare immagini è possibile utilizzare il *subfig* (Non è l'unica soluzione, ma è una delle più semplici). Questo permette di creare “delle piccole immagini” all'interno dell'ambiente `\begin{figure}...\end{figure}`. Specificando grandezza delle immagini o utilizzando i comandi per iniziare una nuova riga. È probabilmente più facile con un esempio.

```

1 \begin{figure}[H]
2   \subfloat[Didascalia immagine 1]{
3     \includegraphics[width=0.45\linewidth]
4       {immagine1}
5   }
6   \hfill
7   \subfloat[Didascalia immagine 2]{
8     \includegraphics[width=0.45\linewidth]
9       {immagine2}
10  }
11  \\
12  \subfloat[Didascalia immagine 3]{
13    \includegraphics[width=\linewidth]
14      {immagine3}
15  }
16  \\
17  \subfloat[Didascalia immagine 4]{
18    \includegraphics[width=0.60\linewidth]
19      {immagine4}
20  }
21  \caption{Didascalia globale delle 4 immagini}
22 \end{figure}

```

Il risultato sarà questo:



(a) Didascalìa immagine 1



(b) Didascalìa immagine 2



(c) Didascalìa immagine 3



(d) Didascalìa immagine 4

Figura 4.2: Didascalìa globale delle 4 immagini

Nell'esempio viene utilizzato il comando `\subfloat` per posizionare le quattro immagini al quale, tra parentesi quadre, viene specificata la didascalìa della singola

immagine, mentre tra parentesi graffe viene indicata l'immagine da mostrare. Con il comando `\caption` viene, infine, specificata la didascalia per le immagini raggruppate.

4.6.2 Affiancare testo e immagini

Sì si può anche affiancare testo e immagini. Serve? Bho vedete voi. In ogni caso è possibile utilizzando il *package* `wrapfig`. Questo funziona in modo analogo al solito modo di inserire le immagini, tranne che l'ambiente sarà `\begin{wrapfig}{pos}{dim}...\end{wrapfig}` dove `pos` può essere `r` o `l` e `dim` serve per specificare le dimensioni di quanto l'immagine deve “affiancarsi” al testo. Di seguito un esempio.

```
1 \lipsum[1]
2 \begin{wrapfigure}{r}{6cm}
3   \includegraphics[scale=0.4]{immagineafianco}
4   \caption{Didascalia dell'immagine}
5 \end{wrapfigure}
6 \lipsum[2-3]
```

Il risultato sarà questo (è stato utilizzato il comando `lipsum` per mettere del testo riempitivo perché non avevamo voglia di scrivere del testo sensato):

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.



Figura 4.3: Didascalia dell'immagine

prefisso	tipologia
sec:	section
subsec:	subsection
fig:	figure
tab:	tabella
eq:	formula
lst:	codice
itm:	elementi di lista numerata
alg:	algoritmo
app:	appendice

4.7 Note a piè di pagina

Spesso capita di dover mettere delle note a piè di pagina, per esempio per mettere un riferimento o spiegare meglio qualcosa⁷. Per fare ciò è necessario utilizzare il comando `\footnote{...}` specificando tra graffe il testo della nota. \LaTeX in automatico si occuperà della numerazione e del posizionamento.

4.8 Collegamenti ipertestuali

4.8.1 Collegamenti nel testo

Quando si inseriscono delle immagini e se ne fa riferimento nel testo oppure quando si scrive una parte e la si riprende in un'altra sezione del documento si sente la necessità di inserire un collegamento tra le due parti. Per fare ciò è necessario marcare con `\label{...}` le parti a cui si vuole fare riferimento. Tra parentesi graffe deve essere inserito un identificativo univoco all'interno del documento che serve per indicare una specifica parte del documento o oggetto in esso. Quando ci si riferisce ad esso, invece, si deve utilizzare `\ref{...}`, specificando tra graffe l'identificativo messo nella `label`. È possibile anche fare riferimento alla pagina in cui si trova l'oggetto marcato con `label` utilizzando `\pageref{...}` (e lo stesso identificativo).

4.8.1.1 Buone regole per gli identificativi

Solitamente, agli identificativi utilizzati dalle `label` e `ref` vengono anteposti alcuni caratteri che permettono di identificare che tipo di oggetto è referenziato. Questo non è di fatto obbligatorio, ma è una buona regola, almeno per un motivo strettamente pratico: ci si ricorda a che tipo di oggetto ci si sta riferendo! Sembra sciocco ma quando si inizia ad avere un documento di svariate pagine, con, magari, anche solo un centinaio di pagine, fare confusione è veramente facile. Altro pericolo che si evita è l'avere etichette con lo stesso nome. (Siete stati avvertiti, fate come volete a vostro rischio e pericolo)

⁷O solo per distrarre il lettore, come in questo caso.

4.8.2 Collegamenti al Web

Il pacchetto che permette l'inserimento in maniera corretta in un documento LaTeX di *link* o *URL* è `hyperref`. Ci sono due modalità principali per inserire queste informazioni. La prima è `\url{indirizzo}`, con la quale verrà mostrato l'URL in formato *monospace*, pensato anche per documenti che devono essere stampati, la seconda è `\href{indirizzo}{testo_da_mostrare}`, la quale non mostrerà esplicitamente l'indirizzo, ma mostrerà solamente il testo voluto (come succede nei siti Web per capirci), pensato per i documenti che devono essere fruiti in formato PDF.

4.8.2.1 Email

Per inserire gli indirizzi email nel testo è preferibile creare questo comando personalizzato, da inserire nel preambolo del documento

```
1 \newcommand{\mail}[1]{\href{mailto:#1}{\texttt{#1}}}
```

(per il momento fidatevi, spiegheremo in seguito come creare i propri comandi). L'utilizzo è molto semplice, infatti nel documento è sufficiente dare il comando `\email{...}` specificando l'indirizzo di posta elettronica voluto.

4.8.2.2 Personalizzazione dei link

È buona norma, almeno nella creazione dei siti Web, colorare i link con un colore differente dal testo normale, in modo da evidenziare a chi visita la pagina che lì ci si può cliccare. Anche nei documenti non fa male evidenziare questi indirizzi e il pacchetto `hyperref` permette la personalizzazione tramite il comando `\hypersetup{...}` da mettere nel preambolo del documento, al quale devono essere specificate le personalizzazioni volute. Le più importanti sono:

- **colorlinks=false** oppure **colorlinks=true**, che indica rispettivamente se si vuole che i link vengano riquadrati oppure colorati;
- **urlcolor=color** che serve per specificare il colore dei link (sostituite *color* con un colore in lingua inglese);

Maggiori informazioni possono essere reperite nel capitolo [Hyperlinks](https://en.wikibooks.org/wiki/LaTeX/Hyperlinks) su Wiki-Book (oppure come abbiamo imparato prima all'URL <https://en.wikibooks.org/wiki/LaTeX/Hyperlinks> così abbiamo anche un esempio dei due modi di inserire i collegamenti ai siti).

Formule matematiche

Una delle cose veramente buone di L^AT_EX è che ci dà la possibilità di scrivere le formule matematiche in maniera molto bella... Però c'è una sfilza lunghissima di comandi da utilizzare in base alle necessità. Per questo vedremo solamente la parte essenziale e per i vari simboli rimanderemo a questo indirizzo https://it.wikipedia.org/wiki/Aiuto:Formule_matematiche_TeX.

5.1 La modalità matematica

Ci sono due ambienti che ci permettono di scrivere formule matematiche. Il primo permette di scrivere formule *inline* e viene aperto e chiuso da \$ oppure rispettivamente da \ (e \). Esempio:

```
1 La terna pitagorica \ 'e $x^2+y^2=z^2$ %oppure \ (x^2+y^2=z^2\)
```

Produrrà questo risultato:

La terna pitagorica è $x^2 + y^2 = z^2$

Con il secondo ambiente, invece, la formula viene scritta in una nuova linea, centrata in mezzo alla pagina. Questo è aperto e chiuso da \$\$ oppure rispettivamente da \ [e \]. Esempio:

```
1 La terna pitagorica \ 'e $$x^2+y^2=z^2$$ %oppure \ [x^2+y^2=z^2\]
```

Produrrà questo risultato:

La terna pitagorica è

$$x^2 + y^2 = z^2$$

5.2 Apici e pedici

Spesso nelle formule (vedi gli esempi sopra) abbiamo apici e pedici. Nella modalità matematica è possibile inserirli con i caratteri ^ e _. Il testo che deve essere apice o pedice deve essere specificato tra graffe se questo è composto da più di un carattere. Esempio:

```
1 $$y^{a+b} = x_1 + x_{01}^2$$
```

Produrrà questo risultato:

$$y^{a+b} = x_1 + x_{01}^2$$

5.3 Ambienti utili

5.3.1 Equazioni

L'ambiente specificato con `\begin{equation}...\end{equation}` permette di scrivere formule in maniera analoga a `$$`, ma, in più, numera le formule.

```
1 \begin{equation}
2   y^{a+b} = x_1 + x_{01}^2
3 \end{equation}
```

Produrrà questo risultato:

$$y^{a+b} = x_1 + x_{01}^2 \tag{5.1}$$

Se si ha inoltre la necessità di centrare le formule, invece, perché occupano più righe è possibile usare l'ambiente `align`. Esempio:

```
1 \begin{align}
2   (a+b)(a-b) &= a^2 - ab + ab + b^2 \\
3   &= a^2 - b^2
4 \end{align}
```

Produrrà questo risultato:

$$(a+b)(a-b) = a^2 - ab + ab + b^2 \tag{5.2}$$

$$= a^2 - b^2 \tag{5.3}$$

5.3.2 Definizione per casi

L'ambiente `cases` ci aiuta nel caso si voglia definire, per esempio, una funzione per casi. Esempio:

```
1 $$
2   f(x) =
3   \begin{cases}
4     1 & \text{se } x > 0 \\
5     0 & \text{se } x = 0 \\
6     -1 & \text{se } x < 0
7   \end{cases}
8 $$
```

Produrrà questo risultato:

$$f(x) = \begin{cases} 1 & \text{se } x > 0 \\ 0 & \text{se } x = 0 \\ -1 & \text{se } x < 0 \end{cases}$$

Bibliografia

Quando si scrive una tesi è importante citare le fonti da cui si sono prese le informazioni (se non altro per far vedere che non si stanno dicendo stupidaggini). Latex prevede due metodi principali per scrivere la bibliografia: uno manuale e uno automatico. Il primo è di gran lunga più semplice, il secondo è più personalizzabile e offre alcune comodità in più rispetto al primo¹.

6.1 Bibliografia manuale

In questa prima versione della bibliografia i riferimenti bibliografici devono essere specificati all'interno dell'ambiente

`\begin{thebibliography}{n}...\end{thebibliography}` dove la `n` nelle seconde parentesi graffe deve essere sostituita o dal primo numero composto solo da nove che è maggiore del numero di riferimenti (per capirci se abbiamo meno di 10 riferimenti si mette 9, se ne abbiamo da 10 fino a 99 si mette 99 e così via), nel caso in cui si utilizzino le etichette automatiche, o dall'etichetta più lunga, nel caso di etichette personalizzate. La specifica dei riferimenti funziona in modo molto simile alle liste: infatti, per ogni riferimento, si specifica un `\bibitem[etichetta_personalizzate]{identificativo}` e di seguito devono essere specificate a mano tutte le informazioni necessarie. `etichetta_personalizzata` è opzionale e serve se, appunto, si vogliono mettere delle etichette che verranno mostrate al posto del testo, `identificativo` solitamente segue la sintassi `autore:titolo` (è una *best practice*, come nel caso di `\label`). Spesso se il titolo del libro è molto lungo, però, viene abbreviato.

6.1.1 Inserire la bibliografia nel documento

Una volta fatta la bibliografia, saremmo molto felici se questa apparisse nel nostro documento e venisse anche mostrata nell'eventuale indice. Per fare ciò, *immediatamente prima* di inserire la bibliografia è necessario dare i comandi:

```
1 \cleardoublepage
2 %\phantomsection
```

¹per la bibliografia si faccia riferimento al capitolo 7.2 del libro *L'Arte di scrivere con L^AT_EX* di Lorenzo Pantieri, dove è spiegato molto bene cosa fare in caso di dubbi

```
3 \addcontentsline{toc}{chapter}{\bibname}
```

Nel caso di book o report, mentre è necessario il comando:

```
1 %\clearpage
2 %\phantomsection
3 \addcontentsline{toc}{section}{\refname}
```

Nel caso di un article. Dove²:

- `\cleardoublepage` fa cominciare la bibliografia in una pagina nuova dispari, assegnando alla voce nell'indice il numero di pagina corretto;
- `\clearpage` va dato per assicurare la corretta assegnazione del numero di pagina alla voce nell'indice solo se a fine composizione il corpo del documento terminasse esattamente a fine pagina e la bibliografia cominciasse all'inizio di una pagina nuova (in tal caso si de-commenti la riga corrispondente);
- `\phantomsection` va dato solo se è caricato anche il *package* `hyperref` (in tal caso si de-commenti la riga corrispondente);
- `chapter` e `section` indicano il livello della sezione bibliografica (un capitolo e un paragrafo, rispettivamente);
- `\bibname` e `\refname` producono nell'indice generale del documento le voci *Bibliografia* e *Riferimenti bibliografici* rispettivamente.

Esempio:

```
1 \documentclass{book}
2 \begin{document}
3 Come ricorda~\cite{adams:guida}, è sempre meglio portarsi un
  asciugamano.
4 \\
5 (oppure se preferiamo)
6 \\
7 Come ricorda~\cite{adams:guida2}, è sempre meglio portarsi un
  asciugamano.
8
9 \cleardoublepage
10 %\phantomsection
11 \addcontentsline{toc}{chapter}{\bibname}
12
13
14 \begin{thebibliography}{9}
15
16 \bibitem{adams:guida} Adams, Douglas (1979), \emph{Guida
  galattica per gli autostoppisti}.
17
18 \bibitem[guida]{adams:guida2} Adams, Douglas (1979), \emph{
  Guida galattica per gli autostoppisti}.
```

²Parte rubata appunto da *L'Arte di scrivere con L^AT_EX*

```

19
20 \end{thebibliography}
21
22
23 \end{document}

```

6.2 Bibliografia automatica

La bibliografia manuale, essendo molto semplice, ha però dei limiti come già accennato. Questi sono:

- Non viene ordinata in ordine alfabetico;
- Tale bibliografia deve essere riscritta per ogni documento;
- Se si vuole cambiarne lo stile, bisogna mettere mano ad ogni singola voce!

La versione automatica, invece, permette di creare un *database* delle opere, riutilizzabile in più documenti. Per andarla a creare si utilizzano i *packages* `biblatex` e `csquotes`, inserendo nel preambolo:

```

1 \usepackage[autostyle,italian=guillemets]{csquotes}
2 \usepackage[backend=biber]{biblatex}

```

6.2.1 Database delle opere

Per fare il *database* con tutte le opere che ci servono è necessario creare un file `.bib`, che deve essere specificato tramite il comando `\addbibresource{path_to_file.bib}` nel documento, e aggiungiamo le nostre fonti con la seguente sintassi:

```

1 @book{eco:tesi,
2   author      = {Eco, Umberto},
3   title       = {Come si fa una tesi di laurea},
4   publisher    = {Bompiani},
5   date        = {1977},
6   location    = {Milano},
7 }
8
9 @article{mori:tesi,
10  author       = {Mori, Lapo Filippo},
11  title        = {Scrivere la tesi di laurea con \LaTeXe},
12  journaltitle = {Nome del giornale},
13  number       = {3},
14  date         = {2007},
15 }
16
17 @manual{mori:poesie,
18  author       = {Mori, Lapo Filippo},
19  title        = {Scrivere poesie con \LaTeX},

```

```

20   date           = {2007},
21   url            = {http://www.guitex.org/},
22 }
23 @online{wiki:latex,
24   title          = {\LaTeX{} su Wikipedia},
25   date           = {2017},
26   url            = {http://it.wikipedia.org/wiki/LaTeX},
27   sortkey        = {wiki},
28   label          = {wiki},
29 }

```

Come possiamo notare il carattere @ ci permette di indicare il tipo di riferimento (articolo, libro, manuale, ...). Questo è seguito da un identificativo univoco (vedi la parte dedicata alla sezione 6.1) e che ogni campo deve seguire la sintassi:

```

1 <nome campo> = {<contenuto del campo>},

```

(ricordarsi la virgola finale, anche all'ultimo campo)

I campi possono essere *obbligatorie* o *opzionali* a seconda del tipo di riferimento (Quelli nell'esempio sono quasi tutti *obbligatorie*).

Le tipologie di riferimenti nell'esempio sono solamente un piccolo sottoinsieme, come i campi che possono essere specificati per ogniuna ³.

Quando si specifica l'autore di un'opera è possibile che:

1. Sia necessario specificare più di un autore;
2. L'autore abbia più cognomi o/e più nomi.

Nel primo caso la sintassi da adottare è:

```

1 author = {CognomeAutore1, NomeAutore1 and CognomeAutore2,
           NomeAutore2},

```

Cioè specificando un `and` tra gli autori. Il secondo caso, invece, richiede questa sintassi:

```

1 author = {Cognome1 Cognome2, Nome1 Nome2},

```

6.2.2 Stile di citazione

Gli stili che possono essere specificati per le citazioni sono:

- **numeric** Riferimento: numerico ([1], [2], ...);
- **alphabetic** Riferimento: misto (es. [Mor07]);
- **authoryear** Riferimento: autore, anno (es. [Mori,2007]);
- **authortitle** Riferimento: autore, titolo (es. Mori,Titolo).

³per un elenco completo consultare al capitolo 7.2 del libro *L'Arte di scrivere con L^AT_EX*

Per scegliere lo stile di citazione è necessario specificarlo quando si importa il pacchetto `biblatex` in questo modo:

```
1 \usepackage[backend=biber, style=<stile scelto>,hyperref,
   backref]{biblatex}
```

(È possibile eliminare `hyperref` e `backref`, servono solamente a fare i collegamenti tra la citazione e il riferimento bibliografico).

6.2.3 Inserire la bibliografia nel documento

Per inserire la bibliografia nel documento è necessario dare i comandi:

```
1 % vedi bibliografia manuale per il significato
2 \cleardoublepage
3 % per inserire anche i riferimenti senza citazioni
4 \nocite{*}
5 % per stampare effettivamente la bibliografia
6 \printbibliography
```


Creare comandi personalizzati

Vi ricordate quando parlavamo delle email nella sezione 4.8.2.1? Ecco è arrivato il momento di capire come si definiscono i comandi personalizzati. Questa possibilità è molto utile per esempio quando abbiamo delle cose ripetitive da fare (immaginate per esempio che nella vostra tesi dobbiate scrivere sempre una parola in rosso). I comandi personali vengono definiti nel preambolo del documento e, per definire un nuovo comando, è necessario seguire la seguente sintassi:

`\newcommand{nome_comando}[argomenti]{definizione}`

Lascio a voi immaginare cosa sia `\nome_comando` e mi concentro sulle altre due parti:

- **argomenti** è il numero di “cose” che vengono passate al comando (come quello che si passa tra parentesi graffe nei normali comandi), il massimo è **9** (se non viene messo, poichè opzionale, si assume pari a 0);
- **definizione** è quello che il nostro comando deve fare. Gli argomenti possono essere sfruttati all’interno del comando tramite `#n` dove `n` è un numero tra 1 e 9, che indica l’argomento appunto.

Esempio di un comando senza argomenti:

```
1 \newcommand{\rosso}{
2   \textcolor{red}{rosso}
3 }
```

Esempio di un comando con tre argomenti:

```
1 \newcommand{\trecolori}[3]{
2   \textcolor{red}{#1}
3   \textcolor{green}{#2}
4   \textcolor{blue}{#3}
5 }
```


Creare presentazioni: Beamer

Si, avete letto bene, non vi stiamo prendendo in giro: con \LaTeX è possibile creare anche delle splendide presentazioni. “Come si fa?” immagino vi starete chiedendo, e la risposta è usando un formato di documento (`\documentclass`) apposito, che si chiama `beamer`. In questo capitolo andremo ad analizzare un attimo il funzionamento di questa classe, che include dei comandi specifici per la creazione di slide e l’inserimento di effetti di transizione tra di esse¹.

Esempio Vediamo subito un minuscolo esempio che creerà una slide contenente un semplice “Ciao!”:

```

1 \documentclass{beamer}
2
3 \begin{document}
4 % Questo comando segnala l'inizio di una slide
5 \begin{frame}
6 % Inseriamo il contenuto che vogliamo
7 Ciao!
8 \end{frame} % Fine slide
9 \end{document}

```

Il risultato che otteniamo non è dei più belli. Un punto molto forte di Beamer però è la sua praticamente infinita possibilità di personalizzazione, con supporto a temi e a differenti set di colori per ogni tema. Più avanti vedremo come selezionare alcuni già esistenti.

8.1 I comandi di base

8.1.1 Creare e dare titoli a frame

Beamer, essendo solo uno stile di documento per \LaTeX , eredita tutti i comandi dello stesso, aggiungendone di nuovi o migliorando quelli che esistono già. Un comando nuovo che abbiamo visto subito nell’esempio che abbiamo fatto all’inizio del capitolo

¹ \LaTeX è in grado di supportare transizioni anche nei PDF, dando la possibilità di avere un documento leggibile con un comune lettore di PDF ma contenente anche effetti, a differenza di PowerPoint per esempio.



Figura 8.1: Una presentazione in L^AT_EX personalizzata: è possibile notare come si possa arrivare ad ottenere un alto livello di personalizzazione tramite i temi

è il `\begin{frame} ... \end{frame}`, che ci permette di creare una nuova slide. Questo comando è ovviamente fondamentale: senza di esso non possiamo fare alcuna presentazione. Esiste la possibilità di dare un titolo alla slide che stiamo creando, e questo titolo si va a dare grazie al comando `\frametitle{ }`, dove all'interno delle parentesi graffe andremo ad inserire il testo che vogliamo. Quindi se vogliamo dare un titolo alla nostra slide di saluto potremmo modificare la nostra già esistente presentazione arricchendola nella maniera seguente:

```

1 \documentclass{beamer}
2
3 \begin{document}
4 % Questo comando segnala l'inizio di una slide
5 \begin{frame}
6 % Titolo della slide
7 \frametitle{Uno splendido saluto}
8 % Inseriamo il contenuto che vogliamo
9 Ciao!
10 \end{frame} % Fine slide
11 \end{document}
```

Se state provando a compilare gli esempi presenti nella guida (cosa che dovrete!) potrete notare cosa in alto a sinistra è successo: una scritta più grande con colori diversi dal resto del testo è apparsa.

8.1.2 Creazione slide iniziale

Tutte le presentazioni iniziano con una slide iniziale: di solito si mette il titolo dell'argomento di cui si parlerà seguito da un sottotitolo che introdurrà l'ascoltatore al tipo di discussione a cui sta partecipando. Infine l'autore, l'istituzione (se sta

presentando per qualche istituzione) e la data della presentazione stessa dovrebbero esserci sempre. Figura 8.1 è un esempio di una classica slide iniziale.

Esempio Vediamo ora come fare dal punto di vista pratico:

```
1 % Usiamo la classe per le presentazioni
2 \documentclass{beamer}
3
4 \usepackage[utf8]{inputenc} % Per gli accenti
5
6 % Inseriamo il titolo
7 \title{Ciao!}
8 % Inserimento del sottotitolo
9 \subtitle{Come salutare con stile}
10 % La data può essere espressa come ci piace
11 \date{31/07/2017}
12 % Autore della presentazione
13 \author{Davide Polonio}
14 % Istituzione per cui presentiamo, facoltativa
15 \institute{Università degli studi di Padova}
16
17 \begin{document}
18   \begin{frame}
19     % Stampa a titolo automaticamente formattato
20     \maketitle
21   \end{frame}
22
23 \end{document}
```

Analisi dei comandi Vediamo un attimo i comandi che sono presenti nell'esempio appena esposto:

- `\title{ }` ci permette, senza sorprese, di inserire il titolo del documento
- `\subtitle{ }` anche qui c'è poco da spiegare: questo comando inserisce il sottotitolo al documento
- `\date{ }` fornisce la funzionalità per inserire la data nella presentazione: è importante notare come la data può essere del formato che più ci piace, come ad esempio *1/11/1111* oppure *1 novembre 1111*.²
- `\institute{ }` permette di inserire l'istituzione per cui presentiamo (o andiamo a presentare)

In generale, tutti i campi qui presenti sono facoltativi: possiamo anche non mettere il titolo, ma con ovvie conseguenze. Inoltre questi campi oltre a servire per creare la prima pagine del documento possono contribuire a fornire *metadata*³

²Nonostante la data molto *cool*, si ipotizza che a quell'epoca le presentazioni al proiettore non andassero per la maggiore.

³<https://it.wikipedia.org/wiki/Metadato>

al PDF che verrà poi generato e per far sì che ciò avvenga basterà importare il pacchetto `hyperref` con l'opzione `pdfusetitle`.

8.2 Temi

Applicare un tema alle slide è una buona pratica per creare una presentazione più piacevole agli ascoltatori, ed è quindi un modo per tenerli in qualche modo anche attenti a quello che state dicendo⁴. Per applicare un tema esistono delle *keyword* che devono essere utilizzate. L'applicazione di un tema si suddivide in due fasi: selezione del tema stesso e, se disponibili, selezione di un set di colori possibili per quel tema. I comandi sono:

- `\usetheme{ }` usato per selezionare il tema
- `\usecolortheme{ }` usato per selezionare il set di colori per il tema

8.2.1 Temi predefiniti

I temi predefiniti sono comodi quando non si ha troppa voglia di farsene uno, o quando semplicemente quello che cerchiamo è già pronto. I temi predefiniti che Beamer offre sono pensati per essere semplici e diretti: non troveremo quindi stili all'ultimo grido, ma grafici semplici tra cui potremmo scegliere diversi abbinamenti di colore. Siccome elencare tutti i temi in questo documento non sarebbe fattibile, invitiamo il nostro affezionato utente ad aprire il proprio browser preferito ed a puntarlo a questo indirizzo web http://deic.uab.es/~iblanes/beamer_gallery/, dove i temi possono essere esplorati secondo diverse modalità (per tema, per colore oppure per entrambe le categorie). Proviamo ora ad applicare un tema alla nostra prima pagina, e vediamo che succede!

```

1 % Usiamo la classe per le presentazioni
2 \documentclass{beamer}
3
4 \usepackage[utf8]{inputenc} % Per gli accenti
5
6 % Inseriamo il titolo
7 \title{Ciao!}
8 % Inserimento del sottotitolo
9 \subtitle{Come salutare con ancora più stile}
10 % La data può essere espressa come ci piace
11 \date{31/07/2017}
12 % Autore della presentazione
13 \author{Davide Polonio}
14 % Istituzione per cui presentiamo, facoltativa
15 \institute{Università degli studi di Padova}
16
17 % Carichiamo il tema Berlin
18 \usetheme{Berlin}

```

⁴A meno che quello che state dicendo non sia terribilmente noioso.

```

19 % Settiamo il set di colori beaver
20 \usecolortheme{beaver}
21
22 \begin{document}
23   \begin{frame}
24     % Stampa a titolo automaticamente formattato
25     \maketitle
26   \end{frame}
27
28 \end{document}

```

Come possiamo vedere, le nostre slide cominciano a farsi fare sempre più carine. Provate pure a cambiare i set di colori e il tema a vostro piacimento, e potrete notare come non dovreste cambiare il contenuto: infatti in \LaTeX i temi non influenzano sulla sostanza della presentazione, come invece potrebbe accadere in una presentazione PowerPoint per esempio: questo ci permette di poter cominciare a scrivere contenuto anche se non abbiamo ancora deciso che aspetto dovrà avere.

8.3 Transizioni ed effetti

Una cosa utile durante una presentazione potrebbe essere nascondere parte del testo all'ascoltatore, in modo da presentare il contenuto di una slide in maniera controllata, permettendogli di assorbire tutto quello che viene presentato. Questo obiettivo può essere ottenuto in diverse maniere: si potrebbero copia-incollare diversi *frame* e aggiungere la parte nuova ogni volta, ma non è una soluzione pratica. \LaTeX infatti fornisce la possibilità di inserire le transizioni. Noi vedremo come applicarle sugli elenchi puntati.

8.3.1 Transizioni per gli elenchi puntati

Come abbiamo già detto, mostrare gradualmente un elenco potrebbe essere utile. Ottenere ciò è super semplice, e si fa ponendo dopo `\item` il comando `<n-m>` dove *n* e *m* rispettivamente indicano quando deve apparire quel punto nella slide e quando deve sparire (nota: *m* è opzionale e può essere omissso).

```

1 % Usiamo la classe per le presentazioni
2 \documentclass{beamer}
3
4 \usepackage[utf8]{inputenc} % Per gli accenti
5
6 % Inseriamo il titolo
7 \title{Ciao!}
8 % Inserimento del sottotitolo
9 \subtitle{Come salutare con ancora più stile}
10 % La data può essere espressa come ci piace
11 \date{31/07/2017}
12 % Autore della presentazione
13 \author{Davide Polonio}

```

```

14 % Istituzione per cui presentiamo, facoltativa
15 \institute{Università degli studi di Padova}
16
17 % Carichiamo il tema Berlin
18 \usetheme{Berlin}
19 % Settiamo il set di colori beaver
20 \usecolortheme{beaver}
21
22 \begin{document}
23   \begin{frame}
24     % Stampa a titolo automaticamente formattato
25     \maketitle
26   \end{frame}
27
28   % Iniziamo la sezione su come salutare
29   \section{Salutare}
30   \begin{frame} % Nuova slide
31     \frametitle{Come salutare} % Titolo slide
32     Step principali:
33     \begin{enumerate} % Elenco puntato
34       % Prima di ogni punto mettiamo quando lo
35       % vogliamo rendere disponibile
36       \item<1-> Alzare la mano
37       \item<2-> Rivolgerla all'interessato
38       \item<3-> Muoverla a destra e a sinistra
39       \item<4-> Fatto!
40     \end{enumerate}
41   \end{frame}
42 \end{document}

```

Se ora apriamo il PDF e lo mettiamo in modalità presentazione vedremo come \LaTeX ha creato delle slide che scorrendo danno il senso di transizioni, permettendoci di far visualizzare il testo che ci interessa in determinati momenti. Questi effetti di transizione si possono usare sia con gli elenchi puntati sia con quelli numerati, il comportamento è indifferente.

8.3.2 Effetti

Qualcuno potrebbe lamentarsi che qui non sono presenti gli effetti sulle transizioni dei lucidi come in PowerPoint. Chi sostiene questo evidentemente non conosce a sufficienza *Beamer*, infatti, è possibile inserire tra una slide e l'altra la transizione che più ci piace. Gli effetti sono elencati nella Tabella 8.1.

È possibile far avanzare automaticamente le slide o gli elenchi puntati tramite la *keyword* `\transduration{ }` dove nelle parentesi graffe andremo a inserire in secondi la pausa tra l'avanzamento di una slide e l'altra. Questo può essere molto comodo e si può combinare con gli effetti di transizioni tra slide e tra elenchi puntati che abbiamo avuto modo di vedere prima.

Riprendendo in mano la presentazione su come salutare, andiamo ad aggiungere l'avanzamento automatico e un effetto di transizione a nostro piacimento tra quelli

Keyword	Effetto
<code>\transblindshorizontal</code>	La slide successiva scorre sopra quella precedente con un movimento da sinistra verso destra
<code>\transblindsvertical</code>	La slide successiva scorre sopra quella precedente con un movimento dall'alto verso il basso
<code>\transboxin</code>	La slide precedente viene “assorbita” in un quadrato da quella nuova
<code>\transboxout</code>	La slide successiva si espande a forma di quadrato su quella precedente
<code>\transdissolve</code>	La slide successiva si dissolve in un effetto “pixelato”
<code>\transglitter</code>	Un misto tra una dissolvenza a pixel e una transizione orizzontale
<code>\transwipe</code>	La slide successiva esegue uno “swipe” sopra quella precedente da sinistra verso destra
<code>\transfade</code>	Effetto di fade tra la slide precedente e quella successiva

Tabella 8.1: Tabella degli effetti per Beamer. Nota: solo uno di questi effetti può essere applicato per pagina: l'applicazione di due o più effetti nella stessa slide causerà un errore durante la compilazione.

disponibili:

```

1  % Usiamo la classe per le presentazioni
2  \documentclass{beamer}
3
4  \usepackage[utf8]{inputenc} % Per gli accenti
5
6  % Inseriamo il titolo
7  \title{Ciao!}
8  % Inserimento del sottotitolo
9  \subtitle{Come salutare con ancora più stile}
10 % La data può essere espressa come ci piace
11 \date{31/07/2017}
12 % Autore della presentazione
13 \author{Davide Polonio}
14 % Istituzione per cui presentiamo, facoltativa
15 \institute{Università degli studi di Padova}
16
17 % Carichiamo il tema Berlin
18 \usetheme{Berlin}
19 % Settiamo il set di colori beaver
20 \usecolortheme{beaver}
21
22 \begin{document}
23   \begin{frame}
24     % Stampa a titolo automaticamente formattato
25     \maketitle
26   \end{frame}
27

```

```

28 % Iniziamo la sezione su come salutare
29 \section{Salutare}
30 \begin{frame} % Nuova slide
31   \frametitle{Come salutare} % Titolo slide
32
33   \transdissolve % Effetto pixel
34   \transduration{2} % Avanzamento ogni 2 secondi
35
36   Step principali:
37   \begin{enumerate} % Elenco puntato
38     % Prima di ogni punto mettiamo quando lo
39     % vogliamo rendere disponibile
40     \item<1-> Alzare la mano
41     \item<2-> Rivolgerla all'interessato
42     \item<3-> Muoverla a destra e a sinistra
43     \item<4-> Fatto!
44   \end{enumerate}
45 \end{frame}
46 \end{document}

```

Come possiamo notare alla fine con poche righe di codice (meno di 50!) siamo riusciti a fare una presentazione con effetti e uno stile suo: dopo aver fatto un po' di pratica gli autori di questa guida vi possono assicurare che la realizzazione di presentazioni tramite \LaTeX diventa più rapida di quelle tramite PowerPoint. Il solo scoglio iniziale è quello di assimilare la sintassi e di prenderci un po' la mano.

La Tesi

In questo capitolo analizzeremo come creare un proprio stile per una Tesi di laurea. È risaputo infatti come una pubblicazione di questo calibro possa essere corposa. Come accennato all’inizio della guida, \LaTeX è particolarmente adatto per documenti di certe dimensioni.

9.1 Template

\LaTeX permette sia la creazione di un proprio stile sia di utilizzarne uno già pronto. Uno dei modi per personalizzare il proprio documento sono i *template*. I template possiamo vederli come un insieme di personalizzazioni grafiche e comandi che servono per cambiare l’aspetto del nostro documento. Come più volte abbiamo ripetuto questa è solo una guida introduttiva e quindi non spiegheremo come creare il proprio template¹. Online sono disponibili molti template e conviene² usare uno di questi (eventualmente apportando delle modifiche). Trovato un template che ci piace basta utilizzare il comando `\input`, per importare il file `tex` contenente il codice del template. I template sono differenti dai temi descritti per la classe `beamer`, ma possiamo dire che hanno scopi simili. Per personalizzare i documenti, inoltre, è anche possibile creare la propria classe di documenti, ma anche questo esula dallo scopo di questa guida.

9.2 Copertina

Anche se di importante in una tesi è il contenuto, anche l’occhio vuole la sua parte. La realizzazione di una copertina non è difficile, e qui vedremo come realizzarla in pochi passi, strutturando il progetto come abbiamo visto nel capitolo apposito.

Prima di tutto, definiremo dei comandi in maniera tale da rendere la creazione della copertina più chiara e anche per separare il contenuto dalla presentazione.

Creazione delle variabili Apriamo il nostro file `res/config/config.tex` e aggiungiamo le nostre generalità sottoforma di variabili:

¹Il punto di partenza è creare un file di comandi propri in per formattare le pagine.

²Consiglio degli autori.

```

1 % Autore
2 \newcommand{\myName}{Davide Polonio}
3 % Titolo
4 \newcommand{\myTitle}{L'arte del saluto: impatto sociale su
   diverse etnie}
5 % Tipo di tesi
6 \newcommand{\myDegree}{Tesi di laurea triennale}
7 % Università
8 \newcommand{\myUni}{Università degli Studi di Padova}
9 % Facoltà
10 \newcommand{\myFaculty}{Corso di Laurea in Informatica}
11 % Dipartimento
12 \newcommand{\myDepartment}{Dipartimento di Matematica}
13 % Relatore
14 \newcommand{\myProf}{Pinco Palla}
15 % Dove
16 \newcommand{\myLocation}{Padova}
17 % Anno
18 \newcommand{\myAA}{2099-2100}
19 % Quando
20 \newcommand{\myTime}{Ott 2100}

```

Creazione del frontespizio Ora che abbiamo creato le variabili con le nostre informazioni andiamo a creare la parte grafica, ovvero quella che poi sarà generata come prima pagina. Per far ciò creiamo un file in `res/sections/Frontespizio.tex` e aggiungiamo il seguente contenuto:

```

1 % Cambiamo la numerazione delle pagine
2 \frontmatter
3 % Definiamo il frontespizio
4 \begin{titlepage}
5 % Settiamolo in posizione centrale
6 \begin{center}
7 % Posizioniamo per prima il nome della
8 % Università, in grassetto e maiuscolo
9 \begin{LARGE}
10 \textbf{\myUni}\\
11 \end{LARGE}
12
13 % Inseriamo 10pt, ovvero 10*0.0138pollici
14 \vspace{10pt}
15 % Anche qui vogliamo la scritta larga, ma
16 % senza il corsivo, definiamo il dipartimento
17 \begin{Large}
18 \textsc{\myDepartment}\\
19 \end{Large}
20 % Altro spazio verticale
21 \vspace{10pt}

```

```
22 % Iniziamo un'altra parte che sarà in
23 % maiuscoletto in cui definiamo la facoltà
24 \begin{large}
25 \textsc{\myFaculty}\\
26 \end{large}
27 % Mettiamo un grande spazio in quando ora
28 % inseriamo il logo dell'Università
29 \vspace{30pt}
30 % Posizioniamo il logo
31 \begin{figure}[htbp]
32 \begin{center}
33 \includegraphics[height=6cm]{logo-unipd}
34 \end{center}
35 \end{figure}
36 % Altro grande spazio verticale per inserire
37 % il titolo
38 \vspace{30pt}
39
40 % Titolo bello grande, centrato, in grassetto
41 \begin{LARGE}
42 \begin{center}
43 \textbf{\myTitle}\\
44 \end{center}
45 \end{LARGE}
46
47 % Ci distanziamo dal titolo
48 \vspace{10pt}
49 % Tipo di laurea
50 \begin{large}
51 \textsl{\myDegree}\\
52 \end{large}
53 % Spazio per inserire relatore/laureando
54 \vspace{40pt}
55 % Inseriamo sulla sinistra il relatore
56 \begin{large}
57 \begin{flushleft}
58 \textit{Relatore}\\
59 \vspace{5pt}
60 Prof. \myProf
61 \end{flushleft}
62 % Nella stessa linea
63 \vspace{0pt}
64 % Inseriamo il laureando
65 \begin{flushright}
66 \textit{Laureando}\\
67 \vspace{5pt}
68 \myName
69 \end{flushright}
70 \end{large}
71 % Andiamo a fondo pagina
```

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA

CORSO DI LAUREA IN INFORMATICA



L'arte del saluto: impatto sociale su
diverse etnie

Tesi di laurea triennale

Relatore

Prof. Pinco Palla

Laureando

Davide Polonio

ANNO ACCADEMICO 2099-2100

Figura 9.1: Risultato della copertina che abbiamo appena scritto

```

72 \vspace{40pt}
73 % Linea orizzontale
74 \line(1, 0){338} \\
75 % Inserimento anno accademico
76 \begin{normalsize}
77 \textsc{Anno Accademico \myAA}
78 \end{normalsize}
79
80 \end{center}
81 \end{titlepage}

```

Ora abbiamo la copertina per la nostra tesi.

9.3 Seconda pagina, ringraziamenti

Come tutte le tesi che si rispettino, è importante anche introdurre prima quello che viene definito come **colofone** (ovvero una parte in cui sono scritti i dettagli della pubblicazione) e poi una sezione di ringraziamenti. Andremo ora a vedere come si fa.

Colofone Prima di tutto, andiamo a creare un altro file, nel percorso `res/sections` e chiamiamolo `colofone`. Ricordiamoci di includere questo file aggiungendolo al

file `listOfSections.tex` che, se abbiamo seguito la guida passo passo, si trova in `res/`.

```

1 % Colofone
2 % Puliamo la pagina da tutti gli stili precedenti
3 \clearpage
4 % Creiamo una sezione speciale
5 \phantomsection
6 % Non definiamo uno stile per questa pagina
7 \thispagestyle{empty}
8
9 % Riempiamo gli spazi vuoti
10 \hfill
11
12 \vfill
13
14 % Inseriamo a fondo pagina il nostro nome
15 % seguito dal titolo in corsivo, con il tipo di laurea,
16 % il simbolo del copyright e l'anno di pubblicazione
17 \noindent\myName: \textit{\myTitle},
18 \myDegree,
19 \textcopyright\ \myTime.

```

Ringraziamenti Creiamo un altro file nel solito percorso, e lo chiameremo `ringraziamenti.tex`. Qui inseriremo la dedica che più ci piace, completata dalla nostra firma che sarà in basso a destra.

```

1 % Ringraziamenti
2 \cleardoublepage
3 % Creiamo una sezione speciale
4 \phantomsection
5 % Non vogliamo nessuno stile
6 \thispagestyle{empty}
7 % Segnamo nella lista delle sezioni dei lettori
8 % PDF questa parte col nome 'ringraziamenti'
9 \pdfbookmark{Ringraziamenti}{ringraziamenti}
10
11 %\bigskip
12
13 % Aggiugiamo impostazioni per l'impaginazione
14 \begingroup
15 \let\clearpage\relax
16 \let\cleardoublepage\relax
17 \let\cleardoublepage\relax
18
19 % Creiamo una sezione che non verrà aggiunta all'indice
20 \section*{Ringraziamenti}
21 % Ringraziamenti generici, in corsivo
22 \noindent \textit{Innanzitutto, vorrei esprimere la}
23 mia gratitudine al Prof. Pinco Palla, relatore della

```

```

24 mia tesi, per l'aiuto e il sostegno fornitomi durante la
25 stesura del lavoro.}\\
26
27 \noindent \textit{Desidero ringraziare con affetto i
28 miei genitori per il sostegno, il grande aiuto e per
29 essermi stati vicini in ogni momento durante gli anni di
30 studio.}\\
31
32 % Distanziamoci un poco dai ringraziamenti
33 \bigskip
34 % Inseriamo la firma composta da posto, data e nome
35 \noindent \textit{\myLocation, \myTime}
36 \hfill \myName
37
38 \endgroup

```

9.4 Si può fare di più?

Certo! Innanzitutto si deve inserire la bibliografia (per esempio in uno dei due modi visti in questa guida). Si possono, inoltre, aggiungere delle appendici a fine tesi (con il comando `\appendix` si segnala l'inizio della sezione dedicata alle appendici) e anche l'aggiunta di un glossario dei termini potrebbe aiutare il lettore a comprendere di più quello che è stato scritto. Queste parti possono essere approfondite rispettivamente in:

- https://en.wikibooks.org/wiki/LaTeX/Document_Structure#Book_structure
- <https://en.wikibooks.org/wiki/LaTeX/Glossary>