
I.S.I.S. Raffaele del Rosso - Giovanni da Verrazzano

Dispense LaTeX

Redatta dal prof. Sbroli Iacopo

per la classe IV scientifico

A.S. 2019-2020

Versione 1 (24/12/2019)

Indice

1	Utilizzare L^AT_EX	3
1.1	Download sul personal computer	3
1.2	Piattaforme online	4
2	Breve descrizione del programma	5
2.1	Un primo approccio a L ^A T _E X	5
2.2	Primo esercizio: Codice minimale	5
3	Pacchetti	7
3.1	Pacchetto <i>babel</i>	7
3.2	Pacchetto <i>inputenc</i>	8
3.3	Comandi	8
3.3.1	Il comando <i>emph</i>	8
3.3.2	Il comando <i>textbf</i>	8
3.3.3	Il comando <i>underline</i>	8
3.3.4	Stesura di liste	8
3.3.5	Caratteri speciali	10
4	Amsmath e graphicx: equazioni e immagini	11
4.1	Pacchetti	11
4.1.1	Pacchetto <i>amsmath</i>	11
4.1.2	L'allineamento e l' <i>e commerciale</i>	12
4.1.3	Pacchetto <i>graphicx</i>	12
5	Lipsum, geometry e riferimenti alle equazioni	13
5.1	Pacchetti	13
5.1.1	Pacchetto <i>lipsum</i>	13
5.1.2	Pacchetto <i>geometry</i>	13
5.2	Referenziare un'equazione	15
5.3	Esempi di codice sorgente e documento pdf associato	16
6	Tabelle	18
6.1	Tabelle: primo esercizio	18
6.2	Tabelle: secondo esercizio	19
6.3	Pacchetto <i>makecell</i> : terzo esercizio	19
6.4	Il comando <i>multicolumn</i>	20
7	Fancyhdr, amsmath e amssymb	21
7.1	Pacchetto <i>fancyhdr</i>	21
7.2	Pacchetto <i>amsmath</i> : l'ambiente <i>split</i>	23
7.3	Pacchetto <i>amssymb</i> : simboli	23
8	Il pacchetto xcolor	25
8.1	Impostazione <i>newcommand</i>	25
9	Frontespizi e liste personalizzate	26
9.1	Creazione di frontespizi personalizzate	26
9.2	Inserzione di un'immagine per lo sfondo	27
9.3	Inserzione di liste personalizzate	28

10 Creare una bibliografia su L^AT_EX	29
10.1 Il concetto di bibliografia	29
10.2 Il file Bibliography.bib	29
10.3 Inserzione della bibliografia nel vostro documento L ^A T _E X	30
10.4 Citazione di un elemento della bibliografia	30
10.4.1 Compito del 30/11/2019	31
10.4.2 Compito del 7/12/2019-21/12/2019	31
Glossario	32

Sezione 1

Utilizzare L^AT_EX

1.1 Download sul personal computer

Per installare L^AT_EX è necessario scaricare **due programmi**:

1. Un **editor di testo**, ovvero un'applicazione tramite cui creare semplici documenti con i caratteri del gruppo ASCII esteso¹. Il *blocco note* è un esempio di editor di testo. **Esistono alcuni editor di testo specifici per L^AT_EX**, che aiutano il redattore tramite scorciatoie, wizard di supporto e pulsanti.
2. Un **gestore dei pacchetti e delle applicazioni**. Si tratta del cuore di L^AT_EX: è il programma che traduce il file semplice scritto con caratteri ASCII in un file PDF. Nel programma sono contenuti una serie di compilatori, i quali (vi ricordo) trasformano il codice sorgente in linguaggio macchina, che può essere letto ed eseguito dal computer. I compilatori effettuano la trasformazione da file di testo a file pdf. Quello che utilizzerete voi si chiama PDFLaTeX.

Una combinazione possibile di **editor e gestori dei pacchetti** si ottiene scaricando T_EX maker (editor) e T_EXlive (gestore pacchetti: Windows e Linux) oppure scaricando T_EX maker (editor) e MacT_EX (gestore: MacOS). Per farlo, seguite la procedura sottostante:

1. Windows:
 - (a) **Scaricate l'installer di T_EXlive al seguente link:**
<http://mirror.ctan.org/systems/texlive/tlnet/install-tl-windows.exe>
 - (b) Aprite l'installer e selezionate l'opzione *custom install* (installazione personalizzata)
 - (c) Alla schermata intitolata «installazione di T_EXlive 2018», selezionate il pulsante «cambia» accanto a «schemi selezionati»
 - (d) Selezionate lo «schema full» (se già non è stata spuntata questa opzione) e procedete all'installazione
 - (e) L'installazione potrà durare un paio d'ore, dato l'elevato numero di pacchetti opzionali che scaricherete con lo «schema full».
 - (f) **Scaricare l'editor T_EX maker che trovate al seguente link:**

<http://www.xmlmath.net/texmaker/download.html>

2. MacOS:

- (a) **Scaricate l'installer di MacT_EX al seguente link:**

<http://www.tug.org/mactex/>

Linux: aprite il terminal e digitate il seguente comando:

```
sudo apt-get install texlive-full
```

- (b) Aprite il file, rispondete affermativamente a tutte le domande e installate il tutto.

¹Sono i 256 caratteri standard, tra cui sono comprese le lettere dell'alfabeto, alcune lettere accentate, simboli quali le parentesi tonde, quadre e graffe, l'apostrofo, la chiocciola, e così via.

- (c) Scaricare l'editor $\text{T}_{\text{E}}\text{X}$ maker che trovate al seguente link:

<http://www.xm1math.net/texmaker/download.html>

3. Linux:

- (a) Scaricate $\text{T}_{\text{E}}\text{X}$ live scrivendo il seguente comando nel prompt:

```
sudo apt-get install texlive-full
```

- (b) Scaricate l'editor $\text{T}_{\text{E}}\text{X}$ maker scrivendo il seguente comando nel prompt:

```
sudo apt-get install texmaker
```

1.2 Piattaforme online

Se non si desidera sprecare tempo per effettuare l'installazione di $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ sul proprio calcolatore, è possibile utilizzare una piattaforma online. La piattaforma in questione è **Overleaf**.

Per accedere a Overleaf è sufficiente cliccare qui: <http://www.overleaf.com/>. Crearsi un profilo è immediato: basta scegliere un nome utente e una password. Overleaf è una piattaforma molto comoda per la tipocomposizione in quanto sono disponibili numerosi template e l'editor offre un notevole supporto all'utente tramite opzioni di completamento automatico.

Sezione 2

Breve descrizione del programma

L^AT_EX (pronuncia: làtec) è sostanzialmente **un software per la tipocomposizione**. Tramite L^AT_EX è possibile costruire testi di alta qualità (relazioni, libri, articoli di giornale, articoli scientifici, tesi di laurea). L^AT_EX possiede alcuni strumenti estremamente potenti, come il file di stile *tikz*, tramite il quale è possibile costruire con facilità immagini e grafici di elevata qualità.

L^AT_EX si distingue sostanzialmente da altri programmi per la tipocomposizione perché esso separa la *strutturazione* e la *presentazione del documento*. Infatti, per scrivere un testo tramite L^AT_EX, è necessario prima costruire un *codice* tramite un editor di testo e poi *tradurre* il codice in un file pdf tramite il *compilatore* PDFL^AT_EX.

2.1 Un primo approccio a L^AT_EX

In questa esercitazione userete per la prima volta L^AT_EX tramite la *distribuzione* T_EXLive e il relativo *editor di testo* T_EXmaker, che possono essere facilmente scaricati dal web (si tratta di *freeware*: il programma è gratuito e può essere liberamente modificato). Ricordate che per lavorare con L^AT_EX avete sempre bisogno di un *editor di testo*, ovvero di un programma che vi permetta di scrivere ciò che pensate, e di un *compilatore*, ovvero di un traduttore che possa interpretare il vostro testo e tradurlo in un file pdf. Come è già stato accennato, tramite L^AT_EX è possibile separare la *strutturazione* del testo dalla sua *presentazione*. La *strutturazione* è prodotta dallo scrittore tramite l'editor, mentre la *presentazione* può essere creata cliccando sul pulsante a forma di freccia in alto a sinistra. La *presentazione* consiste in un singolo file pdf.

2.2 Primo esercizio: Codice minimale

Per abituarsi a L^AT_EX è sicuramente opportuno cominciare costruendo un codice sorgente che contenga il minor numero di comandi possibile (un *codice minimale*).

Tutti i codici in L^AT_EX devono contenere i tre seguenti comandi:

```
\documentclass[dimensione caratteri,altre opzioni]{classi di documento}
\begin{document}
\end{document}
```

Il parametro «dimensione caratteri» può essere uguale solo a 10pt, 11pt e 12pt. **È possibile definire altre dimensioni dei caratteri**, ma non lo faremo oggi. In L^AT_EX, tutti i comandi che precedono «\begin{document}» sono detti collettivamente *preambolo*. Nel preambolo vengono inserite tutte le opzioni che si desidera applicare all'intero documento. Nel nostro caso, l'unico comando presente nel preambolo è il seguente:

```
documentclass[dimensione caratteri,altre opzioni]{classi di documento}
```

Come è naturale, l'impostazione della classe del documento si applica all'intero testo: viene quindi inserita nel preambolo.

¹ **Tenendo conto del fatto che i parametri tra parentesi quadre sono opzionali**, per creare un codice minimale è sufficiente scegliere la classe del documento e poi cliccare sul pulsante verde, inserendo ovviamente del testo tra i comandi \begin{document} e \end{document}.

¹Esistono una decina di *classi di documento*, tra cui *book* (libro), *article* (articolo), *letter* (lettera), *beamer* (presentazioni in stile powerpoint). Ciascuna classe permette l'uso di determinati comandi per comporre il testo nel miglior modo possibile. Ad esempio, la classe *book* dispone del comando *chapter* per creare capitoli, la classe *article* dispone del comando *abstract* per creare il riassunto dell'articolo, la classe *letter* dispone del comando *address* per scrivere l'indirizzo del mittente, e così via.

Scrivete il vostro primo codice minimale e componete il testo cliccando sulla freccia in alto a sinistra. Utilizzare i seguenti comandi:

1. `\chapter{nome capitolo}` (crea un capitolo: solo per i libri)
2. `\section{nome sezione}` (crea una sezione)
3. `\\` (va a capo)

Esercizio 2: Formule matematiche

Creare formule matematiche con \LaTeX è molto semplice. Per farlo, è necessario prima di tutto definire un *ambiente matematico* tramite due comandi (*apertura* e *chiusura* dell'ambiente). Alcuni esempi sono i seguenti:

1. `\$x^2+1\$` (la formula compare all'interno del testo)
2. `\[x^2 +1 \]` (la formula compare in una riga separata)
3. `\begin{equation} x^2 +1 \end{equation}` (la formula compare in una riga separata ed è numerata)

Esistono molti altri comandi per aprire e chiudere l'ambiente matematico, ma non li vedremo oggi.

Provate a comporre una formula utilizzando tutti e tre i metodi e almeno 6 dei seguenti comandi:

1. `A^B` (elevamento a potenza)
2. `A_B` (pedice)
3. `\pi` (pi greco)
4. `\sqrt{A}` (radice quadrata)
5. `\frac{A}{B}` (rapporto)
6. `\pm` (più o meno)
7. `\lim_{A\to B}` (limite)
8. `\sum_{i=A}^B` (sommatoria)

Qui sotto sono presentati alcuni esempi di formule composte tramite \LaTeX :

1. x^2
2. x_2
3. $A_{cerchio} = \pi r^2$
4. $\sqrt{x^3 + 1}$
5. $\frac{3}{4} = 0,75$
6. $x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$
7. $\lim_{x \rightarrow 0} \frac{1}{x} = \infty$
8. $\sum_{i=1}^{\infty} \frac{1}{i^2} = \frac{\pi^2}{6}$

Come potete osservare, **le formule costruite tramite \LaTeX sono di qualità molto elevata**. Tramite il codice **è anche relativamente semplice costruirle**, una volta che si acquisisce una certa dimestichezza.

Sezione 3

Pacchetti

I «pacchetti» (in gergo tecnico: *file di stile*) sono dei file di supporto. Includendoli all'interno di un documento tipocomposto con \LaTeX è possibile usufruire di numerose opzioni:

1. Costruzione di liste personalizzate particolarmente sofisticate.
2. Inserzione di immagini, anche nello sfondo.
3. Costruzione di grafici complessi.
4. Colorazione del testo.
5. Traslitterazione dei testi in altre lingue.
6. Inclusione di caratteri speciali.
7. Alterazione della geometria del documento (margini, larghezza, lunghezza).
8. Inclusione di particolari impaginazioni per le equazioni.
9. Inclusione di intestazioni.
10. Ottimizzazione

Per includere un pacchetto in un documento \LaTeX è necessario inserire la seguente riga nel preambolo:

```
\usepackage {«nome pacchetto»}
```

Dove al posto di «nome pacchetto» è necessario inserire il nome del pacchetto che si desidera utilizzare. Facciamo un esempio. Il file di stile per modificare la geometria del documento si chiama *geometry*. Per includerlo scriveremo:

```
\usepackage {geometry}
```

3.1 Pacchetto *babel*

Non tutti i comandi di \LaTeX sono indipendenti dalla lingua in cui è scritto il documento. Il comando `\today`, che si utilizza per stampare la data corrente, è un esempio. **Se non inserite il pacchetto *babel* nel preambolo** e provate a compilare un codice che contiene tale comando, visualizzerete la data in questo modo:

25 dicembre 2019

Mentre il risultato che desiderate è il seguente:

25 dicembre 2019

Per far sì che ciò è avvenga, è sufficiente scrivere **nel preambolo** la seguente riga:

```
\usepackage[italian]{babel}
```

Il pacchetto *babel* può essere utilizzato anche per passare da una lingua all'altra all'interno del testo. Per farlo, occorrono due passaggi. Il primo è indicare a *babel* che si desidera utilizzare due lingue, come nell'esempio sottostante:


```
\usepackage[italian,spanish]{babel}
```

Così facendo, \LaTeX utilizzerà due lingue in fase di compilazione (la principale è l'ultima elencata: lo **spagnolo** in questo caso). Tutti i comandi del vostro codice sorgente saranno compilati con il pacchetto spagnolo, **a meno che non delimitiate una parte di testo con il comando $\text{\selectlanguage\{italian\}}$** (secondo passaggio), come nell'esempio sottostante:

```
{ \selectlanguage\{italian\} \today }
```

La data comparirà in italiano, nonostante la lingua principale sia lo spagnolo.

3.2 Pacchetto *inputenc*

Il pacchetto *inputenc* è essenziale per far sì che \LaTeX interpreti correttamente il vostro codice sorgente. Infatti, se non fornite specificazioni particolari, \LaTeX lo interpreterà utilizzando i 128 caratteri ASCII originari, i quali non includono le lettere accentate, le lettere con dieresi e i caratteri di svariate lingue che non sfruttano l'alfabeto latino.

Per far sì che questi problemi non sussistano, è sufficiente scrivere nel preambolo la seguente riga:

```
\usepackage[utf8]{inputenc}
```

Così facendo, tutti i caratteri che inserite saranno correttamente riprodotti sul documento pdf finale. L'opzione *utf8* del pacchetto *inputenc* sta a significare *unicode transformation format a 8 bit*, un sistema di codifica molto ampio che comprende più di 100.000 caratteri.

3.3 Comandi

3.3.1 Il comando *emph*

Il comando *emph* può essere utilizzato per scrivere in corsivo una parte di testo, allo scopo di metterla in evidenza. Se ad esempio scrivete:

```
Voglio mettere in \emph\{risalto\} la mia intelligenza
```

La parola *risalto* risulterà scritta in corsivo. **Il comando *emph* è pensato anche per mettere in evidenza una parola o una parte di testo in un paragrafo scritto interamente in corsivo.** Provate a verificarlo autonomamente.

3.3.2 Il comando *textbf*

Il comando *textbf* serve a evidenziare in neretto una parte di testo. La sintassi di tale comando è identica a quella di *emph*. Provate a utilizzarlo.

3.3.3 Il comando *underline*

Il comando *underline* serve a sottolineare una parte di testo. La sintassi di tale comando è identica a quella di *emph* e *textbf*. Provate a utilizzare anche questo.

3.3.4 Stesura di liste

Tramite \LaTeX è possibile stilare agevolmente delle liste. Per farlo, potete utilizzare due sintassi. Tramite la prima potete redigere elenchi puntati:

```
\begin\{itemize\}
\item Elemento
\item Elemento
..
\end\{itemize\}
```

Tramite la seconda potete redigere elenchi numerati:

```
\begin\{enumerate\}
\item Elemento
\item Elemento
```

```
..  
\end{enumerate}
```

Indicando a \LaTeX di utilizzare il pacchetto *enumitem* nel preambolo, gli elenchi numerati potranno essere redatti utilizzando i numeri romani o le lettere latine come ordinali. Per utilizzare questi o altri ordinali, dovrete scrivere:

```
\begin{enumerate}[label=«il vostro ordinale»]
```

Esempi di ordinali utilizzabili dal pacchetto *enumitem* sono `\alph*` (alfabeto latino) e `\roman*` (numeri romani). Un esempio di comando sarà quindi:

```
\begin{enumerate}[label=\roman*]
```

3.3.5 Caratteri speciali

In questa sezione troverete un esempio di codice sorgente prodotto dal docente. Analizzando questo esempio sarete facilitati nella stesura di un codice sorgente in \LaTeX .

Fate attenzione a collocare i comandi del preambolo prima di $\text{\begin{document}}$. Tra questi comandi rientrano tutti gli \usepackage , i parametri del titolo, e in generale tutte le definizioni che valgono nell'intero documento.

È importante sottolineare che alcuni dei caratteri che digitate sono speciali per \LaTeX . Tra questi caratteri speciali rientrano il backslash « \ », il simbolo del dollaro « $\text{\$}$ », la parentesi graffa « $\text{\{}$ », l'underscore « _ » e molti altri. Questi simboli sono utilizzati da \LaTeX per individuare i comandi da voi inseriti: il backslash e la parentesi graffa sono utilizzati nel comando $\text{\begin{document}}$, mentre il simbolo del dollaro e l'underscore sono utilizzati negli ambienti matematici. **Se volete che questi simboli compaiano all'interno del testo nel pdf finale dovrete utilizzare dei comandi appositi, ovvero:**

1. $\text{\textbackslashtextbackslash}$ per il backslash
2. $\text{\textbackslash\$}$ per il simbolo del dollaro.
3. $\text{\textbackslash\{}$ per la parentesi graffa.
4. \textbackslash_ per l'underscore.
5. \textbackslash\& per l'ampersand.
6. \textbackslash\% per il simbolo di percentuale.

```

##### INIZIO PREAMBOLO #####

\documentclass{book} %% PRIMO COMANDO in assoluto (essenziale): definisco la classe di documento

\usepackage[utf8]{inputenc} %% Utilizzo un pacchetto (opzionale): definisco l'insieme di caratteri
letti da LaTeX

\usepackage[italian]{babel} %% Utilizzo un pacchetto (opzionale): chiedo a LaTeX di utilizzare
l'italiano come lingua di base

\usepackage{enumitem} %% Utilizzo un pacchetto (opzionale): utile per definire liste con vari
ordinali (alfabeto latino, numeri romani)

##### FINE PREAMBOLO #####
\begin{document} %% Inizio del documento (obbligatorio)
##### INIZIO DOCUMENTO (tutto ciò che scrivete va qui sotto) #####

\maketitle %% Creo il frontespizio del libro (opzionale)

\chapter{L'invasione dei Normanni} %% Definisco il nome di un capitolo (opzionale)
\begin{enumerate}[label=\alph*] %% Faccio un elenco alfabetico (opzionale)
\item Barbari %% Elemento della lista (obbligatorio: è il primo)
\item Berberi %% Elemento della lista (opzionale)
\end{enumerate} %% Termino l'elenco alfabetico
\end{document} %% Fine del documento (obbligatorio)
##### FINE DOCUMENTO #####

```

Figura 3.1: Codice sorgente di un documento \LaTeX . Si noti la presenza di soli 3 comandi obbligatori ($\text{\documentclass{«classe»}}$, $\text{\begin{document}}$ e $\text{\end{document}}$). Sono stati utilizzati soltanto i pacchetti che avete visto in laboratorio.

Sezione 4

Lipsum, geometry e riferimenti alle equazioni

Dando per assodata l'assimilazione dei concetti esplorati nei precedenti capitoli, nella presente sezione saranno trattati i seguenti argomenti:

1. Uso dei pacchetti *lipsum* e *geometry* .
2. Referenziazione di un'equazione.

Tramite le nozioni che apprenderete oggi, potrete regolare le dimensioni del foglio e del testo al suo interno. Sarete inoltre in grado di citare all'interno del testo un'equazione che avete scritto.

4.1 Pacchetti

4.1.1 Pacchetto *lipsum*

Il pacchetto lipsum viene utilizzato per creare automaticamente un lungo testo. Tale testo è del tutto privo di significato, ma è utile in quanto permette al redattore del documento di valutare se la geometria del foglio è ideale oppure no. Per utilizzare il pacchetto lipsum sarà sufficiente scrivere `\usepackage{lipsum}` nel preambolo e poi il comando `\lipsum` nel corpo del testo (dopo `\begin{document}`).

Esercizio 1: riempite una frazione di testo con il pacchetto lipsum (un capitolo). Fate gli esercizi successivi nello stesso documento, ma in capitoli diversi. Utilizzate la lingua italiana tramite il pacchetto babel descritto nella dispensa precedente.

4.1.2 Pacchetto *geometry*

Talvolta, la geometria del documento standard offerta da L^AT_EX è insoddisfacente. Ci farebbe piacere avere un foglio più grande, oppure ci piacerebbe poter sfruttare una porzione di foglio maggiore. Vorremmo che il margine sinistro del corpo del testo fosse più vicino al margine sinistro del foglio.

Per soddisfare questi bisogni, è possibile utilizzare il pacchetto *geometry*. Le opzioni del pacchetto sono molteplici. Oggi ne esploreremo solo alcune, ovvero:

- (a) Paperwidth (larghezza dell'intero foglio) (standard a4: 597pt)
- (b) Paperheight (altezza dell'intero foglio) (standard a4: 845pt)
- (c) Textwidth (larghezza del corpo del testo) (standard a4: 426pt)
- (d) Textheight (altezza del corpo del testo) (standard a4: 724pt)
- (e) Marginparwidth (larghezza della sezione per le note a margine) (standard: 50pt)
- (f) Footskip (altezza del margine che separa la fine del corpo del testo e la fine della sezione per le note a margine) (standard a4: 30pt)

Ciascuna di queste opzioni può essere utilizzata utilizzando la seguente sintassi:

```
\usepackage[opzione1=dimensioni1,opzione2=dimensioni2,...]{geometry}
```

Esempio:

```
\usepackage[paperwidth=300pt,paperheight=600pt]{geometry}
```

Inserire sempre le **minuscole** nelle opzioni (`paperheight` e `non Paperheight`).

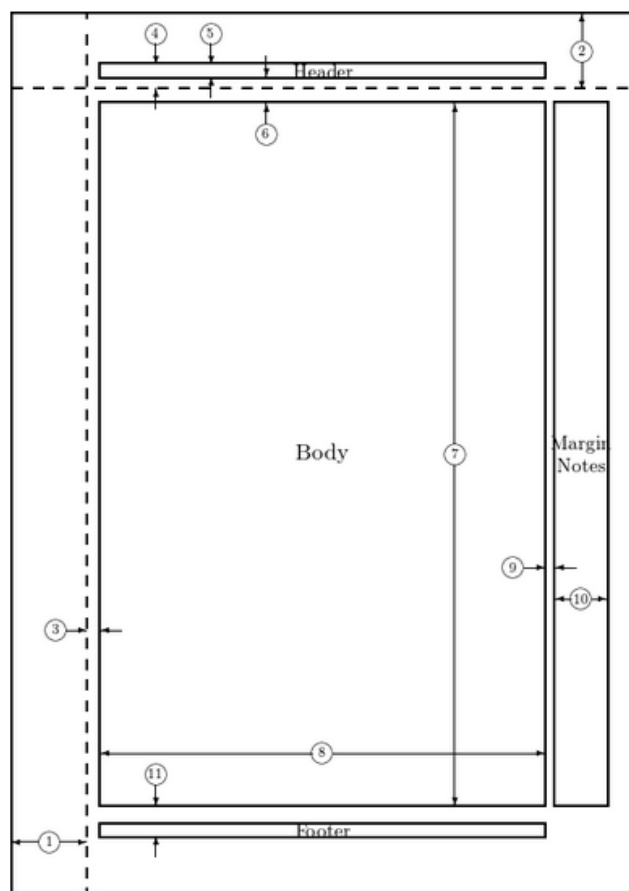
Dove `opzione1` e `opzione2` sono i nomi delle opzioni che sono state elencate sopra, mentre `dimensione1` e `dimensione2` sono le dimensioni che vogliamo associare alle opzioni che abbiamo scelto. Le unità di misura delle dimensioni. Per avere un riferimento sul significato delle varie opzioni, date un'occhiata alla Figura 5.1 (si trova a pagina 3).

Per scrivere una nota a margine, dovete utilizzare il seguente comando:

```
\marginpar{La mia nota}
```

Per scrivere una nota a piè di pagina¹, potete utilizzare il seguente comando

```
\footnote{La mia nota}
```



1pt	1mm	1cm	1ex	1em	1bp	1dd	1pc	1in	
1	2.84526	28.45274	4.30554	10.00002	1.00374	1.07	12	72.26999	pt
0.35146	1	10.00005	1.51323	3.51462	0.35277	0.37607	4.21754	25.40013	mm
0.03514	0.1	1	0.15132	0.35146	0.03528	0.03761	0.42175	2.54	cm
0.23225	0.66084	6.6084	1	2.32259	0.23312	0.24852	2.78711	16.78534	ex
0.1	0.28453	2.84528	0.43056	1	0.10037	0.107	1.2	7.22699	em
0.99628	2.83467	28.34677	4.2895	9.96277	1	1.06602	11.9553	72.00082	bp
0.93457	2.6591	26.59117	4.02385	9.34575	0.93806	1	11.21487	67.54158	dd
0.08333	0.2371	2.37106	0.3588	0.83333	0.08365	0.08917	1	6.0225	pc
0.01384	0.03937	0.3937	0.05957	0.13837	0.01389	0.0148	0.16605	1	in

Figura 4.1: Pannello superiore: geometria del documento. Il numero 7 indica la *textheight*, il numero 8 la *textwidth*, il numero 10 il *marginparwidth*, il numero 11 il *footskip*. Pannello inferiore: unità di misura di \LaTeX , corredate di equivalenze.

Esercizio 2: costruite un documento con una geometria a piacere utilizzando le 6 opzioni elencate del pacchetto e almeno 4 unità di misura diverse. Inserite nel documento un testo con una nota a piè di pagina.²

Nota a margine: cercate di focalizzare la vostra attenzione...

¹Come questa

²Tipo questa

4.2 Referenziare un'equazione

Per referenziare un'equazione scritta nell'ambiente matematico delimitato dai comandi `\begin{equation}` e `\end{equation}` è necessario utilizzare due comandi:

a `\label{nome equazione}`

b `\ref{nome equazione}`

Il primo comando serve a *etichettare* l'equazione, e va inserito subito prima di `\end{equation}`. Il secondo serve a *referenziare* l'equazione all'interno del testo, e potete utilizzarlo in qualsiasi posizione. Vediamo un esempio. Scrivendo nell'editor il seguente codice:

```
\begin{equation}
x^2+1=0
\label{equazione:secgrado}
\end{equation}
\textbf{L'equazione (\ref{equazione:secgrado}) non ha soluzioni reali}
```

Otterrete la seguente equazione e il seguente testo:

$$x^2 + 1 = 0 \tag{4.1}$$

L'equazione (5.1) non ha soluzioni reali.

Come vedete, al posto di `\ref{equazione:secgrado}` compare il numero a destra dell'equazione, automaticamente. L'equazione è stata identificata e citata nel testo con successo!

Esercizio 3: scrivete un paio di equazioni (che includano almeno le radici quadrate, l'elevazione a potenza, le frazioni, il pi greco) e referenziatele.

4.3 Esempi di codice sorgente e documento pdf associato

Nelle Figure 5.2, 5.3 e 5.4 troverete alcuni esempi di codice sorgente e di prodotti creati dal sottoscritto. Analizzando questi esempi, sarete facilitati nella stesura di un codice sorgente in \LaTeX .

Fate attenzione a collocare i comandi del preambolo prima di $\text{\begin{document}}$. Tra questi comandi rientrano tutti gli \usepackage , i parametri del titolo, e in generale tutte le definizioni che valgono nell'intero documento.

```

##### INIZIO PREAMBOLO #####
\documentclass{book} %% PRIMO COMANDO in assoluto (essenziale): definisco la classe di documento

\usepackage[utf8]{inputenc} %% Utilizzo un pacchetto (opzionale): definisco l'insieme di caratteri
letti da LaTeX

\usepackage[italian]{babel} %% Utilizzo un pacchetto (opzionale): chiedo a LaTeX di utilizzare
l'italiano come lingua di base

\usepackage{enumitem} %% Utilizzo un pacchetto (opzionale): utile per definire liste con vari
ordinali (alfabeto latino, numeri romani)

\title{Il mio libro} %% Titolo del libro (opzionale)
\author{Iacopo Sbrolli} %% Nome dell'autore (opzionale)
\date{23 ottobre 2018} %% Data di redazione dell'opera (opzionale)

##### FINE PREAMBOLO #####
\begin{document} %% Inizio del documento (obbligatorio)
##### INIZIO DOCUMENTO (tutto ciò che scrivete va qui sotto) #####

\maketitle %% Creo il titolo tramite i parametri definiti nel preambolo (opzionale)
\chapter{L'invasione dei Normanni} %% Definisco il nome di un capitolo (opzionale)
\begin{enumerate}[label=\alph*] %% Faccio un elenco alfabetico
\item Barbari
\item Berberi
\end{enumerate} %% Termino l'elenco alfabetico
\end{document} %% Fine del documento (obbligatorio)
##### FINE DOCUMENTO #####

```

Figura 4.2: Documento in \LaTeX . Si noti la presenza di soli 3 comandi obbligatori (\documentclass {«classe»}, $\text{\begin{document}}$ e $\text{\end{document}}$). Sono stati utilizzati soltanto i pacchetti che avete visto in laboratorio.

```

%***** INIZIO PREAMBOLO %*****
\documentclass{book} %% PRIMO COMANDO in assoluto (essenziale): definisco la classe di documento
\usepackage[utf8]{inputenc} %% Utilizzo un pacchetto (opzionale): definisco l'insieme di caratteri
letti da LaTeX
\usepackage[italian]{babel} %% Utilizzo un pacchetto (opzionale): chiedo a LaTeX di utilizzare
l'italiano come lingua di base
\usepackage{enumitem} %% Utilizzo un pacchetto (opzionale): utile per definire liste con vari
ordinali (alfabeto latino, numeri romani)
%***** FINE PREAMBOLO %*****
\begin{document} %% Inizio del documento (obbligatorio)
%***** INIZIO DOCUMENTO (tutto ciò che scrivete va qui sotto) %*****
\tableofcontents %% Creo l'indice del libro (opzionale)
\chapter{L'invasione dei Normanni} %% Definisco il nome di un capitolo (opzionale)
\chapter{L'invasione spagnola}
\chapter{Gli Asburgo}
\section{I Savoia barattano la Sicilia per la Sardegna}
\end{document} %% Fine del documento (obbligatorio)
%***** FINE DOCUMENTO %*****

```

Figura 4.3: Documento in \LaTeX . Si noti la presenza del comando `\tableofcontents`, che crea automaticamente un bellissimo indice dei contenuti. In questo esempio ci sono due pacchetti inutilizzati: sapreste individuarli? A vostro parere, è bene tenerli o sarebbe meglio eliminarli dal codice sorgente?

Indice

1	L'invasione dei Normanni	3
2	L'invasione spagnola	5
3	Gli Asburgo	7
3.1	I Savoia barattano la Sicilia per la Sardegna	7

Figura 4.4: Indice dei contenuti ottenuto compilando (o meglio, *componendo*) il codice in figura 5.3.

Sezione 5

Amsmath e graphicx: equazioni e immagini

Dando per assodata l'assimilazione dei concetti esplorati nei precedenti capitoli, nella presente sezione saranno trattati i seguenti argomenti:

1. Uso dei pacchetti *amsmath* e *graphicx*.
2. Inserzione di una didascalia e di un riferimento testuale per un'immagine.

Tramite le nozioni che apprenderete oggi, potrete creare dei sistemi di equazioni e inserire delle immagini all'interno del testo. **Vi ricordo che, per cominciare, è sempre necessario salvare il file su cui lavorate con un nome possibilmente privo di spazi.** A questo punto, sarebbe anche opportuno che create una cartella «informatica» dove inserire i vostri file, in modo tale che non vadano persi!

5.1 Pacchetti

5.1.1 Pacchetto *amsmath*

Il pacchetto *amsmath* è molto vasto. Tramite tale pacchetto è possibile creare numerosi ambienti matematici e inserire numerosi simboli all'interno delle equazioni.

Oggi proveremo a utilizzare l'ambiente matematico aperto e chiuso dai comandi `\begin{align}` e `\end{align}`. **Tramite questo ambiente, potrete creare equazioni multiple ben allineate.** Ciò può avere varie applicazioni:

- Rappresentazione di sistemi di equazioni
- Rappresentazione di tutti i passaggi fatti per arrivare a un certo risultato.

Vediamo un esempio di sistema di equazioni scritto tramite l'ambiente *align*:

$$x = y + z + w \tag{5.1}$$

$$2x = w + 3 + y \tag{5.2}$$

$$w = y - x^2 \tag{5.3}$$

Tale sistema di equazioni è stato prodotto dal seguente codice:

```
\begin{align}
x&=y+z+w \\
2x&=w+3+y \\
w&=y-x^2
\end{align}
```

5.1.2 L'allineamento e l'*e commerciale*

Gli elementi distintivi dell'ambiente matematico `align` sono l'*ampersand* o *e commerciale* (`&`) e il doppio backslash (`\\`). Il primo elemento definisce il punto di allineamento dell'equazione, mentre il secondo indica al codice che è necessario andare a capo per cominciare una nuova equazione.

Vediamo cosa succede se cambio il punto di allineamento della seconda equazione, ponendo l'ampersand di fronte al 3:

```
\begin{align}
x&=y+z+w \\
2x&=w+&3+y \\
w&=y-x^2
\end{align}
```

Il risultato è il seguente:

$$x = y + z + w \tag{5.4}$$

$$2x = w + 3 + y \tag{5.5}$$

$$w = y - x^2 \tag{5.6}$$

Compito 1: provate a scrivere tre diversi insiemi di tre equazioni tramite l'ambiente `align`, e fate in modo che nel primo caso siano perfettamente allineate sul segno di uguaglianza, che nel secondo caso la prima e la seconda siano allineate e la terza no, e che nel terzo caso siano disallineate tutte e tre. Create dei riferimenti a questi insiemi di equazioni all'interno del testo (del tipo: «l'equazione 4.6 è bellissima»).

5.1.3 Pacchetto `graphicx`

Tramite il pacchetto `graphicx` potrete inserire delle immagini all'interno del testo. Per farlo, potete utilizzare l'ambiente delimitato dai comandi `\begin{figure}` e `\end{figure}`, seguendo la sintassi sottostante:

```
\begin{figure}
\includegraphics[width=valore1]{nomefile}
\end{figure}
```

Il file che volete inserire all'interno del testo deve essere localizzato nella stessa cartella del vostro codice latex. **La larghezza (width) dell'immagine può essere espressa con le unità di misura che abbiamo utilizzato nel pacchetto `geometry`.** In più, potrete utilizzare anche l'unità `\textwidth`, ovvero la larghezza della porzione di pagina occupata dal testo: se volete un'immagine larga la metà del testo, dovrete scrivere:

```
width=0.5\textwidth
```

Compito 2: provate a inserire un'immagine nel vostro documento.

Potete anche inserire una didascalia per la vostra immagine. La sintassi per farlo è la seguente:

```
\begin{figure}
\includegraphics[width=valore1]{nomefile}
\caption{Didascalia dell'immagine \label{nomeimmagine}}
\end{figure}
```

Attenzione: se volete inserire un riferimento all'immagine nel testo dovete inserire il comando `\label{nomeimmagine}` all'interno del comando `\caption` (che crea le didascalie). Infatti, voi in realtà non create riferimenti alle immagini (che *non* sono numerate), ma alle didascalie (che sono numerate).

Compito 3: create un'immagine con una didascalia e create un riferimento a tale didascalia all'interno del testo

Sezione 6

Tabelle

Nel corso delle scorse lezioni avete inserito delle figure nel testo tramite il pacchetto *graphicx* e avete creato riferimenti a tali figure nel testo con i comandi `\label{}` (etichetta) e `\ref{}` (riferimento).

Dando per assodata l'assimilazione dei concetti esplorati la volta scorsa, l'esercitazione di oggi verterà sul seguente argomento:

1. Creazione di tabelle, utilizzando anche il pacchetto *makecell*.

Tramite le nozioni che apprenderete oggi, potrete creare delle simpatiche tabelline, con annessa didascalia. Provateci!

6.1 Tabelle: primo esercizio

Per creare una tabella nell'ambiente *tabular* (non è l'unico: ve ne sono anche altri), è necessario rispettare la seguente sintassi:

```
\begin{table}[h]
\centering %% Comando utile a centrare la tabella
\begin{tabular}{ccccc}
\hline
Testo 1 & Testo 2 & Testo 3 & Testo 4 & Testo 5 \\ \hline
Testo 6 & Testo 7 & Testo 8 & Testo 9 & Testo 10 \\ \hline
\end{tabular}
\caption{Didascalia}
\end{table}
```

Il risultato sarà il seguente:

Testo 1	Testo 2	Testo 3	Testo 4	Testo 5
Testo 6	Testo 7	Testo 8	Testo 9	Testo 10

Tabella 6.1: Didascalia

Esercizio 1: create una tabella con un numero arbitrario di colonne e 7 righe, con relativa didascalia e riferimento nel testo.

6.2 Tabelle:secondo esercizio

Inserendo al posto delle «c» altre lettere (l ed r) il testo all'interno delle celle viene allineato a destra o a sinistra. Inoltre, se si separano le lettere «c», «r» ed «l» con dei trattini verticali (|) è possibile separare le colonne delle tabelle. Esempio:

```
\begin{table}[h]
\begin{tabular}{c | c |} %%% Il numero di «c» corrisponde al numero di colonne
\hline (viene creata una riga orizzontale)
Testo 1 & Testo 2 \\ \hline
Testo 3 & Testo 4 \\ \hline
\end{tabular}
\caption{Didascalia \label{etichetta}}
\end{table} %%% Il numero di «c» corrisponde al numero di colonne
```

Lasciate uno spazio vuoto.

Esercizio 2: create una tabella con 3 colonne e 5 righe. Tutte le colonne dovranno essere separate da una riga verticale. Le tre colonne dovranno presentare tre diversi allineamenti del testo all'interno delle celle.

6.3 Pacchetto *makecell*: terzo esercizio

Il pacchetto *makecell* permette di inserire un testo lungo all'interno delle celle, ripartendolo in più righe. Il comando offerto dal pacchetto, anch'esso denominato *makecell*, funziona come segue:

```
\makecell{Testo lungo. \\ Devo andare a capo \\ continuamente!}
```

Il testo soprastante deve essere inserito al posto di una cella della tabella (per esempio al posto di «testo 1» nella tabella 6.1). Questa opzione può essere comoda quando all'interno di una cella è necessario riportare un grande numero di informazioni (ad esempio la descrizione di un concetto complesso).

Esercizio 3: create una tabella con 2 righe e 2 colonne. All'interno di ciascuna cella devono essere presenti testi lunghi 20 parole. La tabella non deve oltrepassare i limiti del foglio.

6.4 Il comando *multicolumn*

È spesso utile riunire le celle di più colonne in un'unica cella più grande (per dare un titolo alla tabella, per esempio). Vediamo subito il codice sorgente per farlo:

```
\begin{table}[h]
\centering %% Comando utile a centrare la tabella
\begin{tabular}{|c|c|c|c|c|}
\hline
\multicolumn{5}{|c|}{\textbf{La pasta}}
\multicolumn{3}{|c|}{\textit{Pasta fresca}} & \multicolumn{2}{|c|}{\textit{Pasta secca}}
& Bontà & Aspetto & & Aspetto \\ \hline
Pici & Ottimi & Splendidi & Penne & Spigolose \\ \hline
\end{tabular}
\caption{Pasta... e basta}
\end{table}
```

Il risultato sarà il seguente: Il comando `multicolumn` ha tre opzioni. La prima è il numero di colonne che si vogliono raccogliere

La pasta				
<i>Pasta fresca</i>			<i>Pasta secca</i>	
	Bontà	Aspetto		Aspetto
Pici	Ottimi	Splendidi	Penne	Spigolose

Tabella 6.2: Pasta... e basta

(nell'esempio soprastante, è stato inserito tre valori distinti ogni volta che è stato utilizzato il comando: 5, 3 e 2). La seconda è la centratura del testo (con la stessa sintassi utilizzata come opzione per l'ambiente `tabular`). La terza è il testo che vogliamo veder apparire.

Sezione 7

Fancyhdr, ambiente split e amssymb

Dando per assodata l'assimilazione dei concetti esplorati in precedenza, la presente sezione verterà sui seguenti argomenti:

1. Creazione di un'intestazione tramite il pacchetto *fancyhdr*
2. Uso dell'ambiente *split* del pacchetto *amsmath* e di alcuni simboli del pacchetto *amssymb*

Nel corso di questo episodio, potrete abbellire il vostro documento creando un'intestazione che aiuta i lettori a capire quale è il contenuto della pagina corrente. Potrete inoltre creare formule matematiche più avanzate.

7.1 Pacchetto *fancyhdr*

L'uso del pacchetto *fancyhdr* è semplicissimo. Nell'intestazione dovete scrivere un paio di comandi:

```
\usepackage{fancyhdr}
\pagestyle{fancy}
```

Il primo comando serve a segnalare al compilatore l'uso del pacchetto *fancyhdr*, mentre il secondo comando modifica lo stile delle vostre pagine, inserendo una riga nella parte più alta della pagina.

Se non inserite altri comandi, sopra a tale riga, a destra, comparirà il nome del capitolo corrente (se state utilizzando la classe *book*) o della sezione corrente (se state utilizzando la classe *article*). A sinistra, invece, comparirà il nome della sezione corrente (se state utilizzando la classe *book*) o della sottosezione corrente (se state utilizzando la classe *article*).

Tuttavia, questa configurazione standard può essere modificata a piacimento tramite i comandi del gruppo *fancypagestyle*. I comandi del gruppo *fancypagestyle* sono i seguenti:

1. *fancyhead*[C]: definisce il testo che comparirà nel centro dell'intestazione.
2. *fancyhead*[R]: definisce il testo che comparirà nella parte destra dell'intestazione.
3. *fancyhead*[L]: definisce il testo che comparirà nella parte sinistra dell'intestazione.
4. *renewcommand*{\headrulewidth}: modifica la larghezza della riga dell'intestazione.

Esempio d'uso:

```
\fancypagestyle{nomestile}{
\fancyhead[L]{Salumi}
\fancyhead[C]{E salami}
\fancyhead[R]{A Salemi}
\renewcommand{\headrulewidth}{1mm}
}
```

Questi comandi dovranno essere inseriti nell'intestazione. **Per utilizzare lo stile appena definito in una determinata pagina**, sarà sufficiente scrivere in tale pagina il seguente comando:

`\pagestyle{nomestile}`

Se volete invece adottare lo stesso stile in tutte le pagine sarà sufficiente utilizzare i comandi del gruppo *fancypagestyle* all'interno del testo.

Esercizio 1: create un primo documento definendo due stili di intestazione. Utilizzate il primo stile nella prima pagina e il secondo stile nella seconda pagina. Per riempire le pagine, utilizzate il pacchetto *lipsum*.

Esercizio 2: create un secondo documento definendo un unico stile di intestazione a piacere, che valga in tutto il documento.

7.2 Pacchetto *amsmath*: l'ambiente *split*

Tramite l'ambiente *align* avete già imparato a definire più equazioni in un colpo solo. Il problema è che tali equazioni sono tutte numerate, come se fossero formule distinte, mentre a volte il nostro desiderio è semplicemente rappresentare i vari passaggi algebrici che facciamo per arrivare a un certo risultato. In questo caso, è meglio utilizzare l'ambiente *split*, perché tramite di esso la numerazione assegnata all'equazione è unica.

Qui sotto trovate un esempio di equazione scritta tramite l'ambiente *split*:

$$\begin{aligned}
 x &= (x + 1)^3 \\
 x &= (x + 1)(x + 1)(x + 1) \\
 x &= x^3 + 3x^2 + 3x + 1 \\
 x^3 + 3x^2 + 2x + 1 &= 0
 \end{aligned} \tag{7.1}$$

L'ambiente *split* deve essere definito all'interno dell'ambiente *equation*. Infatti, il suo uso è il seguente:

```

\begin{equation}
\begin{split}
6&=6\\
3&=3
\end{split}
\end{equation}

```

Come nell'ambiente *align*, l'ampersand (&) serve ad allineare le righe delle equazioni nel punto desiderato, mentre il doppio backslash (\\) serve ad andare a capo.

7.3 Pacchetto *amssymb*: simboli

Il pacchetto *amsmath* permette di definire alcuni simboli. I simboli e i comandi associati sono stati riportati nella Tabella 7.1.

Esercizio 3: create 2 (due) equazioni con il comando *split*, svolgendo 4 (quattro) passaggi algebrici che coinvolgano le funzioni seno e coseno e 5 (cinque) dei simboli definiti in questa sottosezione. Le equazioni dovranno avere un senso fisico o matematico, che dovrà essere spiegato da voi nel documento.

Per far sì che PDFLaTeX compili il vostro codice scrivendo correttamente nell'output le funzioni seno e coseno, dovrete porre un backslash di fronte alle scritture *sin* e *cos* nelle vostre equazioni. Esempio di esercizio svolto (lavoro della forza di attrito radente sul piano inclinato):

Un corpo si trova su un piano inclinato (si tratta di un grosso televisore posto su Via dell'Appetito a Porto Santo Stefano). Sapendo che la massa del corpo è $m = 30$ kg, che l'accelerazione di gravità vale $9,8 \frac{m}{s^2}$, che la quota in cui si trova il corpo rispetto al suolo è $h = 30$ m e che l'angolo di inclinazione del piano è $\theta = \frac{\pi}{6}$, otteniamo:

$$\begin{aligned}
 L(\text{attrito radente}) &= m \cdot g \cdot h \cdot \frac{\cos(\theta)}{\sin(\theta)} \\
 &= 30\text{kg} \cdot 9,8 \frac{\text{m}}{\text{s}^2} \cdot 30 \text{ m} \cdot \frac{\cos(\frac{\pi}{4})}{\sin(\frac{\pi}{4})} \\
 &= 30 \cdot 9,8 \cdot 30 \text{ kg} \frac{\text{m}^2}{\text{s}^2} \cdot 1 \\
 &= 8820 \text{ J}
 \end{aligned} \tag{7.2}$$

Simboli matematici del pacchetto <i>amssymb</i>		
Simbolo	Significato	Comando
\mathbb{N}	Numeri naturali	<code>\mathbb{N}</code>
\mathbb{Z}	Numeri interi	<code>\mathbb{Z}</code>
\mathbb{Q}	Numeri razionali	<code>\mathbb{Q}</code>
\mathbb{R}	Numeri reali	<code>\mathbb{R}</code>
\mathbb{C}	Numeri complessi	<code>\mathbb{C}</code>
\nexists	Non esiste	<code>\nexists</code>
\propto	Proporzionale a	<code>\propto</code>
$\not>$	Non maggiore di	<code>\ngtr</code>
$\not<$	Non minore di	<code>\nless</code>

Tabella 7.1: Alcuni simboli utilizzabili tramite il pacchetto `amssymb`

Sezione 8

Il pacchetto xcolor

Tramite il pacchetto xcolor è possibile colorare i testi. Per inserirlo scrivete `\usepackage {textcolor}` nel preambolo.

Il pacchetto contiene un comando, con due possibili sintassi:

`\textcolor {colore}{testo}`, dove «colore» è il colore in inglese e «testo» è il testo da colorare;
`\textcolor [rgb]{ROSSO VERDE BLU}{testo}`, per creare colori personalizzati mescolando il rosso, il verde e il blu. VERDE, ROSSO e BLU hanno un valore che va da 0 (assenza del colore) a 1 (valore massimo). Ad esempio, per ottenere il fucsia, avremo solo rosso e blu, quindi scriveremo `\textcolor [rgb]{1 0 1}{testo}`.

8.1 Impostazione *newcommand*

Vi è sicuramente capitato di pensare che per scrivere determinate istruzioni in \LaTeX è necessario avere molta pazienza. I comandi del pacchetto *xcolor*, ad esempio, sono piuttosto farraginosi. **In casi come questo, conviene creare un nuovo comando, più corto e più semplice, definendolo nel preambolo.**

Per creare un nuovo comando è necessario seguire i passi sottostanti:

1. Trovare un nome per il nuovo comando.
2. Determinare il numero di argomenti del comando.
3. Individuare il blocco di istruzioni che desideriamo sostituire utilizzando il nuovo comando.
4. Scrivere nel preambolo il seguente testo:

$$\backslash\text{newcommand}\{\backslash\textit{nome comando}\}[numero\ argomenti]\{\textit{blocco istruzioni}\{\#ordinale\ argomento}\}$$

Ad esempio, se desidero sostituire il comando `\textbf` con il comando `\ev` (dove **ev** sta per *evidenzia*), dovrò scrivere il seguente testo:

$$\backslash\text{newcommand}\{\backslash\text{ev}\}[1]\{\backslash\text{textbf}\{\#1}\}$$

Esercizio 1: provate a creare dei nuovi comandi (con un solo argomento) per definire il testo colorato rosso, il testo colorato blu e il testo colorato verde.

Esercizio 2: provate a creare dei nuovi comandi per definire il titolo di un capitolo.

Esercizio 3: crea un comando per definire un testo, suddiviso in due parti, una colorata in arancione scuro e una in marrone chiaro.

Sezione 9

Frontespizi e liste personalizzate

Studiando i paragrafi del presente capitolo apprenderete altre nozioni che saranno utili a rendere professionale e completo il vostro documento. Le nozioni che apprenderete oggi vi permetteranno di effettuare le seguenti operazioni:

1. Creazione di frontespizi personalizzati.
2. Inserzione di un'immagine per lo sfondo.
3. Creazione di liste personalizzate, definite nel preambolo e facilmente utilizzabili.

9.1 Creazione di frontespizi personalizzate

Se apprenderete a creare frontespizi personalizzati potrete dare un tocco di classe al vostro documento.

Prima di tutto, si evidenzia la possibilità di creare **frontespizi standard** tramite la seguente procedura:

1. Inserzione nel preambolo dei dati del manoscritto tramite i comandi `\author`, `\title` e `\date`
2. Inserzione nel corpo del testo del comando `\maketitle`

Ad esempio, potrete scrivere `\author {Manfredi Mascioli}`, `\title {La conquista del Mondo}`, `\date {\today}` nel preambolo e `\maketitle` subito dopo il comando `\begin {document}` per far comparire uno scarno frontespizio contenente alcune informazioni essenziali.

I frontespizi standard presentano il vantaggio della rapidità della creazione, ma anche un importante svantaggio, ovvero la rigidità dell'impostazione. Se si desidera creare un libro, una relazione o una tesi di laurea, sarà spesso necessario saper impostare un frontespizio personalizzato. Per crearlo, dovrete saper gestire:

1. L'ambiente *titlepage*.
2. L'ambiente *center* per centrare il testo.
3. L'inserzione delle righe orizzontali (*rules*).
4. Il comando *vspace* per creare spazio verticale.

Un esempio schematico di frontespizio personalizzato può essere ottenuto seguendo la procedura sottostante:

1. Creo un ambiente *titlepage* dove inserire il frontespizio. Il comando è il seguente:

```
\begin{titlepage}
```

2. Creo uno spazio verticale di 2 centimetri. Scrivo:

```
\vspace*{2cm}
```

3. Apro l'ambiente *center* per centrare il testo nel frontespizio. Scrivo:

```
\begin{center}
```

4. Inserisco il titolo dell'opera utilizzando una dimensione del carattere molto grande. Scrivo:

`\textbf{\huge{titolo}}`

Oppure:

`\textbf{\Large{titolo}}`

5. Inserisco una riga orizzontale lunga venti centimetri e spessa mezzo millimetro scrivendo:

`\rule{20cm}{0.5mm}`

6. Sfruttando lo stesso comando utilizzando per scrivere il titolo, inserisco il nome dell'autore, la data e altre informazioni, separando queste informazioni tramite degli spazi verticali.

7. Chiudo gli ambienti *center* e *titlepage*

9.2 Inserzione di un'immagine per lo sfondo

Potrebbe essere utile e simpatico inserire un'immagine per lo sfondo in una qualche pagina. La pagina che si presta maggiormente a questa inserzione è chiaramente il frontespizio.

Per inserire un'immagine per lo sfondo dovrete scrivere il seguente comando (**che può essere inserito solo se nel preambolo è stato incluso il pacchetto `eso-pic`**):

`\AddToShipoutPictureBG*\includegraphics[width=\paperwidth,height=\paperheight]{immagine.png}`

La larghezza e l'altezza dell'immagine sono uguali a quelle del foglio perché lo sfondo deve coprire l'intero foglio.

Per inserire correttamente un'immagine per lo sfondo, evitando di deformarla, dovrete adattare la geometria del documento all'immagine che volete inserire, oppure ritagliare l'immagine stessa. Per visualizzare la larghezza e l'altezza dell'immagine cliccate con il tasto destro sul file immagine e poi selezionate «Proprietà». La larghezza e l'altezza dell'immagine dovranno essere direttamente proporzionali a quelli del foglio (con lo stesso fattore) affinché l'immagine non risulti deformata. In termini matematici

$$l_{\text{foglio}} = a \cdot l_{\text{imm}} \quad (9.1)$$

$$h_{\text{foglio}} = a \cdot h_{\text{imm}} \quad (9.2)$$

Quindi, se l'immagine è larga 200 pixel e alta 400 pixel, il foglio potrà essere largo 300 pixel e alto 600 pixel (il fattore a varrà 1,5 in questo caso) ma non largo 300 pixel e alto 900 pixel (l'immagine risulterà deformata).

Esercizio 1: create un frontespizio standard utilizzando i classici campi `author`, `title` e `date`.

Esercizio 2: create un frontespizio personalizzato contenente un'immagine per lo sfondo non deformata.

9.3 Inserzione di liste personalizzate

Potrebbe essere comodo creare liste tramite un ambiente personalizzato, definito nel preambolo e valido nell'intero documento.

Ipotizziamo di voler creare un ambiente che definisca liste di 3 livelli (quindi, desideriamo che il nostro ambiente possa creare liste, al cui interno siano presenti altre liste, al cui interno siano presenti ancora una volta altre liste).

Il primo livello dovrà contenere i numeri arabi, il secondo dovrà contenere i numeri arabi del primo livello e le lettere, il terzo dovrà contenere i numeri arabi del primo livello, le lettere del secondo livello e i numeri romani.

Ipotizziamo di voler chiamare il nostro ambiente «giungla». Per definire tale ambiente personalizzato dovremo scrivere nel preambolo i seguenti comandi:

```
\newlist{giungla}{enumerate}{3}
\setlist[giungla,1]{label=\arabic*}
\setlist[giungla,2]{label=\arabic{giunglai}.\alph*}
\setlist[giungla,3]{label=\arabic{giunglai}.\alph{giunglaii}.\roman*}
```

dove «giunglai» e «giunglaii» sono i **contatori** del primo livello (giungla i) e del secondo livello (giungla ii). **I contatori del terzo e quarto livello** sarebbero stati «giunglaiii» e «giunglaiiv»: si segue il sistema di numerazione romano per indicarli. Tramite questi contatori ci accertiamo che le «sottoliste» siano correttamente riferite alla lista «madre».

Per creare la vostra lista nel testo dovrete scrivere, molto semplicemente:

```
\begin{giungla}
\item Primo elemento del primo livello
\item Secondo elemento del primo livello
\begin{giungla}
\item Primo elemento del secondo livello.
\begin{giungla}
\item Primo elemento del terzo livello
\item Secondo elemento del terzo livello\[\[10pt]
\end{giungla}
\end{giungla}
\end{giungla}
```

Il risultato sarà il seguente:

- 1 Primo elemento del primo livello
- 2 Secondo elemento del primo livello
 - 2.a Primo elemento del secondo livello.
 - 2.a.i Primo elemento del terzo livello
 - 2.a.ii Secondo elemento del terzo livello

Esercizio 3: create una ambiente personalizzato per definire una lista a 4 livelli chiamata «pit». I primi tre livelli della lista dovranno contenere i numeri arabi, e il quarto livello dovrà contenere i numeri romani.

Sezione 10

Creare una bibliografia su L^AT_EX

10.1 Il concetto di bibliografia

La bibliografia è un elemento essenziale nella redazione di molte tipologie di testo (saggi, testi argomentativi, tesi, presentazioni): il lettore, analizzandola, può facilmente individuare le fonti utilizzate dall'autore e verificarne l'attendibilità. Il lettore, studiando la varietà e la qualità di tali fonti, può valutare indirettamente la profondità e la forza dimostrativa dell'opera. **Un'opera che non riporti le fonti utilizzate può essere considerata plagio, sia in ambito scolastico (dove si traduce con un voto pari a 1 per l'elaborato), sia in ambito accademico (dove si traduce con la fine della carriera del redattore dell'opera).**

Costruire la bibliografia con L^AT_EX

Ogni elemento (*item*) della bibliografia contiene numerose informazioni (nome e cognome dell'autore, titolo dell'opera citata, anno di pubblicazione, editore). Gli elementi della bibliografia devono essere ordinati in ordine alfabetico secondo il cognome dell'autore, e qualora in bibliografia fossero presenti più opere di uno stesso autore sarebbe necessario ordinarle per anno. **Con L^AT_EX questo problema non si pone:** il programma bib_TE_X, una delle applicazioni del pacchetto, effettua automaticamente questo ordinamento. **Tutto ciò che dovete fare è creare un file parallelo al codice sorgente, chiamato Bibliography.bib, dove dovrete inserire gli *item* della bibliografia e tutte le informazioni relative a ciascuno di essi.**

10.2 Il file Bibliography.bib

Nel file Bibliography.bib, che dovrete creare voi tramite l'editor L^AT_EX, sono raccolti tutti gli item bibliografici secondo una sintassi ben precisa. **Il file bibliography.bib è un semplice file di testo, privo di preambolo, che deve essere integrato tramite uno specifico comando nel vostro documento L^AT_EX affinché possa mostrare correttamente la bibliografia.** Il file bibliography.bib contiene *solo* gli item bibliografici e nient'altro.

Gli item bibliografici sono strutturati in modo diverso a seconda del tipo di opera (atti di conferenza, articolo, libro, rapporto tecnico, tesi di laurea, tesi di dottorato). Per adesso vedremo soltanto la struttura degli item **libro** e **articolo**:

1. Articolo:

```
@article{etichetta1, author={Cognome1, N1 and Cognome2, N2 and Cognome3, N3}, title={Titolo dell'articolo}, journal={Rivista}, volume={Numero del volume}, pages={Pagine occupate dall'articolo nel volume della rivista}, year={Anno di pubblicazione}}
```
2. Libro:

```
@book{etichetta2, author={Cognome1, N1 and Cognome2, N2 and Cognome3, N3}, title={Titolo del libro}, publisher={Editore}, year={Anno di pubblicazione}}
```

Dove l'«etichetta» è una sequenza di caratteri che vi ricorda l'opera che citate. Ad esempio, se volete citare il romanzo pubblicato da Anna Busonero nel 2016 intitolato «Le mie avventure a Magliano», l'etichetta può essere «magliano». **Il numero di item nel vostro file deve essere pari al numero di opere che volete citare.** Attenzione: N1, N2 e N3 sono le iniziali del nome

dell'autore seguite da un punto. Le «etichette» sono i nomi di cui vi dovrete servire qualora voleste citare le opere elencate nel file `Bibliography.bib` nel vostro documento principale tramite il comando `\cite{etichetta}`.

10.3 Inserzione della bibliografia nel vostro documento \LaTeX

Per far apparire la bibliografia all'interno del vostro documento finale dovrete semplicemente aggiungere in fondo al vostro codice sorgente (o nella vostra ultima slide) i due seguenti comandi:

```
\bibliographystyle{nomestile}
\bibliography{Bibliography.bib}
```

Dove *nomestile* è un'opzione che dispone gli *item* bibliografici in un certo modo (per ora inserite *plain* o *apalike*, sperando che non vi dia problemi). Il file *Bibliography.bib* deve essere contenuto nella stessa cartella del vostro file \LaTeX . Una volta inserite queste due righe dovrete compilare il documento con `bibTeX` (lo trovate aprendo il menù a tendina dei compilatori). Dopodiché, compilate il documento con `pdfLaTeX` e la bibliografia sarà pronta.

10.4 Citazione di un elemento della bibliografia

Per creare un riferimento a una delle opere citate sarà sufficiente utilizzare il comando `\cite {etichetta}`, dove «etichetta» è il nome che avete utilizzato per catalogare gli item bibliografici.

10.4.1 Compito del 30/11/2019

Scrivete un documento \LaTeX riempiendolo con un testo random e inserite in questo testo una bibliografia con 5 elementi; citate tutte e 5 le opere inserite nella bibliografia.

10.4.2 Compito del 7/12/2019-21/12/2019

Scrivete stavolta un testo riguardante la storia dell'Impero Sasanide citando le fonti (almeno 6). **Wikipedia non è una fonte (primaria) attendibile**, ma una fonte *secondaria*: gli articoli dell'Enciclopedia Libera sono stati scritti in base a quanto rinvenuto nelle fonti attendibili normalmente raccolte nella bibliografia delle pagine.

Il vostro testo dovrà essere suddiviso in due capitoli, ciascuno dei quali dovrà articolarsi in tre paragrafi. Nel testo dovrà essere presente un frontespizio personalizzato.

Nel testo dovrà essere integrata una risposta esaustiva alle seguenti domande:

1. Quale era la lingua più parlata nell'Impero? Di che famiglia linguistica faceva parte? Questa lingua è imparentata con l'arabo? È imparentata con l'italiano? È imparentata con l'uzbeko?
2. Quale era la religione più diffusa nell'Impero? Quali imperatori la promuovevano?
3. Quali territori erano inclusi nell'Impero?
4. Quale era la politica estera dell'Impero?
5. Come erano considerati la lingua greca e la religione cristiana nell'Impero?
6. Quando fu raggiunta la massima estensione nell'Impero?
7. Quali fatti militari causarono la caduta nell'Impero? Chi lo conquistò?

Se nel testo desideri citare il documento cui fai riferimento, usa il comando `\citeauthor{etichetta}`. Per citare l'anno di pubblicazione, usa il comando `\citeyear{etichetta}`.

I modi per citare un'opera all'interno del testo sono i seguenti:

1. Citazione indiretta: scrivete il contenuto del testo referenziato, rielaborandolo, e poi tra parentesi scrivete «cfr. autore anno». Esempio:

L'impero Sasanide fu uno stato dell'antico Medio Oriente esistito tra il III secolo d.C. e il VII secolo d.C. (cfr. Mascioli 2009).

2. Citazione testuale: scrivere il contenuto del testo referenziato, senza rielaborarlo, e poi tra parentesi scrivere «autore anno». La citazione testuale deve essere riportata tra virgolette basse. Esempio:

«L'impero sasanide riportò la stirpe iranica in Mesopotamia, ma non arrivò mai alle coste mediterranee (Corm 2009)»

Nel vostro testo inserite almeno 4 immagini: una per quanto riguarda la religione, una per quanto riguarda la rappresentazione di un imperatore, una mappa dell'impero e una fotografia o rappresentazione di una delle maggiori città dell'Impero.

Più l'elaborato è ricco, meglio è. Dovrà essere consegnato entro il 22 dicembre 2019.

Glossario

Ambiente Qualsiasi comando caratterizzato da un'apertura e una chiusura è un ambiente. Ad esempio, «figure», «tabular», « $...$ » e « $\left[...\right]$ » sono esempi di ambiente. (Pagine 1, 6, 11, 12, 15, 18, 20, 21, 23, 26, 28)

Comando Detto anche «macro». È una qualsiasi espressione \LaTeX che cominci con un backslash.. (Pagine 1, 3, 4, 5, 7, 8, 10, 12, 13, 14, 15, 16, 19, 20, 21, 23, 25, 26, 27, 29, 30, 31)

Compilatore Programma che traduce un codice da linguaggio di alto livello, leggibile dagli umani, in linguaggio macchina, interpretabile direttamente dalla CPU. (Pagine 5, 21)

Editor Programma tramite cui è possibile scrivere un testo. Un famoso editor è il blocco note. Gli editor di \LaTeX sono ottimizzati per la stesura di codici sorgente, ma la traduzione in PDF può essere realizzata solo tramite un compilatore. (Pagine 3, 4, 5, 15, 29)

Pacchetto Detto anche file di stile, è un file con estensione .sty localizzato nelle directory di texlive nel quale sono contenute le definizioni di alcuni ambienti e comandi. (Pagine 1, 7, 8, 11, 12, 13, 14, 18, 19, 21, 22, 23, 25, 27, 29)

Preambolo Parte del documento \LaTeX che precede il comando $\text{\begin\{document\}}$!. (Pagine 5, 7, 8, 10, 13, 16, 25, 26, 27, 28, 29)