

# Guida Rapida a L<sup>A</sup>T<sub>E</sub>X

Giorgio Delzanno

## 1 Cos'è L<sup>A</sup>T<sub>E</sub>X?

L<sup>A</sup>T<sub>E</sub>X è un pacchetto software basato sul software T<sub>E</sub>X che permette di costruire documenti formattati in maniera sofisticata. Come in HTML per costruire un documento in L<sup>A</sup>T<sub>E</sub>X bisogna utilizzare una serie di comandi con i quali si decora il testo del documento. L<sup>A</sup>T<sub>E</sub>X fornisce una serie di stili e comandi predefiniti per produrre articoli, libri, lucidi e formule matematiche. Per vedere il risultato della esecuzione occorre compilare un file con le annotazioni L<sup>A</sup>T<sub>E</sub>X in un file visualizzabile.

I sorgenti L<sup>A</sup>T<sub>E</sub>X hanno l'estensione `.tex`. Per vedere il risultato dell'elaborazione occorre compilare il file L<sup>A</sup>T<sub>E</sub>X usando il comando `latex nomefile.tex` che produce un file DVI visualizzabile tramite l'applicazione `xdvi` (Unix), `yap` (Windows). Un file DVI può essere trasformato a sua volta in un file postscript (`.ps`) o PDF (`.pdf`) in modo da essere poi stampato

Riepilogando,

- Prima bisogna creare il file sorgente `prova.tex` (con un qualsiasi editor, salvandolo come file testo) che contiene il testo e i comandi L<sup>A</sup>T<sub>E</sub>X
- Da una command shell (Start → Run → command) oppure dall'ambiente di lavoro (ad es. WinTex) si possono eseguire i seguenti passi
  1. **Da `.tex` a `.dvi`:** `latex prova` (crea `prova.dvi`)
  2. **Visualizzazione `.dvi`:** `yap prova`
  - 3a. **Da `.dvi` a `.ps`:** `dvips prova -o psprova.ps` (crea il file postscript)
  - 3b. **Da `.dvi` a `.pdf`:** `dvipdfm prova -o pdfprova.pdf` (crea il file PDF)

Durante la compilazione il programma `latex` potrebbe rilevare degli errori interrompendo l'esecuzione come in Fig. 1. In tal caso digitare "x", correggere gli errori, salvare il file L<sup>A</sup>T<sub>E</sub>X e lanciare di nuovo la compilazione.

## 2 Struttura di un file L<sup>A</sup>T<sub>E</sub>X

Un file L<sup>A</sup>T<sub>E</sub>X ha la seguente struttura:

- Preambolo
  - Specifiche dello stile del documento (stile, layout, ecc)
  - Comandi e macro che definiscono pattern utilizzati comunemente nel testo
- Corpo del documento che contiene il testo e i comandi di formattazione

```

This is TeX, Version 3.14159 (MiKTeX 2.1) (preloaded format=latex
2000.11.28) 3 NOV 2002 21:23 **root.tex (root.tex LaTeX2e
<2001/06/01> Babel <v3.7h> and hyphenation patterns for english,
french, german, ngerman, du mylang, nohyphenation, loaded. File:
ulasy.fd 1998/08/17 v2.2e LaTeX symbol font definitions ) !
Undefined control sequence. 1.17
? x

```

Figura 1: Errori in fase di compilazione

In generale la struttura è come segue:

```

% Stile
\documentclass{...}
\usepackage[...]{...}
...
% Comandi
\newcommand{...}{...}
\renewcommand{...}{...}

% Corpo del testo
\begin{document}
...
\end{document}

```

Segue un esempio di documento con stile article:

```

\documentclass{article}
\usepackage[italian]{babel}
\title{Esempio Documento \LaTeX}
\author{Giorgio Delzanno}
\newcommand{\AGrassetto}{ {\textbf A} }
\begin{document}
\maketitle
\section{Introduzione}
Esempio di uso di comando \AGrassetto ...

\end{document}

```

Il precente sorgente da origine al file .dvi in Fig. 2.

## 2.1 Sintassi di $\LaTeX$ ( $\TeX$ )

I comandi  $\LaTeX$  e  $\TeX$  hanno la seguente sintassi  $\backslash\text{nomecomando}[\dots]\{\dots\}\dots\{\dots\}$

- Le parentesi quadre delimitano una lista di argomenti *opzionali*
- Le parentesi graffe vengono usate per delimitare ogni singolo argomento non opzionale passato ad un comando

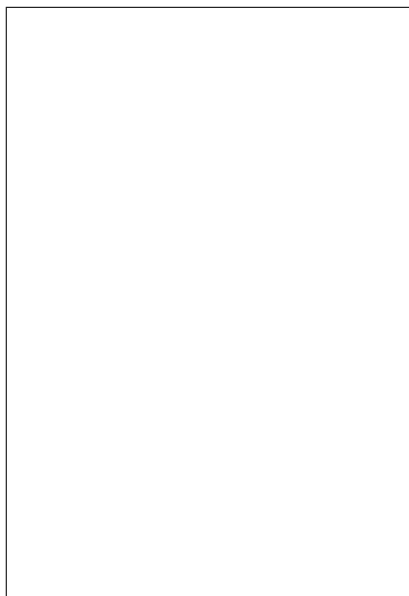


Figura 2: Esempio di file .dvi

Es. `\documentclass[12pt]{article}`

- La sintassi di  $\text{\LaTeX}$  distingue maiuscole e minuscole. Ad es. `\Ca1A` è diverso da `\ca1A`

Le parentesi graffe vengono anche utilizzate per definire degli ambienti locali all'interno dei quali ha effetto una particolare formattazione. Es. `esempio di {\bf grassetto}`

Il simbolo `%` introduce una linea commento che non verrà visualizzata nel documento (il commento va da `%` alla fine delle riga del sorgente)

## 2.2 Preambolo di un file .tex: Classi e Package

I documenti LaTeX hanno uno *stile principale* (classe) e una serie di opzioni dichiarate attraverso il comando

```
\documentclass[opzioni]{nomeclasse}
```

Una classe corrisponde ad un pacchetto predefinito (ad es. `article.cls`, `book.cls`) distribuito insieme a  $\text{\LaTeX}$  o definito dall'utente. Inoltre si possono importare degli stili secondari tramite il comando

```
\usepackage[opzioni]{nomepackage}
```

Alcuni Esempi di Classi e Package:

**article** Per definire rapporti tecnici: comandi per titolo iniziale, sezioni, bibliografia, ecc

**book** Per definire libri, tesi, ecc: comandi per suddivisioni in capitoli, sezioni, ecc.

**babel** Pacchetto per selezionare la lingua (default: inglese) tramite l'opzione (ad es. `italian`)

**graphics** Per inserire figure postscript

**slides** Per costruire presentazioni (foglio in landscape)

**amssymb,amsfonts,latexsym** Collezioni di simboli matematici predefiniti, es.  $\varphi$   $\Theta$   $\Rightarrow$   $\rightarrow$   $\star$   $\in$

Ecco un esempio di preambolo

```
\documentclass[12pt]{article}
\usepackage[italian]{babel}
\usepackage{latexsym}
\usepackage{amssymb}
\usepackage{amsfonts}

\title{Esempio Documento \LaTeX}
\author{Giorgio Delzanno}
```

## 2.3 Comandi $\LaTeX$

Oltre ai comandi  $\LaTeX$  (che vedremo in seguito) è possibile definire *nuovi comandi* ad es. una macro che rappresenta una serie di simboli/stile usati frequentemente. I comandi possono avere dei parametri. I caratteri si definiscono tramite la definizione seguente

```
\newcommand{\nomecomando}[numero di argomenti]{definizione}
```

- `\nomecomando`: nome del comando
- `definizione`: contiene il testo da sostituire a `nomecomandoarg...arg`. Gli argomenti si indicano con `#1`, `#2`, ecc.

Ecco un esempio di comando. I comandi  $\LaTeX$  `\em` `\bf` definiscono testo in italico e grassetto

```
\newcommand{\escom}[2]{ PRIMO {\em #1} SECONDO {\bf #2}} Nel corpo del documento l'invo-
```

```
\escom{cane}{gatto}
```

ca ha il seguente effetto:

```
PRIMO cane SECONDO gatto
```

## 2.4 Ambienti/**Environment**

Gli *ambienti* suddividono logicamente un documento associando alle sottoparti una formattazione particolare. Nel corpo del testo troviamo ad es. un ambiente per centrare il testo nella pagina

```
\begin{center} ... \end{center}
```

Nel preambolo si possono definire dei nuovi ambienti usando il comando

```
\newenvironment{nome}[numero arg] {inizio} {fine}
```

Inizio e fine identificano i comandi che aprono e chiudono l'ambiente.

Consideriamo la seguente definizione (dove `\\` rappresenta il fine linea)

```
\newenvironment{proof}[0]{\textbf{Proof}}{\textbf{QED}}
```

Nel corpo del testo

```
\begin{proof}
Sia  $N=2^L$  allora  $L=\log_2 N$ .
\end{proof}
```

Ha il seguente effetto

**Proof**

Sia  $N = 2^L$  allora  $L = \log_2 N$ .

**QED**

## 2.5 Corpo del testo

Il testo vero e proprio del documento deve essere racchiuso nell'ambiente predefinito `document`. All'interno di `\begin{document}... \end{document}` possiamo:

- richiamare altri file tramite il comando `input`
- inserire testo
- inserire formule matematiche
- inserire figure (tramite comandi definiti da package appositi)
- invocare comandi di formattazione (predefiniti dallo stile selezionato o definiti dall'utente)
- inserire ad annidare ambienti (array, tabelle, equazioni, teoremi, definizioni, quote, center, ecc)

**Comando `input` e strutturazione di un file `.tex`** Quando si costruiscono documenti molto grandi è buona norma lavorare su più file separati e importare poi le sottoparti in un file *radice*. Il comando `input` permette di importare un file ed inserirlo nel punto in cui compare il comando. Esempio di file radice

```
\documentclass{book}
...
\input "macro.tex"
\begin{document}
\input "capitolo1.tex"
\input "capitolo2.tex"
...
\end{document}
```

Nota: basta compilare il file radice. Per una compilazione efficiente vedere comando `\include`

**Inserimento testo** Il testo si inserisce insieme alle annotazioni per la formattazione (come in HTML). La finestra di testo è definita dallo stile prescelto (article, ecc). Il layout della pagina può essere ridefinito utilizzando il comando `\setlength{variabile}{valore}` dove *variabile* è uno dei nomi predefiniti associati alle dimensioni di un documento (vedi manuale). Es. `\topmargin`, `\oddsidemargin`, `\textheight`  
Per forzare il fine linea

- se si lascia *almeno* una riga vuota nel sorgente si termina un paragrafo nel documento risultante (si va a capo e la riga successiva viene indentata a destra)
- se si usa il comando `\\` si va a capo nel documento risultante

**Esempio: impostazione layout pagina** Tutte le misure sono relative ad un'area rettangolare predefinita (a seconda dello stile). Nel preambolo

```

\documentclass{article}
\setlength{\topmargin}{0.5in}
\setlength{\oddsidemargin}{0.5in}
\setlength{\evensidemargin}{6in}
\setlength{\textwidth}{6in}
\setlength{\textheight}{18in}
\begin{document}
...

```

### Esempio

- Sorgente L<sup>A</sup>T<sub>E</sub>X:

Lascio una riga

ora vado a capo con l'apposito comando `\space`  
 verticale `\medskip` maggiore

- Risultato:

Lascio una riga

ora vado a capo  
 con l'apposito comando  
 spazio verticale  
 maggiore

## 3 Esempi di comandi

Ecco un esempio di citazione [AJ01] all'interno del documento.

### 3.1 Formattazione Testo

**Layout testo** L<sup>A</sup>T<sub>E</sub>X fornisce una vasta gamma di comandi per la formattazione dello stile del testo. Ecco alcuni esempi

Comandi L <sup>A</sup> T <sub>E</sub> X	Risultato
<code>\'e \'e \'a \'a \'o \'u</code>	è é à á ò ù
<code>\textit{Ancora un Esempio}</code>	<i>Ancora un Esempio</i>
<code>\textsf{Esempio}</code>	<b>Esempio</b>
<code>\texttt{Esempio}</code>	<b>Esempio</b>
<code>\textbf{Esempio}</code>	<b>Esempio</b>
<code>\textsc{Esempio}</code>	ESEMPPIO
<code>{\rmfamily Ancora un Esempio}</code>	Ancora un Esempio
<code>{\sffamily Esempio}</code>	Esempio
<code>{\ttfamily Esempio}</code>	Esempio
<code>{\bfseries Esempio}</code>	<b>Esempio</b>

### 3.2 Tabular

L'ambiente tabular permette di costruire tabelle.

```
\begin{center}
\begin{tabular}{|l|c|c|}
Citt\ 'a & Max & Min\\
\hline\hline
Torino & 18 & 5\\
\hline
Milano & 17 & 10\\
\hline
Genova & 20 & 15
\end{tabular}
\end{center}
```

Ecco il risultato dei precedenti comandi.

Città	Max	Min
Torino	18	5
Milano	17	10
Genova	20	15

### 3.3 Formule matematiche

Le formule matematiche devono essere racchiuse all'interno di un particolare ambiente delimitato dal simbolo \$ (che applica una formattazione ad hoc). Es. da  $x+y=2$  e  $\$x+y=2\$$  otteniamo  $x+y=2$  e  $x + y = 2$ . L'ambiente delimitato da \$\$ centra le formule corrispondenti all'interno della pagina. Es. da  $\$3x^2+5x^4+x=2\$$  otteniamo

$$3x^2 + 5x^4 + x = 2$$

#### Esempi di formule

Comando L <sup>A</sup> T <sub>E</sub> X	Risultato
$x^y$ $x^{y^4}$	$x^y$ $x^{y^4}$
$a_i$ $a_{i_j}$	$a_i$ $a_{i_j}$
$a^k_i$	$a_i^k$
$\sqrt{1+x}$ $\sqrt{1+\sqrt{1+x}}$	$\sqrt{1+x}$ $\sqrt{1+\sqrt{1+x}}$
$\frac{x}{y+1}$ $\frac{x}{\frac{3}{x+1}}$	$\frac{x}{y+1}$ $\frac{x}{\frac{3}{x+1}}$
$\Sigma_{i \in I} x^i$	$\Sigma_{i \in I} a^i$
$\underbrace{x+y+z}_{\text{somma}}$	$\underbrace{x+y+z}_{\text{somma}}$

#### Formule complesse

Comando L <sup>A</sup> T <sub>E</sub> X	Risultato
<code>\$A \Rightarrow B\$</code>	$A \Rightarrow B$
<code>\$\$\langle A,B \rangle\$</code>	$\langle A, B \rangle$
<code>\$\$\stackrel{A}{B}\$</code>	$\frac{A}{B}$
<code>\$\$\langle A,B \rangle \stackrel{\gamma}{\Longrightarrow} A\$</code>	$\langle A, B \rangle \xrightarrow{\gamma} A$

### Package amsmath

Comando L <sup>A</sup> T <sub>E</sub> X	Risultato
<code>\$A \xrightarrow{aaaaaaaa} B\$</code>	$A \xrightarrow{aaaaaaaa} B$
<code>\$\$\overset{a}{\underset{b}{X}}\$</code>	$\overset{a}{\underset{b}{X}}$
<code>\$\$\sideset{}^{1_2}{^3_4}\prod\$</code>	$\prod_{2_4}^{1_3}$

### Ambienti Matematici: Equation Array

```
\begin{equation}\label{eqz}
3x^2+4y^5+xy \geq 5
\end{equation}
```

L'equazione `\ref{eqz}` rappresenta ...

$$3x^2 + 4y^5 + xy \geq 5 \tag{1}$$

L'equazione 1 rappresenta ...

### Ambienti Matematici: Array

```
\begin{array}{cc}
a+b \geq 5 & a \geq 1 \\
c \geq 5 & c+b \geq 3
\end{array}
```

$$\begin{array}{cc} a+b \geq 5 & a \geq 1 \\ c \geq 5 & c+b \geq 3 \end{array}$$

### Ambienti Matematici: Array

```
$$
f(x)=\left\{
\begin{array}{cc}
\sqrt{x} & x \geq 0 \\
\sqrt{|x|} & x < 0
\end{array}
\right.
$$
```



$$f(x, y) = x^y$$

Figura 3: Definizione della funzione  $f$ .

$$f(x) = \begin{cases} \sqrt{x} & \text{se } x \geq 0 \\ \sqrt{|x|} & \text{se } x < 0 \end{cases}$$

### 3.4 Figure e didascalie

```
\begin{figure}[t]
$$f(x,y)= x^y$$
\caption{Definizione della funzione $$f$$.} \label{deff}
\end{figure}
```

Fig. \ref{deff} presenta la definizione ....

Fig. 3 presenta la definizione ....

### 3.5 Riferimenti incrociati

Negli stile article e book il testo viene suddiviso in capitoli e sezioni alle quali si può associare un'etichetta che viene utilizzata per i riferimenti incrociati. Ecco un esempio

```
\section{Introduzione}\label{sec:intro}
\subsection{Notazione}\label{sec:notazione}
In Sez. \ref{sec:notazione} \dots
```

Ecco il risultato dei precedenti comandi.

## 4 Introduzione

### 4.1 Notazione

In Sez. 4.1 ....

### 4.2 Riferimenti bibliografici tramite BIBTEX

In un file separato (con estensione .bib) si prepara una lista di citazioni bibliografiche (vedremo dopo la sintassi). Un campo particolare funge da etichetta della citazione ad es. BG99. Nel testo si utilizza il comando \cite{BG99} per fare riferimento alla citazione considerata. Nel file  $\LaTeX$  principale si inserisce il seguente comando per importare la lista di citazioni

```
\bibliographystyle{stile bibliografia}
\bibliography{nomefile BibTeX}
```

Dopo aver compilato il file latex, occorre eseguire il comando `bibtex nomefile` (nome del file  $\LaTeX$  radice), successivamente occorre eseguire nuovamente  $\LaTeX$ .

I file .bib consistono di una lista di record con i campi dei riferimenti bibliografici

```
@article{AJ01,
  author = "P.~A.~Abdulla and B.~Jonsson",
  title = "{Verification of Infinite-State Systems}",
  journal = "{Theoretical Computer Science}",
  volume = "256",
  number = "1-2",
  pages = "145-167",
  year = "2001"
}
```

In questo esempio AJ01 rappresenta l'etichetta da utilizzare con il comando `\cite` nel corpo del testo. Nel corpo del documento si utilizzano le citazioni come segue.

In `\cite{AJ01}` Abdulla e Jonsson ....

Il risultato è come segue.

In [AJ01] Abdulla e Jonsson ....

## Riferimenti bibliografici

- [AJ01] Abdulla, P. A., & Jonsson, B. (2001). Ensuring Completeness of Symbolic Verification Methods for Infinite-State Systems. *Theoretical Computer Science*, **256**(1-2), 145–167.
- [ACJT96] Abdulla, P. A., Cerāns, K., Jonsson, B., & Tsay, Y.-K. (1996). General Decidability Theorems for Infinite-State Systems. *Pages 313–321 of: Proceedings of LICS'96*. New Brunswick, New Jersey: IEEE Computer Society Press.

## 5 Esercizi

All'interno dell'ambiente `tabular` il comando `\multicolumn{cols}{tipocol}{testo}` permette di costruire righe che si estendono su più colonne (tante quante specificate in `cols`). Il secondo parametro specifica il tipo della cella risultante (`r,l,c`, ecc) Il comando `\hline` inserisce una riga orizzontale (va usato subito dopo `\\`). Il comando `\cline{col-col'}` inserisce una riga orizzontale dalla colonna `col` (compresa) alla colonna `col'` (compresa) Ad esempio i seguenti comandi

```
\begin{tabular}{|1|1|r|}
\hline
\multicolumn{3}{|c|}{Titlo}\\
A & B & 2\\
C & D & 4\\
E & F & 10
\hline
\end{tabular}
```

producono la tabella

Titlo		
A	B	2
C	D	4
E	F	10

Come esercizio provate a ricostruire la seguente tabella.

A	B		
C			D
E	F		G
H	I	L	M

## 5.1 Esercizio

Riprodurre la seguente figura usando l'ambiente tabular

A				B
	A			B
C		A		B
	C	C	A	B
			C	•

## 5.2 Comandi

- Definire tre nuovi comandi `\prova`, `\provab`, `\provac` ognuno con 2 argomenti (usare la definizione `\newcommand{\prova}[2]{...}`)

1. `\prova{a}{b}` produca  $\sqrt{a + \sqrt{b}}$
2. `\provab{a}{b}` produca  $a^b + a^{a^b} + a^{a^{a^b}}$
3. `\provac{a}{b}` produca  $\begin{vmatrix} a & b \\ b & a \end{vmatrix}$

- Nota:

- `\sqrt{A}` =  $\sqrt{A}$
- `A^B` =  $A^B$
- `\array` produce tabelle come `tabular` ma in ambiente matematico