

Introduzione al
L^AT_EX
Lezioni per la Libera Pluriversit  di Napoli

di
Giuseppe Romano

21 febbraio 2009

Breve introduzione:

Il presente documento vuole essere una sorta di “appunti” di quelle che sono le lezioni tenute presso la Libera Pluriversità di Napoli, allo scopo di fornire agli studenti del materiale di base facilmente consultabile per produrre eventuali documenti \LaTeX . Ogni capitolo rappresenta una lezione del corso che è stato stimato in un massimo di tre lezioni.

Indice

1	lezione 1 (di Sabato 21/02/2009)	3
1.1	Cos'è \LaTeX	3
1.1.1	La sintassi dei comandi	4
1.1.2	La compilazione	5
1.2	Struttura del documento	5
1.2.1	Il preambolo	6
1.2.2	Il corpo del documento	10
1.2.3	Il primo documento \LaTeX	12

Capitolo 1

lezione 1 (di Sabato 21/02/2009)

Nella prima lezione del corso è stato dato uno sguardo generale a quello che è il mondo LaTeX, la sua filosofia e le compatibilità. Abbiamo visto com'è strutturato un file di input e dato uno sguardo ai primi semplici comandi.

1.1 Cos'è L^AT_EX

Il LaTeX è uno strumento particolarmente efficace nello scrivere libri, relazioni, documenti in genere. Esso fa parte di quei linguaggi di programmazione detti di *markup*¹ e si basa su TeX che ne rappresenta l'anima. Un generico comando LaTeX è composto da due marcatori, uno di apertura e uno di chiusura; all'interno dei due marcatori va scritto il testo relativo. Un marcatore di apertura è generalmente composto da una sequenza di caratteri del tipo `\begin{comando}`, mentre un marcatore di chiusura è generalmente del tipo `\end{comando}`. Il testo scritto all'interno dei due marcatori verrà interpretato da LaTeX in modo diverso a seconda del *comando* indicato; Ad esempio nel caso volessi scrivere una porzione di testo centrato rispetto al resto del documento, mi basterà scrivere tale testo tra i due marcatori `\begin{center} ... \end{center}`.

Al contrario di editor (o word processor) più conosciuti quali ad esempio Microsoft Word, WordPerfect, Works, Writer della suite StarOffice (o OpenOffice.org), che si basano sul paradigma WYSIWYG (What You See Is What

¹Il termine markup (o marcatura) deriva dall'ambiente tipografico dove si usava marcare con annotazioni le parti del testo che andavano evidenziate o corrette, allo scopo di segnalarle al compositore o al dattilografo. In generale un linguaggio di markup descrive i meccanismi di rappresentazione (strutturali, semantici o presentazionali) del testo che, utilizzando convenzioni standardizzate, sono utilizzabili su più supporti. Il LaTeX è un linguaggio di markup di tipo procedurale, ossia di quelli che indicano le procedure di trattamento del testo aggiungendo le istruzioni che devono essere eseguite per visualizzare la porzione di testo referenziata. [1]

You Get, cioè ciò che vedi è quello che ottieni), con LaTeX si scrive un testo preoccupandosi essenzialmente del contenuto (della struttura) e non della forma. Il testo del documento, detto sorgente, conterrà anche delle istruzioni (direttive di LaTeX): per ottenere l'output finale è necessario che tale sorgente sia poi compilato. Questo approccio viene anche definito WYSIWYM (What You See Is What You Mean, cioè ciò che vedi è quello che intendi): con LaTeX l'autore inizialmente può occuparsi delle convenzioni da usare (nel preambolo), ma una volta fissate queste, si può concentrare soltanto sul contenuto del testo. L'impaginazione, l'indice (generale e analitico), l'inserimento delle figure e delle tabelle sarà semi-automaticamente curato da LaTeX. Il file prodotto da LaTeX era, in passato, esclusivamente in formato DVI (DeVice Independent). Grazie al contributo degli sviluppatori della comunità open source ora LaTeX è in grado di produrre un file nel più comune e diffuso standard PDF (Portable Document Format) ed anche in HTML (le eventuali formule matematiche in esso presenti verranno incluse in formato grafico come se fossero immagini, se non in MathML). È anche possibile, partendo da un file prodotto da LaTeX, ottenere un qualsiasi altro formato, anche .doc di Microsoft Word (oppure un .odt di OpenOffice.org) [2].

Un file LaTeX è un file di tipo testo e pertanto può essere scritto con qualsiasi editor di testo, l'unica accortezza da avere è quella di terminare il nome del file con “.tex” . Il file con estensione “tex” è detto *file sorgente*; esso è il file che deve essere mandato al compilatore che, interpretando i comandi contenuti, genererà (a seconda del comando ordinatogli) il file di output, ossia il documento finale, con estensione .pdf e/o doc e/o html e/o dvi ...ecc.

Esempio:

Il seguente codice:

la parola `\begin{center} testo \end{center}` è scritta al centro.

produce questo output:

La parola

testo

è scritta al centro.

1.1.1 La sintassi dei comandi

i comandi LaTeX sono, all'inizio, un pò difficili da interpretare ma poi lavorando ci assumono una certa familiarità. Sostanzialmente ci sono due tipologie di comandi: quelli che hanno il marcatore di apertura e di chiusura e quelli *di riga* che non hanno un marcatore di chiusura. Tutti i comandi LaTeX iniziano con il simbolo “\” (backslash) seguito da una sequenza di lettere minuscole, ad esempio il comando `\textbf{ciao}` indica al compilatore di

scrivere la parola “ciao” in grassetto. I comandi più semplici sono quelli di riga perché in genere non hanno opzioni da impostare, mentre l'altra tipologia sono più complessi: essi iniziano con un `\begin{comando}`, e finiscono con un `\end{comando}` e al posto della parola *comando* va indicato il nome del comando che si vuole utilizzare. Ad esempio nel caso volessi scrivere una porzione di testo centrato rispetto al resto del documento, mi basterà scrivere tale testo tra i due marcatori `\begin{center}` ed `\end{center}`. Badate che i comandi di LaTeX sono *case-sensitive*, cioè fanno distinzione tra maiuscolo e minuscolo, per cui il comando `\Textbf` è errato.

1.1.2 La compilazione

Per compilare un file sorgente LaTeX occorre inviare il file `.tex` al compilatore precedentemente istallato. Per chi usa sistemi Linux, il compilatore è istallato di default, mentre per i sistemi windows occorre scaricarsi un software adatto² e di solito non hanno bisogno di comandi particolari ma contengono un pulsantino tipo “run” o “exec” che permette di compilare il sorgente. Ritornando ai pinguini, è possibile compilare un sorgente LaTeX aprendo il terminale e digitando il comando `latex nomedelfile.tex`; questo comando genererà un file output con estensione `.dvi`, mentre se si vuole il formato `.pdf` occorre digitare `pdflatex nomedelfile.tex`. Per gli utenti Linux che preferiscono istallare un software dedicato, le modalità di compilazione cambiano in base al software, mentre quelle su mensionate sono universali.

1.2 Struttura del documento

Un documento LaTeX, oltre ad avere l'estensione `.tex`, deve anche essere strutturato in un modo ben determinato per far sì che il compilatore riesca a interpretarlo. La struttura di un file LaTeX è abbastanza semplice, essa è sostanzialmente divisa in due parti, il *preambolo* e il *corpo* del documento. Il preambolo serve a

- specificare il tipo di documento che si vuole realizzare (*impostazione della Classe*)
- indicare la lingua utilizzata
- impostare il titolo, l'autore e la data
- importare i comandi necessari mediante i cosiddetti *pacchetti*

²controllare sui siti <http://www.tug.org/protext/> e <http://www.miktex.org/>

- impostare il layout del documento

Il corpo del documento invece è la parte contenente il contenuto, ossia il testo, le figure, gli esempi eccetera; esso è delimitato dai due marcatori `\begin{document}` ... `\end{document}`. Tutto ciò che è scritto dopo il marcatore di chiusura `\end{document}` viene ignorato dal compilatore, pertanto non sarà presente nel documento prodotto, mentre tutto ciò che è scritto prima del marcatore di apertura `\begin{document}` farà parte del preambolo.

1.2.1 Il preambolo

Abbiamo detto che il preambolo serve sostanzialmente a impostare la struttura del documento, in modo tale da poter successivamente interessarsi solo al contenuto. Nel preambolo si importano anche i cosiddetti *pacchetti* ossia quelle *collezioni* di comandi che possono servire per la stesura del documento; alcuni di questi pacchetti sono necessari, altri sono più o meno facoltativi. La sintassi per utilizzare un pacchetto è:

```
\usepackage[opzioni]{nome pacchetto}
```

I pacchetti necessari sono:

nome pacchetto	sintassi
babel	Serve a specificare la lingua utilizzata. Per impostare l'italiano si deve sostituire a <i>opzioni</i> la stringa <i>italian</i>
inputenc	Serve a specificare la <i>codifica dei caratteri</i> . Per i sistemi Microsoft Windows si deve sostituire a <i>opzioni</i> la stringa <i>latin1</i> mentre per i sistemi LINUX la stringa utf8x

Oltre ai pacchetti, nel preambolo si specifica il tipo di documento che si vuole realizzare, questo avviene tramite la cosiddetta *impostazione di classe*. La sintassi per impostare la classe è la seguente

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

Le classi più comuni sono:

- **book**
quando si intende scrivere un libro

- **report**

quando si intende scrivere una relazione abbastanza lunga, una tesina ecc..

- **article**

quando si intende scrivere articoli in riviste scientifiche, presentazioni, brevi relazioni ...

Nelle opzioni, invece, si specifica la grandezza del carattere, se il documento deve essere stampato fronte/retro o meno, su una o più colonne, il tipo di foglio utilizzato eccetera... Le opzioni più comuni sono:

- 0pt, 11pt, 12pt
Imposta la dimensione del font principale nel documento; se non è specificata, si assume 10pt.
- a4paper, letterpaper, a5paper, b5paper, executivepaper, legalpaper
per definire le dimensioni del foglio. La dimensione predefinita è letterpaper.
- fleqn
Compone le formule visualizzate allineandole a sinistra invece che centrandole.
- leqno
Dispone la numerazione delle formule sulla sinistra invece che sulla destra.
- titlepage, notitlepage
Specifica se dopo il titolo del documento debba avere inizio o no una nuova pagina. La classe article non dà inizio a una nuova pagina per impostazione predefinita, mentre report e book agiscono in questo modo.
- twocolumn
Dà istruzioni a LaTeX di impaginare il documento in due colonne.
- twoside, oneside
Specifica se deve essere generato un output a doppia o singola facciata. Le classi article e report sono a singola facciata e la classe book è a doppia facciata per impostazione predefinita. Va notato che quest'opzione riguarda solo lo stile del documento; l'opzione twoside non comunica alla stampante usata che l'output di stampa debba effettivamente essere a due facciate.

- `openright`, `openany`

Fa iniziare i capitoli solo nella pagina destra o nella successiva pagina a disposizione. Questa opzione non funziona con la classe `article`, poichè questa non contiene informazioni riguardo ai capitoli. La classe `report` fa iniziare in maniera predefinita i capitoli sulla prossima pagina a disposizione e la classe `book` li fa iniziare sulle pagine destre.

Questo documento ha usato questa definizione di classe:

```
\documentclass[12pt, oneside, a4paper]{report}
```

IMPORTANTE: *la definizione della classe deve essere sempre la prima riga di un documento LaTeX.*

Oltre alla definizione della classe e all'importazione di comandi tramite i pacchetti, nel preambolo possono essere utilizzati alcuni comandi che possono essere utili alla stesura del documento: ad esempio, ci sono dei comandi che servono alla definizione del titolo o dell'autore o della data, oppure alla definizione dello stile della pagina, dell'interlinea piuttosto che i margini. Segue quindi un breve e non esaustivo (ci vorrebbe un manuale dedicato) elenco dei possibili comandi da poter richiamare nel preambolo:

- `\title{}`
tra le parentesi graffe va inserito il titolo del documento
- `\author{}`
tra le parentesi graffe va inserito l'autore del documento
- `\date{}`
tra le parentesi graffe va inserita la data del documento
- `\linespread{1.5}`
per impostare un interlinea di 1.5
- `\evensidemargin=3.75cm`
per impostare il margine destro a 3.75 cm
- `\oddsidemargin=3.35cm`
per impostare il margine sinistro a 3.35 cm
- `\pagestyle{stile}`
per stabilire lo *stile* della pagina. Ovviamente la parola *stile* va sostituita con una che indica lo stile scelto, ad esempio si usa:

1. plain
per stampare i numeri di pagina nella parte inferiore della stessa, al centro del pié di pagina. Questo è lo stile di pagina predefinito.
2. headings
per stampare il titolo del capitolo corrente e il numero di pagina sulla intestazione di ciascuna pagina, mentre il pié di pagina rimane vuoto.
3. empty
imposta la intestazione e il pié di pagina in modo che non vi sia stampato nulla.

[3]

1.2.2 Il corpo del documento

Una volta scritto il preambolo e impostato il layout della pagina, ci possiamo occupare del corpo del documento. Esso ha un inizio e una fine, l'inizio viene indicato con il comando `\begin{document}` e la fine con il comando `\end{document}`. In mezzo a questi due comandi va scritto tutto il documento. E' importante evidenziare che nello scrivere il documento si potrebbe aver bisogno di scrivere in grassetto o corsivo, creare degli elenchi, impostare dei capitoli, dei paragrafi ecc.. e per fare ognuna di queste cose occorre conoscere il comando LaTeX dedicato. Quindi passo alla rassegna di alcuni di questi fondamentali comandi.

comando	resa
<code>\textit{testo in corsivo}</code>	<i>testo in corsivo</i>
<code>\textbf{testo in grassetto}</code>	testo in grassetto
testo <code>\emph{enfattizzato}</code> nel normale	testo <i>enfattizzato</i> nel normale
<code>\textit{ testo \emph{enfattizzato} nel corsivo }</code>	<i>testo enfattizzato nel corsivo</i>
Scrivo una semplice funzione: <code>\$f(x)=\frac{x^2}{\sqrt{1-x}}\$</code>	Scrivo una semplice funzione: $f(x) = \frac{x^2}{\sqrt{1-x}}$
scrivere un elenco numerato: <code>\begin{enumerate} \item primo \item secondo \end{enumerate}</code>	scrivere un elenco numerato: 1. primo 2. secondo
scrivere un elenco puntato: <code>\begin{itemize} \item primo \item secondo \end{itemize}</code>	scrivere un elenco puntato: • primo • secondo
questa ‘parola’ è virgolettata	questa “parola” è virgolettata
<code>1 \leq 5 \geq 3</code> che è <code>\neq 2</code>	$1 \leq 5 \geq 3$ che è $\neq 2$
<code>\\$ \& \% \# _ \{ \}</code>	$\$ \& \% \# - \{ \}$

[3] [4]

Un'altra particolarità interessante di LaTeX è la gestione degli spazi, infatti

LaTeX fa in modo che qualsiasi sia il numero di spazi che lasciate tra una parola e l'altra il compilatore ne stampi in output sempre e solo uno, cosicché il seguente testo:

```
(lascio molto spazio tra          le          parole)
```

viene così reso:

```
(lascio molto spazio tra le parole)
```

Durante la scrittura di un documento può essere necessario scrivere delle note a piè di pagina, il comando per scrivere tali note è `\footnote{testo della nota}`. E' importante far notare che LaTeX numera le note in modo automatico. Altrettanto in modo automatico è la numerazione dei capitoli, delle sezioni e delle sottosezioni di un libro, così come dell'indice principale. I comandi per fare queste cose sono:

- `\Chapter{titolo del capitolo}`
- `\section{titolo della sezione}`
- `\subsection{titolo della sottosezione}`
- `\paragraph{titolo del paragrafo}`³
- `\maketitle`⁴
- `\tableofcontents`⁵

³i paragrafi non hanno una numerazione

⁴stampa il titolo definito nel preambolo nel punto in cui viene digitato

⁵stampa l'indice del documento nel punto in cui viene digitato

1.2.3 Il primo documento LaTeX

Arrivati a questo punto, occorre fornire un semplice esempio di un documento LaTeX:

```
\documentclass[12pt, oneside, a4paper]{report}
\title{primo documento in \\ LaTeX}
\author{\small{di} \\ \small{Giuseppe Romano}}
\date{\small{21 febbraio 2009}}
\usepackage[italian]{babel}
\usepackage[utf8x]{inputenc}
% per windows usare "latin1" al posto di "utf8x"
\pagestyle{headings}
% FINE DEL PREAMBOLO
\begin{document}
\maketitle
\tableofcontents
\chapter{Esempio documento}
  scrivo      un pochino di  testo
\section{sezione del capitolo}
bla bla bla bla bla
\subsection{soto-sotto-sezione}
testo testo, ... testo
\chapter{nuovo capitolo}
  scrivo ancora un altro pochino di  testo
\section{sezione del capitolo secondo}
\end{document}
```

Con un semplice copia-incolla, compilatelo e verificatene la resa. Come tutti i linguaggi di programmazione, è prevista anche per LaTeX la fase di *debug*, ovvero la fase di correzione degli eventuali errori. E' opportuno, quando si scrive un documento LaTeX, utilizzare un editor di testo che permetta la numerazione delle linee di testo. Infatti quando il compilatore trova un errore, vi comunica il tipo di errore e a quale linea è stato trovato. Nel caso di documenti molto lunghi, come i libri, diventa necessario avere tale numerazione allo scopo di individuare più velocemente l'errore.

CIAO A TUTTI

ALLA PROSSIMA

bye bye a tutti

Prossima lezione 21/03/2009

Bibliografia

- [1] http://it.wikipedia.org/wiki/Linguaggio_di_markup
- [2] <http://it.wikipedia.org/wiki/LaTeX>
- [3] Tobias Oetiker, Hubert Partl, Irene Hyna e Elisabeth Schlegl *Una mica tanto breve guida al LaTeX - Ovvero LaTeX in 93 minuti* Versione 3.16, 25 settembre 2000
- [4] Marc Baudoin *Impara LATEX! (...e mettilo da parte)*, 1994-1998