

Corso intensivo di LaTeX

Tutto quello che c'è da sapere per scrivere un documento (tesi, tesine, articoli, relazioni, ...) con LaTeX

Autore e speaker: Luca Tesei

Corso offerto dal Camerino Linux User Group

<http://camelug.unicam.it>

WYSIWYG

- **W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et
- I “moderni” word processor sono per lo più di questo tipo
- Viene mostrato in tempo reale il risultato che si produrrà in output (stampato, ebook, ...)
- Microsoft Word, OpenOffice Text Editor, Editor di Pagine Web, ...

Problemini...

- Chi scrive è portato a concentrarsi troppo sulla formattazione
- Chi scrive in genere non è un tipografo professionista
- Il documento risultante non è formattato in maniera ottimale
- La struttura logica del documento è debole e/o non ben evidenziata

Comunque...

- Le applicazioni word processor WYSIWYG sono preziose per testi
 - Relativamente brevi (lettere, brevi relazioni, avvisi)
 - Senza struttura rilevante
 - Con pochi riferimenti interni
 - Informali

Comunque...

- Le applicazioni **non word processor WYSIWYG** sono utilissime:
 - Creatori di presentazioni
 - Ritoccatori di immagini
 - Publishing
 - CAD
 - ...

WYGIWYM

- **W**hat **Y**ou **G**et **I**s **W**hat **Y**ou **M**ean
- Chi scrive si concentra solo sulla struttura **logica** del testo
- Questo è il titolo, questa è una sezione, questo è un paragrafo, questa è una tabella, questa è una lista di opzioni, questo è un teorema, questa è una dimostrazione, questo è un algoritmo, ...

WYGIWYM

- Descrizione logica degli elementi del testo + testo
- Tutto ciò che serve ad un esperto del settore per:
 - Decidere il layout
 - Impaginare il tutto
 - Ottenere un documento stampato armonioso, altamente leggibile, ineccepibile

Costi

- Quanto costano i servizi di un tipografo professionista e di un esperto di editing?
- Non sono certo alla portata di tutti
- Lavorano per case editrici
- E gli studenti che devono scrivere la tesi??

Una soluzione economica

- Installare LaTeX su un pc ed ottenere così un perfetto tipografo ed esperto di editing automatico

LATEX

- E' un interprete di macro basato su TeX
- Scritto da Leslie Lamport nel 1984
- Legge un file di testo contenente descrizione del testo + testo
- Produce in output un documento ben formattato in vari formati

TeX

- Scritto da Donald E. Knuth a partire dal 1978
- Qui vediamo la mascotte :-))



Tipico approccio hacker ad un problema :-))

- Knuth stava scrivendo il suo libro colosso “The art of computer programming”
- Era insoddisfatto della qualità e laboriosità della formattazione
- Decise di crearsi un sistema di formattazione automatico: il TeX
- Ebbe molto successo :-))

Come Funziona

```
%File di  
%testo  
  
\documentsty  
  
\begin{docu  
\maketitle  
.....
```



L^AT_EX

TeX



Device
Independent
File (.dvi)



PostScript File (.ps)

Portable Document
Format (.pdf)



Figure

TeX o LaTeX?

- Scrivere direttamente in codice TeX è difficile
- LaTeX è stato creato appositamente per rendere accessibile a tutti le potenzialità di Tex
- LaTeX viene compilato in TeX
- LaTeX è sufficiente per normali esigenze
- In questo corso studieremo esclusivamente LaTeX

Che cosa serve?

- Un editor di testi
 - Editor classici oppure specializzati: Kile, Texmaker, Winefish, Lyx, TeXnicCenter, ...
- Le applicazioni TeX e LaTeX
 - Entrambe **Free Software**
 - Hanno una licenza personalizzata
 - Multiplatforma: Unix/Linux, Windows, Mac
- Software per visualizzare/stampare file di tipo PostScript e/o PDF
 - ggv, xpdf, Acrobat Reader, Ghostview, ...

Utenti Unix/Linux

- Probabilmente in tutte le installazioni base è incluso il TeX e il LaTeX
- Per verificare provare a dare il comando
> latex
dalla linea di comando
- Sicuramente emacs o un altro editor che evidenzi la sintassi LaTeX è disponibile in tutte le installazioni base
- Altrimenti installare il pacchetto fornito dalla propria distribuzione/fornitore

Utenti Windows

- Installare MikTeX, implementazione di (La)TeX per Windows, da <http://www.miktex.org/>
- Installare un editor di testo che conosca la sintassi LaTeX:
 - Emacs: <http://ftp.gnu.org/pub/gnu/emacs/windows/>
 - TeXnic Center: <http://www.toolscenter.org/>
- Installare GhostScript e GhostView da <http://www.cs.wisc.edu/~ghost/>

Utenti Mac

- Per vecchi sistemi operativi:
<http://www.esm.psu.edu/mac-tex/>
- Per Mac OS X:
<http://ii2.sourceforge.net/>
(installer grafico + Ghost per Mac)
- Emacs per Mac:
<http://www.emacswiki.org/cgi-bin/wiki/EmacsForMacOS>
- Ghost per Mac:
<http://www.cs.wisc.edu/~ghost/macos/>

Problemi Tecnici?

- Il Camelug offre assistenza
 - Forum
 - Email sulla lista
 - Cd di installazione

Cominciamo

- Tutto il nostro onere è produrre un file di testo con un nome la cui estensione sia .tex?
- Cosa ci va scritto dentro?
- Il testo del nostro documento + il “codice” LaTeX che indica la struttura del documento stesso + comandi LaTeX per inserire formule, figure,

Quasi programmare

- Nella scrittura del nostro file sorgente
 - Ordine di idee simile al programmare in un linguaggio di programmazione
 - Ricordiamoci sempre che LaTeX è un programma
 - Siamo liberissimi nel testo, MA
 - Tutto ciò che deve essere interpretato dal LaTeX ha una struttura precisa
 - Se non conosciamo/seguiamo le regole **precisamente** addio tipografo automatico :-))

Macro struttura del file

1. Preambolo:

- Dichiarazione di uso di Packages
- Definizione di comandi
- Meta-Informazioni che riguardano il documento

2. Documento:

- Capitoli, sezioni, paragrafi, appendici
- Figure, tabelle
- Bibliografia

Preambolo

- Va dall'inizio del file al comando
`\begin{document}`
- Inizia con la dichiarazione del tipo del documento:
`\documentclass{book}`

Documento

- Inizia dal comando `\begin{document}`
- Termina con il comando `\end{document}`
- Tutto ciò che segue `\end{document}` non viene preso in considerazione

Preambolo: tipi di documenti

- La prima riga di un file .tex (a meno di righe di commenti) deve essere

`\documentclass[opzioni]{classe}`

- Tutte le distribuzioni forniscono almeno tre classi:
 - `book` : libro
 - `article` : articolo
 - `report` : relazione

Opzioni

- Le opzioni, se si vogliono specificare, vanno inserite tra parentesi quadre separate da virgole senza spazi. Le più usate:
- **11pt** o **12pt**: il font sarà di 11 o 12 punti (10 di default)
- **twoside**: impaginazione per stampa fronte-retro
- **twocolumn**: testo su due colonne per pagina

Tipo del documento

- La dichiarazione del `\documentclass` influenza la formattazione di tutto il documento
- Un libro segue regole diverse da quelle di un articolo o di una relazione
- Un giornale, un editore, un'organizzazione può fornire la sua particolare classe LaTeX (file nome-classe.cls)
- Basta inserire il nome di questa classe per ottenere un documento stilisticamente uniforme con quelli del fornitore!

Tutto qui?

- Spesso l'unica cosa di cui l'autore si deve preoccupare è inserire il nome giusto di classe
- I comandi LaTeX sono indipendenti dal tipo di classe scelto
- Essi verranno interpretati nella maniera più coerente possibile con quella del tipo di documento e la formattazione avverrà di conseguenza

Altra roba da preambolo

- Dichiarazioni di uso di Packages
- Definizione di comandi (macro)
- Meta-dati del documento

Packages

- Sono delle specie di plugin
- Forniscono delle funzionalità in più rispetto a quelle di base del LaTeX
- Per poter essere usati devono essere stati installati
- Si dichiarano con:

```
\usepackage[opzioni]{nome-package}
```

Babel: un package utile

- Fornisce la localizzazione linguistica
- Indicatori delle parti del testo: Chapter o Capitolo? Bibliography o Bibliografia?
- Divisione in sillabe (hyphenation) per andare a capo: regole dell'inglese diverse da quelle per l'italiano!
- Lingua di default di LaTeX: Inglese
- Se si scrive in italiano:
`\usepackage[italian]{babel}`

Inputenc: lettere accentate

- Nelle impostazioni di default il LaTeX non riconosce le lettere accentate
- Il pacchetto **inputenc** permette di inserire nel testo lettere accentate e fa in modo che queste vengano interpretate correttamente
- Bisogna inserire nelle opzioni il tipo di codifica di carattere che si usa.

Parentesi: codifica dei caratteri

- La codifica standard che LaTeX si aspetta è quella ASCII a 7 bit
- In generale una codifica associa ad ogni carattere un numero intero
- Nella codifica ASCII a 7 bit ci sono 128 caratteri codificati da 0 a 127
- In questi 128 non ci sono le lettere accentate
- Per risolvere questo problema sono state create altre codifiche

ISO 8859-1 o ISO Latin 1

- E' la codifica standard per gli alfabeti dei paesi dell'Europa occidentale
- Codifica 256 caratteri (8 bit)
- I primi 128 sono uguali a quelli della codifica ASCII a 7 bit
- I restanti contengono tutte le lettere accentate delle lingue dell'Europa occidentale

ISO 8859-15 o ISO Latin 9

- Uguale a ISO 8859-1 o ISO Latin 1 per tutti i caratteri tranne alcune ridefinizioni
- Tra queste quella più importante è la ridefinizione del carattere numero 164
- Nella versione 15 o 9 tale carattere è il simbolo dell'euro €

Codifica e salvataggio file

- Quando si usano lettere accentate l'editor di file di testo decide quale codifica usare per salvarle sul file
- E' possibile in quasi tutti gli editor di testi decidere la codifica da utilizzare
- Il modo più semplice è impostare la codifica ISO Latin 1 (o 9) per il salvataggio
- Di solito è l'opzione di default

Package inputenc

- Se abbiamo specificato nell'editor la codifica ISO Latin 1 (o 9)
- Inseriamo nell'opzione del pacchetto latin1 (o latin9):

```
\usepackage[latin1]{inputenc}
```

- oppure

```
\usepackage[latin9]{inputenc}
```

inputenc

- In questo modo LaTeX riconoscerà anche le lettere accentate
- Vedremo più avanti anche un modo alternativo (ma più macchinoso) per ottenere le lettere accentate
- Si potrà usare ad esempio per accentare le lettere maiuscole

Package per figure

- Per inserire figure di diverso tipo:
 - .eps (Encapsulated PostScript)
 - .gif
 - .png
 - .pdf
- Graphics: va bene per .eps, fornisce comandi per la scalatura
- Graphix: il più usato. Gestisce anche .pdf o .png
`\usepackage{graphics}`

Documentazione

- Ogni package ha associata la sua documentazione che spiega
 - Tutti i comandi forniti con relativa sintassi di uso
 - Le opzioni possibili in fase di dichiarazione
- Quando parleremo dell'inserimento di figure approfondiremo parte della documentazione di graphics e graphix

Definizione di comandi

- Stesso principio delle definizioni di

- Costanti
- Funzioni
- Macro

nei linguaggi di programmazione

- Vedremo la sintassi e alcuni esempi più avanti

Meta-dati

- Titolo:

`\title{testo del titolo}`

- Autore o autori:

`\author{Nome \and AltroNome`
`\and ... }`

- Data

`\date{data}`

Iniziamo il documento

- LaTeX inizia a produrre output a partire dal carattere inserito dopo

```
\begin{document}
```

- Che tipo di caratteri possiamo scrivere?
- Dobbiamo seguire delle regole di indentazione?

Il titolo

- In genere il primo comando dopo `\begin{document}` è il comando per inserire il titolo:
`\maketitle`
- In un book, il titolo prenderà una pagina a parte
- In un articolo prenderà solo l'inizio della prima pagina
- Deleghiamo queste questioni alla classe scelta e a LaTeX

E dopo?

- Iniziamo a scrivere il nostro testo! :-))
- Intanto vediamo i caratteri ammessi
- Poi vedremo come strutturare il testo in capitoli, sezioni, ecc.
- Poi iniziamo a vedere alcuni dettagli di comandi, casi particolari ecc.
- Nelle lezioni successive generalizzeremo le nozioni di comando, environment, blocco, ...

Caratteri Ammessi

- Tutte le lettere dell'alfabeto latino a-z, A-Z
- Tutte le cifre 0-9
- Punteggiatura: . : ; , ? ! ' ` () [] - / * @
- Spazio, Tab, “newline”
- Le lettere accentate sono ammesse solo se si usa il pacchetto inputenc
- Altri caratteri del set ASCII a 7 bit sono ammessi, ma sono speciali

Caratteri Speciali

- Servono per definire i comandi, delimitare gli argomenti dei comandi, marcare le celle di una tabella, ...
- Sono: # \$ % & ~ _ ^ \ { }
- Inseriti da soli nel documento non vengono interpretati come loro stessi, ma come (parti di) “tag”
- Per ottenerli come caratteri basta quotarli: \# \\$ \% \& \~ _ \^ \\ \{ \}

Tastiera Italiana

- I caratteri { } ~ ` non sono presenti sulla tastiera italiana
- Tuttavia, in particolare { e }, sono molto frequenti nel codice LaTeX
- In base al sistema operativo che stiamo usando abbiamo diversi modi di produrli

Linux: { } ~ `

- In ambiente Linux di solito:
- $\text{Alt Gr} + 7 = \{$
- $\text{Alt Gr} + 0 = \}$
- $\text{Alt Gr} + \grave{\text{i}} = \sim$
- $\text{Alt Gr} + ' = `$

Windows: { } ~ `

- In Windows di solito:
- Alt Gr + Shift + è = {
- Alt Gr + Shift + + = }
- Per la tilde (~) l'unico modo è usare il tastierino numerico
- Tenere premuto Alt sinistro e digitare 126 sul tastierino numerico (nei portatili questo viene simulato premendo prima un tasto funzione di tipo NumLock)

Windows: { } ~ `

- In alcune versioni l'apice rovesciato si ottiene con Alt Gr + Shift + ' oppure con Alt Gr + Ctrl + '
- Se non dovesse funzionare installare sia la tastiera italiana che quella inglese
- Tramite Alt + Shift Sinistro si può passare da una tastiera all'altra
- Settando la tastiera inglese, ` si ottiene premendo il tasto corrispondente a \
- Quindi: Alt + Shift Sinistro **rilasciare** premere \ **rilasciare** Alt + Shift Sinistro

Indentazione

- Per LaTeX valgono le stesse regole di un linguaggio di programmazione
- Non importa l'indentazione del codice sorgente
- In teoria (non utilizzando paragrafi) potremmo scrivere tutto il file su una sola riga!!
- Tuttavia è consigliabile usare un certo stile anche nel sorgente, per aiutarci nella rilettura, correzione, aggiornamento

Il concetto di blank

- Per separare le parole c'è bisogno di almeno un “blank”
- LaTeX interpreta con **un** “blank” una sequenza qualsiasi, lunga almeno uno, di caratteri:
 - Spazio
 - Tab
- Tale sequenza può contenere anche qualsivoglia caratteri “newline” (a capo)
non consecutivi

Il concetto di blank

- Quindi scrivere

La vispa Teresa

- Oppure

La

vispa

Teresa

- Produce lo stesso risultato

Commenti

- LaTeX ignora qualsiasi carattere che segue un `%` fino alla fine della linea
- In questo modo possiamo inserire nel file dei commenti che non appariranno nel documento
- Sono utili per inserire note per noi stessi, per documentare le motivazioni della struttura scelta, per inserire delle intestazioni con autore, copyright, versione...

Paragrafi

- LaTeX considera un paragrafo un qualsiasi blocco di parole separate da un blank
- Un paragrafo termina quando c'è almeno una riga vuota (cioè almeno due caratteri newline consecutivi)
- Più linee vuote equivalgono ad una

Paragrafi

Questo è un paragrafo
anche
se
vado a capo.

Ora che ho lasciato una linea
vuota sono in un altro
paragrafo!

Paragrafi e indentazione

- La prima riga di un nuovo paragrafo viene indentata di un certo spazio
- Questo non avviene se il paragrafo è il primo di una sezione, capitolo, sottosezione, ...
- Per evitare l'indentazione si può usare il comando `\noindent`
- Va inserito come **primo** comando in un nuovo paragrafo

Sottosezioni, sezioni, capitoli

- In un testo i paragrafi vengono raggruppati in sottosezioni
- Le sottosezioni vengono raggruppate in sezioni
- Le sezioni vengono raggruppate in capitoli
- I capitoli possono venire raggruppati in parti
- Alcuni capitoli o sezioni possono essere considerati delle appendici

Struttura del testo in LaTeX

- Per iniziare una parte si utilizza il comando

`\part{titolo-parte}`

- All'interno di una parte possono essere dichiarati diversi capitoli con

`\chapter{titolo-capitolo}`

- All'interno di un capitolo possono essere dichiarate sezioni con

`\section{titolo-sezione}`

Struttura del testo in LaTeX

- All'interno di una sezione possono essere dichiarate sottosezioni con `\subsection{ titolo-sottosezione }`
- All'interno di una sottosezione si possono dichiarare sottosottosezioni con `\subsubsection{ titolo-sottosottosezione }`
- Ulteriori sezionamenti non sono possibili
- Il comando `\chapter` non è disponibile nella classe `article`

Numerazione

- LaTeX numera automaticamente e coerentemente i vari tipi di strutture
- Se cambiamo la struttura logica i numeri verranno ricalcolati automaticamente
- Lo stile di visualizzazione dei titoli e il tipo di numerazione viene decisa in base al tipo di documento
- Con il comando `\tableofcontents` LaTeX genera automaticamente l'indice di tutto il documento!!

Appendici

- Se ad un certo punto nel testo inseriamo il comando

\appendix

- Nel seguito le strutture saranno numerate come si conviene alle appendici
- Ad esempio i capitoli diventano Appendice A, B, ...

Esempio

...

```
\chapter{Mammiferi}
```

In questo capitolo parleremo dei mammiferi.

Ogni specie verrà suddivisa in
sottospecie...

```
\section{Felini}
```

I felini sono animali molto affascinanti ...

```
\subsection{Gatti}
```

Miao, il tipico verso dei gatti, è ...

```
\section{Equini}
```

...

```
\chapter{Uccelli}
```

...

Lettere accentate

- LaTeX è stato scritto avendo in mente la produzione di testi in lingua inglese
- L'inglese non usa caratteri accentati
- Per produrre le lettere accentate bisogna utilizzare il package `inputenc` oppure dei codici che possono risultare macchinosi

Lettere accentate

- Per le normali accentate la cosa migliore è senz'altro usare il package **inputenc**
- Per tutto il resto vediamo quali sono i comandi per ottenere gli accenti con LaTeX di base

Lettere accentate

- Per ottenere lettere accentate con accento **grave** (à è ì ò ù) bisogna utilizzare il comando `\`` seguito dalla lettera
- `\`a` produce à
- `\`e` produce è
- Serve soprattutto per le maiuscole:
- `\`A` produce À

Lettere accentate

- Per ottenere l'accento **acuto** il comando è lo stesso, ma con l'apice normale
- \ ' e produce é
- \ ' E produce É
- Nota: la “è” grave corrisponde al suono aperto in cioè, è, ...
- Nota: la “é” acuta corrisponde al suono chiuso in perché, sé, benché, ...

Lettere accentate

- In realtà il comando completo sarebbe del tipo `\`{stringa}`
- Nel caso di un solo carattere da accentare è prevista l'abbreviazione che abbiamo visto
- Esistono anche altri comandi simili per ottenere accenti diversi

- Ad esempio

`\u{c}` = č

`\^{\i}` = î

Punteggiatura

- La punteggiatura va inserita attaccandola alla parola che la precede e facendola seguire da uno spazio
- SI: Ciao, come stai?
- NO: Ciao , come stai?

Fine periodo

- Il punto, il punto esclamativo e quello interrogativo in genere terminano dei periodi.
- Quando questo accade andrebbe lasciato uno spazio maggiore
- LaTeX, essendo un programma, non sa decidere se un punto termina un periodo oppure indica ad esempio un'abbreviazione: ecc.

Fine periodo

- LaTeX utilizza questa regola:
Se la lettera che precede il punto è minuscola allora il punto termina un periodo, altrimenti no
- Lo spazio maggiore viene inserito di conseguenza
- Nella maggior parte dei casi è giusto
- Quando non va bene dobbiamo indicare noi la corretta interpretazione

Fine periodo

- Ad esempio in “I gatti, le tigri, i leoni ecc. sono felini”
- Il punto di `ecc.` non termina il periodo!
- Per evitare che LaTeX inserisca spazio extra dopo di esso va inserito `\` (backslash seguito da uno spazio) dopo il punto:

I gatti, le tigri, i leoni
`ecc. \` sono felini

Fine periodo

- In “I romani scrivevano I + II = III. Simpatici!”
- Il punto dopo III termina il periodo!
- Per far inserire lo spazio extra dovuto basta inserire il comando \@ prima del punto:
- I romani scrivevano I + II = III\@. Simpatici!

Fine periodo

- Inoltre quando al punto segue una parentesi chiusa o virgolette chiuse si applicano le stesse regole:
- `"Gli agrumi (arance, limoni ecc.) \ sono ricchi di vitamina C \@."`
- Tutto si applica allo stesso modo anche a `? ! e :`

Trattini

- Un meno indica a LaTeX di inserire un trattino fra parole:

Unità aritmetico-logica

- Due trattini inseriscono un trattino lungo che indica un intervallo:

Nelle pagine 2--5

- Tre trattini inseriscono un trattino lungo che indica incisi o discorso diretto:

--- Ciao, come stai?

--- Bene, grazie.

Alcuni comandi utili

- Alcuni comandi producono un certo testo
- `\LaTeX` e `\TeX` producono i rispettivi loghi. E' bene farli seguire da uno spazio ordinario tra parole (backslash seguito da uno spazio)
- `\today` produce la data odierna
- `\ldots` produce i puntini di sospensione

Testo enfatizzato

- Il font con cui viene scritto il testo è deciso dalla classe del documento
- Quando si vogliono mettere in evidenza certe parole in genere si usa il corsivo o il grassetto
- In LaTeX basta indicare che il testo è enfatizzato e la sua resa sarà data automaticamente da quanto previsto nella classe

Testo enfattizzato

- Per indicare che un certo testo è enfattizzato si usa la dichiarazione `\em`

```
{\em testo-enfattizzato}
```

I cani {\em non} sono felini!

- “em” sta per emphasized
- Viene enfattizzato solo il testo tra le parentesi graffe

Testo enfatizzato

- Quando il testo enfatizzato o quello normale è scritto in corsivo c'è un problema quando si passa dall'uno all'altro senza spazi intermedi
- No, I {\em did}n't!
- Produce

No, I *didn't*!

Testo enfattizzato

- Per aggiungere un piccolo spazio in modo da evitare la sovrapposizione si usa il comando `\/`
- `No, I {\em did\/}n't!`
- Produce

No, I *didn't*!

Note a fine pagina

- Per inserire una nota si usa il comando `\footnote{testo}`
- `Fuffi\footnote{Il mio gatto.}`
di solito preferisce il
pesce.

Fuffi¹ di solito preferisce il pesce.

...

¹Il mio gatto.

Interruzioni di riga e pagina

- LaTeX indenta automaticamente i paragrafi andando a capo e cambiando pagina nel punto ottimale
- Utilizza anche la sillabazione (hyphenation) per spezzare le parole
- In genere non ci deve interessare intervenire sulle interruzioni di riga o pagina
- Ci sono però delle eccezioni

Interruzioni di riga e pagina

- Non sono opportune
- In un nome con un titolo: “Il Dr. Rossi” dopo Dr.
- In un riferimento: “In Figura 7” dopo Figura
- ...

Evitare interruzioni

- Si usa il comando `~` invece dello spazio:
- `"Il Dr.~Rossi"`
- `"In Figura~7"`
- E' bene non esagerare con i `~`
- In caso contrario LaTeX potrebbe non trovare più un posto dove andare a capo!

Virgolette

- Il carattere “ (sopra al 2 nella tastiera italiana) non viene interpretato come ci si aspetta
- Per ottenere un testo tra virgolette vanno inseriti due apici rovesciati all'inizio e due apici normali alla fine:

`` `Fra virgolette' '`

Un esempio di environment

- Per evidenziare un testo, ad esempio una citazione, LaTeX mette a disposizione un **environment**

```
\begin{quote}  
Testo in evidenza  
\end{quote}
```

- Gli environment sono un costrutto standard di LaTeX, ne vedremo molti

Compilare

- A questo punto abbiamo abbastanza nozioni per scrivere un file .tex di prova abbastanza complesso
- Proviamo a vedere cosa viene fuori dal file CamelugLatexBookSample.tex
- La compilazione avviene a linea di comando
- In emacs ed editor predisposti può essere fatta anche dall'interno

Step by Step

- Aprire una finestra di shell (in Windows Prompt dei Comandi)
- Portarsi con il comando `cd` (change directory) nella directory (cartella) che contiene il file da compilare
- Dare il comando:

```
> latex CamelugLatexBookSample.tex
```

Risultato

```
This is TeX, Version 3.14159 (MiKTeX 2.2)
(CamelugLatexBookSample.tex
LaTeX2e <2001/06/01>
Babel <v3.7h> and hyphenation patterns for english, french, german, ngerman, du
mylang, nohyphenation, loaded.
(C:\AddOn\texmf\tex\latex\base\book.cls
Document Class: book 2001/04/21 v1.4e Standard LaTeX document class
(C:\AddOn\texmf\tex\latex\base\bk10.clo))
(C:\AddOn\texmf\tex\generic\Babel\babel.sty
(C:\AddOn\texmf\tex\generic\Babel\italian.ldf
(C:\AddOn\texmf\tex\generic\Babel\babel.def)

Package babel Warning: No hyphenation patterns were loaded for
(babel)                the language `Italian'
(babel)                I will use the patterns loaded for \language=0 instead.

)) (C:\AddOn\texmf\tex\latex\graphics\graphics.sty
(C:\AddOn\texmf\tex\latex\graphics\trig.sty)
(C:\AddOn\texmf\tex\latex\00miktex\graphics.cfg)
(C:\AddOn\texmf\tex\latex\graphics\dvips.def)) (CamelugLatexBookSample.aux)
(C:\AddOn\texmf\tex\latex\cyrillic\ot2cmr.fd) [1] [2]
(CamelugLatexBookSample.toc) [3] [4]
Capitolo 1.
[5] [6]
Capitolo 2.
[7] [8]
Appendice A.
[9] (CamelugLatexBookSample.aux) )
Output written on CamelugLatexBookSample.dvi (9 pages, 7044 bytes).
Transcript written on CamelugLatexBookSample.log.
```

E ora?

- La compilazione è andata bene
- Nella stessa cartella sono stati creati dei file con lo stesso nome del .tex, ma con estensioni diverse
- .log: contiene tutto l'output che è stato stampato a video durante la compilazione
- .toc .aux: ausiliari, non ci interessano
- .dvi: **Device Independent File**

DVI

- Il formato dvi è una descrizione intermedia del documento
- Può essere visualizzato con un viewer
- Su Unix/Linux: `> xdvi nome-file.dvi`
- Su Windows: Yap dvi Viewer fornito con MikTeX
- Su Mac: ?
- Spesso il viewer non visualizza le immagini .eps o .pdf inserite nel documento

Da dvi a PostScript

- Se il preview del documento .dvi va bene possiamo passare a generare il PostScript
- Dal prompt dei comandi/shell

```
> dvips -o nome-file.ps nome-file.dvi
```

- Viene generato il file .ps che può essere visualizzato o stampato tramite GhostView

E se vogliamo il pdf?

- Tutte le implementazioni recenti di TeX/LaTeX mettono a disposizione **pdf_latex**
- E' un compilatore che produce direttamente il file .pdf senza passare per il .dvi
- Funziona esattamente come **l_atex**:

```
> pdflatex nomefile.tex
```

PDF da DVI

- Un'alternativa è generare il .pdf dal .dvi
- Si usa il comando **dvipdfm**

```
> dvipdfm -o nomefile.pdf nomefile.dvi
```

Compilare due volte

- Se nel testo ci sono riferimenti (li vedremo) e viene generato l'indice
- LaTeX ha bisogno di due passate per generare correttamente le numerazioni
- In questo caso viene stampato un messaggio: “Rerun latex to get cross reference right”

L'imprevisto in agguato

- Purtroppo non sempre la compilazione va a buon fine
- Come per un linguaggio di programmazione qualsiasi il compilatore LaTeX può trovare errori
- Essi vengono segnalati in output durante la compilazione
- Normalmente la compilazione è interattiva

Debug

- Proviamo a inserire `\begin{quot}` invece di `\begin{quote}` nel file di esempio.
- Durante la compilazione compare:

`! LaTeX Error: Environment quot undefined.`

`See the LaTeX manual or LaTeX Companion for
explanation.`

`Type H <return> for immediate help.`

`...`

`1.62 \begin{quot}`

`?`

Che roba è?

- La riga

`! LaTeX Error: Environment quot undefined.`

indica che LaTeX ha trovato un errore

- Siccome anche TeX gira durante la compilazione potrebbe essere lui a trovare errori!
- Dopo i due punti c'è la descrizione dell'errore trovato. **In questo caso** chiara e giusta :-))

Che roba è?

- La riga

1.62 `\begin{quot}`

- Indica il numero di riga dove è stato trovato l'errore e la riporta
- Il ? che segue è un prompt
- LaTeX si aspetta una risposta da parte nostra...

Niente panico

- Se si è capito l'errore
 - Correggere il sorgente .tex
- Se si vuole continuare col prossimo errore premere invio (alcuni errori potranno derivare da questo)
- Altrimenti si può interrompere
 - Premere “q” + Invio per uscire: LaTeX genera l'output fino al punto dell'errore
 - Premere “x” + Invio per uscire brutalmente e non generare nessun output.

Riprovare

- **Salvare** le modifiche fatte al file .tex
- Provare a ricompilare
- Ripetere fino a quando la compilazione ha successo
- Ciclo infinito? :-)))

Errori TeX

- Se non appare
! LaTeX error
significa che è stato TeX a trovare
l'errore!!
- In questo caso siamo nei guai :-))
- Noi non conosciamo TeX!
- In genere una cosa così accade quando
abbiamo dato parametri sbagliati a
qualche comando LaTeX

Warning

- Durante la compilazione possono apparire anche altri messaggi
- Un esempio è quello dei warning
- Segnalano problemi che non vengono considerati veri e propri errori
- Tuttavia andrebbero di solito risolti
- La compilazione non si ferma per chiedere il nostro intervento
- Si può guardare il file.log per correggerli

Overfull

- Scriviamo questa frase nel nostro file di esempio:

Mathematical formulas may also be displayed. A displayed formula is gnomonly.

- Se compiliamo otteniamo il warning:

```
Overfull \hbox (5.79588pt too wide) in paragraph at lines 71--73
```

```
[ ]\OT1/cmr/m/n/10 Mathematical for-mu-las  
may al-so be dis-played. A dis-played  
for-mu-la is gnomon-
```

Overfull??

- Significa che LaTeX non è riuscito a trovare un posto adatto per andare a capo
- La stringa gnomon- sporge un po' dal margine della pagina
- Non è grave
- Un modo per correggerlo è quello di dire a LaTeX dove possiamo andare a capo nella parola gnomonly
- Basta scriverla come `gno\-monly`