Introduzione al LaTeX

Seconda lezione

Daniele Serra

15 Marzo 2014

Nella puntata precedente

Abbiamo:

Nella puntata precedente

Abbiamo:

- imparato a scrivere un primo documento con LaTeX :
  - `\documentclass{}`
  - `\begin{document} ... \end{document}`

Nella puntata precedente

Abbiamo:

- imparato a scrivere un primo documento con LaTeX :
  - `\documentclass{}`
  - `\begin{document} ... \end{document}`
- distinto il *preambolo* dal *corpo* del documento;

Nella puntata precedente

Abbiamo:

- imparato a scrivere un primo documento con LaTeX :
  - `\documentclass{}`
  - `\begin{document} ... \end{document}`
- distinto il *preambolo* dal *corpo* del documento;
- imparato i primi comandi (`\emph{}`, `\textbf{} ...`);

Nella puntata precedente

Abbiamo:

- imparato a scrivere un primo documento con LaTeX :
  - `\documentclass{}`
  - `\begin{document} ... \end{document}`
- distinto il *preambolo* dal *corpo* del documento;
- imparato i primi comandi (`\emph{}`, `\textbf{} ...`);
- scrivere formule matematiche *in linea* (`$...$`) o *in display* (`\[...\]`);

Nella puntata precedente

Abbiamo:

- imparato a scrivere un primo documento con LaTeX :
  - `\documentclass{}`
  - `\begin{document} ... \end{document}`
- distinto il *preambolo* dal *corpo* del documento;
- imparato i primi comandi (`\emph{}`, `\textbf{} ...`);
- scrivere formule matematiche *in linea* (`$...$`) o *in display* (`\[...\]`);
- visto cosa sono i pacchetti e alcuni ambienti di testo.

Matrici

Un modo conveniente di scrivere matrici è usare l'ambiente `pmatrix` del pacchetto `amsmath`.
Si scrive la matrice per righe, con elementi separati da `&`. Per andare a capo, si usa `\\`.

Matrici

Un modo conveniente di scrivere matrici è usare l'ambiente `pmatrix` del pacchetto `amsmath`.
Si scrive la matrice per righe, con elementi separati da `&`. Per andare a capo, si usa `\\`.

```
\[
 A=
      \begin{pmatrix}
      -1 &  2 &  0 \\
       2 & -1 &  2 \\
       0 &  2 & -1
      \end{pmatrix}.
\]
```

$$A = \begin{pmatrix} -1 & 2 & 0 \\ 2 & -1 & 2 \\ 0 & 2 & -1 \end{pmatrix}.$$

Matrici, un paio di aggiunte

1. Se non vi piacciono le parentesi tonde, potete usare le:

   quadre , con `bmatrix`;
   graffe , con `Bmatrix`;
   barre verticali , con `vmatrix`.

Matrici, un paio di aggiunte

1. Se non vi piacciono le parentesi tonde, potete usare le:

   quadre , con `bmatrix`;
   graffe , con `Bmatrix`;
   barre verticali , con `vmatrix`.

2. È fondamentale saper fare i puntini:
   - `\cdots` fa i puntini orizzontali $\cdots$
   - `\vdots` fa i puntini verticali $\vdots$
   - `\ddots` fa i puntini diagonali $\ddots$

Spezzare le formule

Molto spesso ci si ritrova a scrivere formule molto lunghe che, per questioni di leggibilità, è opportuno mettere su più righe.
L'ambiente `align` di `amsmath` ci viene incontro.

- Occorre specificare con \\ il punto in cui vogliamo spezzare la formula;

Spezzare le formule

Molto spesso ci si ritrova a scrivere formule molto lunghe che, per questioni di leggibilità, è opportuno mettere su più righe.
L'ambiente `align` di `amsmath` ci viene incontro.

- Occorre specificare con \\ il punto in cui vogliamo spezzare la formula;
- si usano gli & per allineare le righe.

Spezzare le formule

Molto spesso ci si ritrova a scrivere formule molto lunghe che, per questioni di leggibilità, è opportuno mettere su più righe.
L'ambiente `align` di `amsmath` ci viene incontro.

- Occorre specificare con \\ il punto in cui vogliamo spezzare la formula;
- si usano gli & per allineare le righe.

L'ambiente `align` numera ogni riga: se vogliamo escludere qualche riga, basta farla seguire da `\notag`.

Spezzare le formule

Un esempio

Vogliamo ottenere la seguente formula:

$$\begin{aligned} f(x) &= \sin x \\ &= \sin(x + 2\pi). \end{aligned} \tag{1}$$

Spezzare le formule

Un esempio

Vogliamo ottenere la seguente formula:

$$\begin{aligned} f(x) &= \sin x \\ &= \sin(x + 2\pi). \end{aligned} \tag{1}$$

Il codice giusto è

```
\begin{align}
 f(x)  &=\sin x \\
       &=\sin(x+2\pi). \notag
\end{align}
```

Spezzare le formule

Un esempio

Vogliamo ottenere la seguente formula:

$$\begin{aligned} f(x) &= \sin x \\ &= \sin(x + 2\pi). \end{aligned} \tag{1}$$

Il codice giusto è

```
\begin{align}
 f(x)  &=\sin x \\
       &=\sin(x+2\pi). \notag
\end{align}
```

Attenzione: l'ambiente `align` non va inserito in ambiente matematico: è lui stesso un ambiente matematico.

Comandi senza argomenti

È possibile personalizzare LaTeX creando nuovi comandi.
Ad esempio, se si volesse scrivere un libro di Algebra Lineare, si dovrebbe scrivere spesso `\mathbb{K}` per ottenere il simbolo di campo $\mathbb{K}$. Troppo lungo!
Vediamo come dire a LaTeX di fare la stessa operazione, ma con un'espressione più breve (diciamo `\K`).

Comandi senza argomenti

È possibile personalizzare LaTeX creando nuovi comandi.
Ad esempio, se si volesse scrivere un libro di Algebra Lineare, si dovrebbe scrivere spesso `\mathbb{K}` per ottenere il simbolo di campo $\mathbb{K}$. Troppo lungo!
Vediamo come dire a LaTeX di fare la stessa operazione, ma con un'espressione più breve (diciamo `\K`).
Nel *preambolo*, basta dare l'istruzione

```
\newcommand{\K}{\mathbb{K}}
```

Comandi senza argomenti

È possibile personalizzare LaTeX creando nuovi comandi.
Ad esempio, se si volesse scrivere un libro di Algebra Lineare, si dovrebbe scrivere spesso `\mathbb{K}` per ottenere il simbolo di campo $\mathbb{K}$. Troppo lungo!
Vediamo come dire a LaTeX di fare la stessa operazione, ma con un'espressione più breve (diciamo `\K`).
Nel *preambolo*, basta dare l'istruzione

```
\newcommand{\K}{\mathbb{K}}
```

In generale, la sintassi è

```
\newcommand{nome comando}{definizione comando}
```

Comandi con argomenti

In generale, un'istruzione può ammettere diversi argomenti (la norma ne ha uno, il prodotto scalare ne ha due..).

Comandi con argomenti

In generale, un'istruzione può ammettere diversi argomenti (la norma ne ha uno, il prodotto scalare ne ha due..).
Per ottenere l'output $\|x\|$, si può digitare `$\lVert x \rVert$`.

Comandi con argomenti

In generale, un'istruzione può ammettere diversi argomenti (la norma ne ha uno, il prodotto scalare ne ha due..).
Per ottenere l'output $\|x\|$, si può digitare `$\lVert x \rVert$`.
Sarebbe comodo avere un comando ad un parametro di nome `\norma{}`. Con LaTeX ciò è possibile!

Comandi con argomenti

In generale, un'istruzione può ammettere diversi argomenti (la norma ne ha uno, il prodotto scalare ne ha due..).
Per ottenere l'output $\|x\|$, si può digitare `$\lVert x \rVert$`.
Sarebbe comodo avere un comando ad un parametro di nome `\norma{}`. Con LaTeX ciò è possibile! Nel *preambolo*,

```
\newcommand{\norma}[1]{\lVert #1 \rVert}
```

Comandi con argomenti

In generale, un'istruzione può ammettere diversi argomenti (la norma ne ha uno, il prodotto scalare ne ha due..).
Per ottenere l'output $\|x\|$, si può digitare `$\lVert x \rVert$`.
Sarebbe comodo avere un comando ad un parametro di nome `\norma{}`. Con LaTeX ciò è possibile! Nel *preambolo*,

```
\newcommand{\norma}[1]{\lVert #1 \rVert}
```

In generale, la sintassi è

```
\newcommand{nome}[numero argomenti]{definizione}
```

dove, nella definizione, si scrive `#k` al posto del k-esimo argomento ($k \leq 9$).

Vantaggi

Si supponga di dover cambiare, nel nostro documento, il simbolo $\|x\|$ in $|x|$. Bisognerebbe farlo manualmente! Con l'uso di `\newcommand`, basta cambiarlo *una volta*, nella dichiarazione del comando `\norma`. Basta modificare l'istruzione data prima:

```
\newcommand{\norma}[1]{\lVert #1 \rVert}
```

in

```
\newcommand{\norma}[1]{\lvert #1 \rvert}
```

e automaticamente cambierà in tutto il documento.

Riassegnazione di comandi

Esempio

I comandi `\tilde` e `\widetilde` mettono una tilde sopra la parola che segue:

- `\tilde{G}` $\rightarrow \tilde{G}$;
- `\widetilde{G}` $\rightarrow \widetilde{G}$.

Riassegnazione di comandi

Esempio

I comandi `\tilde` e `\widetilde` mettono una tilde sopra la parola che segue:

- `\tilde{G}` $\rightarrow \tilde{G}$;
- `\widetilde{G}` $\rightarrow \widetilde{G}$.

Con le maiuscole, può essere preferibile usare `\widetilde`, ma è troppo lungo. Come fare?

Riassegnazione di comandi

Esempio

I comandi `\tilde` e `\widetilde` mettono una tilde sopra la parola che segue:

- `\tilde{G}` $\rightarrow \tilde{G}$;
- `\widetilde{G}` $\rightarrow \widetilde{G}$.

Con le maiuscole, può essere preferibile usare `\widetilde`, ma è troppo lungo. Come fare?

Si può usare `\renewcommand`, che assegna ad un comando *già esistente* la funzione di un altro comando.

```
\renewcommand{\tilde}{\widetilde}
```

Riassegnazione di comandi

Esempio

I comandi `\tilde` e `\widetilde` mettono una tilde sopra la parola che segue:

- `\tilde{G}` $\rightarrow \tilde{G}$;
- `\widetilde{G}` $\rightarrow \widetilde{G}$.

Con le maiuscole, può essere preferibile usare `\widetilde`, ma è troppo lungo. Come fare?

Si può usare `\renewcommand`, che assegna ad un comando *già esistente* la funzione di un altro comando.

```
\renewcommand{\tilde}{\widetilde}
```

La sintassi è la stessa di `\newcommand`.

Ambienti-teorema

La matematica è una lista di definizioni, proposizioni e osservazioni: in un articolo matematico sarà fondamentale creare degli ambienti con questi scopi.

Ambienti-teorema

La matematica è una lista di definizioni, proposizioni e osservazioni: in un articolo matematico sarà fondamentale creare degli ambienti con questi scopi.
LaTeX dà la possibilità di creare degli *ambienti-teorema*: basta inserire nel preambolo il comando

```
\newtheorem{de}{Definizione}
```

e verrà creato un ambiente, chiamato *de*, che nell'output verrà stampato come *Definizione*.

Ambienti-teorema

La matematica è una lista di definizioni, proposizioni e osservazioni: in un articolo matematico sarà fondamentale creare degli ambienti con questi scopi.
LaTeX dà la possibilità di creare degli *ambienti-teorema*: basta inserire nel preambolo il comando

```
\newtheorem{de}{Definizione}
```

e verrà creato un ambiente, chiamato *de*, che nell'output verrà stampato come *Definizione*.
Più in generale, la sintassi è

```
\newtheorem{nome dell'ambiente}{titolo}
```

Stili diversi ad enunciati diversi

Quello che abbiamo detto prima, però, non basta: LaTeX gradisce la specifica dello stile desiderato per questi ambienti (tipo di carattere del titolo, tipo di carattere del corpo...).

Stili diversi ad enunciati diversi

Quello che abbiamo detto prima, però, non basta: LaTeX gradisce la specifica dello stile desiderato per questi ambienti (tipo di carattere del titolo, tipo di carattere del corpo...).

Per fortuna, in `amsthm` ne troviamo tre diversi preconfezionati:

plain Per Teoremi, Lemmi, Corollari, Proposizioni.

Stili diversi ad enunciati diversi

Quello che abbiamo detto prima, però, non basta: LaTeX gradisce la specifica dello stile desiderato per questi ambienti (tipo di carattere del titolo, tipo di carattere del corpo...).

Per fortuna, in `amsthm` ne troviamo tre diversi preconfezionati:

plain Per Teoremi, Lemmi, Corollari, Proposizioni.

definition Per Definizioni, Esempi.

Stili diversi ad enunciati diversi

Quello che abbiamo detto prima, però, non basta: LaTeX gradisce la specifica dello stile desiderato per questi ambienti (tipo di carattere del titolo, tipo di carattere del corpo...).

Per fortuna, in `amsthm` ne troviamo tre diversi preconfezionati:

plain Per Teoremi, Lemmi, Corollari, Proposizioni.

definition Per Definizioni, Esempi.

remark Per Osservazioni.

Stili diversi ad enunciati diversi
(continua)

Come dire a LaTeX che vogliamo assegnare un certo stile ad un certo ambiente?

Stili diversi ad enunciati diversi
(continua)

Come dire a LaTeX che vogliamo assegnare un certo stile ad un certo ambiente?
Nel *preambolo*, prima di `\newtheorem`, si scrive:

```
\theoremstyle{stile}
```

e tutti gli ambienti-teorema dichiarati in seguito (prima di un altro `\theoremstyle`) avranno quello stile.

Stili diversi ad enunciati diversi
(continua)

Come dire a LaTeX che vogliamo assegnare un certo stile ad un certo ambiente?
Nel *preambolo*, prima di `\newtheorem`, si scrive:

```
\theoremstyle{stile}
```

e tutti gli ambienti-teorema dichiarati in seguito (prima di un altro `\theoremstyle`) avranno quello stile. Per le dimostrazioni, esiste l'ambiente `proof`.

Un esempio

Un esempio

Definizione

Chiameremo LaTeX quel programma per creare documenti che sto usando adesso.

Un esempio

Definizione

Chiameremo LaTeX quel programma per creare documenti che sto usando adesso.

Teorema

LaTeX è il programma per creare documenti più fico che esista.

Un esempio

Definizione

Chiameremo LaTeX quel programma per creare documenti che sto usando adesso.

Teorema

LaTeX è il programma per creare documenti più fico che esista.

Dimostrazione.

È evidente. □

Figure

Per inserire una figura con LaTeX conviene usare l'ambiente `figure`, che serve a dire a LaTeX di ritagliare lo spazio per una figura e si lascia a lui l'onere di posizionarlo adeguatamente (questa cosa ha i suoi pro e contro).

Figure

Per inserire una figura con LaTeX conviene usare l'ambiente `figure`, che serve a dire a LaTeX di ritagliare lo spazio per una figura e si lascia a lui l'onere di posizionarlo adeguatamente (questa cosa ha i suoi pro e contro).
Con `\includegraphics[opzioni]{nomefigura}` (pacchetto `graphicx`) diciamo a LaTeX di inserire la figura di nome `nomefigura`.

Figure

Per inserire una figura con LaTeX conviene usare l'ambiente `figure`, che serve a dire a LaTeX di ritagliare lo spazio per una figura e si lascia a lui l'onere di posizionarlo adeguatamente (questa cosa ha i suoi pro e contro).
Con `\includegraphics[opzioni]{nomefigura}` (pacchetto `graphicx`) diciamo a LaTeX di inserire la figura di nome `nomefigura`. Attenzione:

1. La figura deve essere nella stessa cartella del file `.tex`

Figure

Per inserire una figura con LaTeX conviene usare l'ambiente `figure`, che serve a dire a LaTeX di ritagliare lo spazio per una figura e si lascia a lui l'onere di posizionarlo adeguatamente (questa cosa ha i suoi pro e contro).
Con `\includegraphics[opzioni]{nomefigura}` (pacchetto `graphicx`) diciamo a LaTeX di inserire la figura di nome `nomefigura`. Attenzione:

1. La figura deve essere nella stessa cartella del file `.tex`
2. Estensioni ammesse: `.jpg`, `.png`, `.pdf` ...

Figure

Per inserire una figura con LaTeX conviene usare l'ambiente `figure`, che serve a dire a LaTeX di ritagliare lo spazio per una figura e si lascia a lui l'onere di posizionarlo adeguatamente (questa cosa ha i suoi pro e contro).
Con `\includegraphics[opzioni]{nomefigura}` (pacchetto `graphicx`) diciamo a LaTeX di inserire la figura di nome `nomefigura`. Attenzione:

1. La figura deve essere nella stessa cartella del file `.tex`
2. Estensioni ammesse: `.jpg`, `.png`, `.pdf` ...
3. È possibile regolare le dimensioni dell'immagine con `scale`, ruotarla con `rotate`...

Didascalie, un esempio

È possibile inserire una didascalia nelle figure con il comando `\caption{}`.

Didascalie, un esempio

È possibile inserire una didascalia nelle figure con il comando `\caption{}`.

```
\begin{figure}
\begin{center}
\includegraphics{giove}
\caption{Il pianeta Giove}
\end{center}
\end{figure}
```

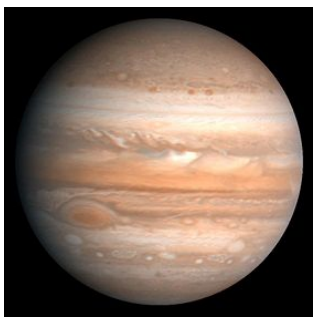

Figura : Il pianeta Giove

Tabelle

L'ambiente `table` ha la stessa funzione di `figure`, ma serve per le tabelle.

Tabelle

L'ambiente `table` ha la stessa funzione di `figure`, ma serve per le tabelle.
Esistono due tipi di tabelle:

- Prevalenza di testo: ambiente `tabular`

Tabelle

L'ambiente `table` ha la stessa funzione di `figure`, ma serve per le tabelle.
Esistono due tipi di tabelle:

- Prevalenza di testo: ambiente `tabular`
- Prevalenza di matematica: ambiente `array`

Tabelle di testo: un esempio

Output

Osserviamo la seguente tabella:

Prodotti	Prezzo (€)	Destinatario
Pane	2	Nonna
Latte	1.2	Gatto
Nutella	1.5	Me

Tabella : Lista della spesa

- 3 colonne: testo a sinistra, al centro, a destra
- Linee separatrici delle righe

Tabelle di testo: un esempio

Sorgente

```
\begin{table}
 \begin{center}
   \begin{tabular}{lcr} %Numero colonne e posizione del testo
    \toprule            %Prima riga
    Prodotti & Prezzo (\officialeuro) & Destinatario \\
    \midrule            %Riga di mezzo
    Pane     & $2$                   & Nonna        \\
    Latte    & $1.2$                 & Gatto        \\
    Nutella  & $1.5$                 & Me           \\
    \bottomrule         %Ultima riga
   \end{tabular}
\caption{Lista della spesa}
\end{center}
\end{table}
```

Le righe sono presenti nel pacchetto `booktabs`. Senza bisogno di pacchetti, `\hline` disegna una linea orizzontale.

Tabelle matematiche: un esempio

Funziona tutto come in `tabular`.

```
\begin{table}
 \begin{center}
\[
 \begin{array}{cc}
   \toprule
   \text{Funzione} & \text{Derivata} \\
   \midrule
   \sin x          & \cos x          \\
   x^n             & nx^{n-1}        \\
   \bottomrule
 \end{array}
\]
\caption{Un paio di derivate}
 \end{center}
\end{table}
```

Funzione	Derivata
$\sin x$	$\cos x$
x^n	nx^{n-1}

Tabella : Un paio di derivate

Un manuale di riferimento

Per preparare questo seminario mi sono ispirato alla guida di Lorenzo Pantieri & Tommaso Gordini,

L'arte di scrivere con LaTeX,

che trovate facilmente googlando "pantieri".

Un manuale di riferimento

Per preparare questo seminario mi sono ispirato alla guida di Lorenzo Pantieri & Tommaso Gordini,

L'arte di scrivere con LaTeX,

che trovate facilmente googlando "pantieri".
Troverete queste slides alla pagina

`http://poisson.phc.unipi.it/~dserra`

Un manuale di riferimento

Per preparare questo seminario mi sono ispirato alla guida di Lorenzo Pantieri & Tommaso Gordini,

L'arte di scrivere con LaTeX,

che trovate facilmente googlando "pantieri".
Troverete queste slides alla pagina

`http://poisson.phc.unipi.it/~dserra`

Grazie per l'attenzione!