

Fondamenti del linguaggio \LaTeX per la scrittura scientifica

Lezione 2: Oltre testo e formule

Andrea Di Primio (andrea.diprimio@polimi.it)

2 Dicembre 2021

Warm-up: il pacchetto `amsthm`

Cominciamo a programmare!

Create un nuovo file su T_EXStudio e salvatelo in una cartella dedicata come Lezione2.tex.

```
% Lezione2.tex
\documentclass[10pt, a4paper]{article}
\usepackage[italian]{babel}
\usepackage{amsmath, amssymb, amsthm}
\usepackage[top=2cm, bottom=2cm, left=2cm, right
            =2cm]{geometry}
\begin{document}
    contenuto...
\end{document}
```

Il pacchetto `amsthm`, parte 1: a cosa serve?

Il pacchetto `amsthm` fornisce ambienti dedicati per la corretta impaginazione di parti di testo contenenti, ad esempio, definizioni, teoremi, dimostrazioni, ad esempio:

Teorema (Pitagora)

Siano a, b e c le lunghezze dei lati di un triangolo rettangolo tali che $c > a$ e $c > b$. Allora si ha che

$$c^2 = a^2 + b^2.$$

Il pacchetto `amsthm`, parte 1: a cosa serve?

Il pacchetto `amsthm` fornisce ambienti dedicati per la corretta impaginazione di parti di testo contenenti, ad esempio, definizioni, teoremi, dimostrazioni, ad esempio:

Teorema (Pitagora)

Siano a, b e c le lunghezze dei lati di un triangolo rettangolo tali che $c > a$ e $c > b$. Allora si ha che

$$c^2 = a^2 + b^2.$$

Il pacchetto `amsthm`, parte 2: creazione di ambienti-teorema

Aggiungendo **nel preambolo** il comando

```
\newtheorem{envname}{label}[numbering]
```

è possibile creare un ambiente personalizzato, di nome `envname`, la cui etichetta è `label`.

Nel corpo, potremo utilizzare l'ambiente

```
\begin{envname}[testo opzionale...]  
    Contenuto...  
\end{envname}
```

Il pacchetto `amsthm`, parte 2: creazione di ambienti-teorema

Aggiungendo **nel preambolo** il comando

```
\newtheorem{envname}{label}[numbering]
```

è possibile creare un ambiente personalizzato, di nome `envname`, la cui etichetta è `label`.

Nel corpo, potremo utilizzare l'ambiente

```
\begin{envname}[testo opzionale...]  
    Contenuto...  
\end{envname}
```

Il pacchetto `amsthm`, parte 3: l'ambiente `proof`

L'ambiente `proof` serve a formattare correttamente una dimostrazione.

Esempio:

```
\begin{proof}
  Lasciata al lettore come utile esercizio.
\end{proof}
```

Dimostrazione.

Lasciata al lettore come utile esercizio.



Il pacchetto `amsthm`, parte 3: l'ambiente `proof`

L'ambiente `proof` serve a formattare correttamente una dimostrazione.

Esempio:

```
\begin{proof}
  Lasciata al lettore come utile esercizio.
\end{proof}
```

Dimostrazione.

Lasciata al lettore come utile esercizio.



Esercizio 1: il nostro primo teorema (e sua dimostrazione)

Replicare, in una sezione numerata, il seguente output:

Proposizione (Vitali)

Se $A \subset \mathbb{R}^n$ è un insieme aperto, allora esiste $B \subset A$ non misurabile secondo Lebesgue.

Dimostrazione (solo un commento).

Una costruzione dell'insieme B , detto insieme di Vitali, non è banale e richiede l'uso dell'assioma della scelta. □

Esercizio guidato!

Programmate seguendo le istruzioni fornite! Attenzione: l'output nella classe `article` sarà graficamente diverso da quello su questa slide.

Il pacchetto amsthm, parte 4: commenti finali

- La variante `\newtheorem*` crea un ambiente-teorema non numerato (stessi argomenti, eccetto `numbering`).
- Aggiungendo `\theoremstyle{keyword}` è possibile prescrivere la formattazione del testo dell'ambiente creato.

```
% Preambolo
\theoremstyle{plain} % plain: contenuto in
    corsivo, label in grassetto
\newtheorem{prop}{Proposizione}[section]
\theoremstyle{definition} % definition:
    contenuto normale, label in grassetto
\newtheorem{def}{Definizione}[section]
```

Le keyword di `\theoremstyle{keyword}` **non** sono arbitrarie!

Il pacchetto `amsthm`, parte 4: commenti finali

- La variante `\newtheorem*` crea un ambiente-teorema non numerato (stessi argomenti, eccetto `numbering`).
- Aggiungendo `\theoremstyle{keyword}` è possibile prescrivere la formattazione del testo dell'ambiente creato.

```
% Preambolo
\theoremstyle{plain} % plain: contenuto in
    corsivo, label in grassetto
\newtheorem{prop}{Proposizione}[section]
\theoremstyle{definition} % definition:
    contenuto normale, label in grassetto
\newtheorem{def}{Definizione}[section]
```

Le keyword di `\theoremstyle{keyword}` **non** sono arbitrarie!

Floating objects (o la causa di numerose imprecisioni)

Cosa c'è oltre a testo e formule?

Un documento complesso non è composto solo da testo e/o formule. Infatti, esistono altri elementi come figure o tabelle.

Con essi, arrivano i problemi di impaginazione (e con essi, a loro volta, le imprecisioni).

Cosa c'è oltre a testo e formule?

Un documento complesso non è composto solo da testo e/o formule. Infatti, esistono altri elementi come figure o tabelle.

Con essi, arrivano i problemi di impaginazione (e con essi, a loro volta, le imprecazioni).

Vengono chiamati *floating objects* (o semplicemente *floats*) oggetti con tre caratteristiche principali:

1. non possono essere spezzati su più pagine;
2. il loro posizionamento viene gestito separatamente dal resto del documento;
3. possiedono una didascalia descrittiva (chiamata *caption*).

Ricadono in questa definizione, principalmente, figure e tabelle.

Vengono chiamati *floating objects* (o semplicemente *floats*) oggetti con tre caratteristiche principali:

1. non possono essere spezzati su più pagine;
2. il loro posizionamento viene gestito separatamente dal resto del documento;
3. possiedono una didascalia descrittiva (chiamata *caption*).

Ricadono in questa definizione, principalmente, figure e tabelle.

STOP DOING MATH

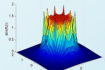
- NUMBERS WERE NOT SUPPOSED TO BE GIVEN NAMES
- YEARS OF COUNTING yet NO REAL-WORLD USE FOUND for going higher than your FINGERS
- Wanted to go higher anyway for a laugh? We had a tool for that: It was called "GUESSING"
- "Yes please give me ZERO of something. Please give me INFINITY of it" - Statements dreamed up by the utterly Deranged

LOOK at what Mathematicians have been demanding your Respect for all this time, with all the calculators & abacus we built for them

(This is REAL Math, done by REAL Mathematicians):




?????



??????



????????????????

"Hello I would like  apples please"

They have played us for absolute fools

Figure 1: Un meme rubato. Credits: @welcometomymemepage

Per la gestione dei float, aggiungiamo due pacchetti nel preambolo:

```
\usepackage{graphicx, float} % per la gestione  
    dei floating objects  
\usepackage{caption} % per la gestione delle  
    didascalie
```

Floating objects, parte 4: struttura di un ambiente per float

```
\begin{floatname}[positioning]  
    contenuto del float...  
    \caption{didascalia...}  
\end{floatname}
```

Il parametro `positioning` può avere i seguenti valori:

- `t`, top, in cima alla pagina;
- `b`, bottom, in fondo alla pagina;
- `p`, page, in una nuova pagina dedicata solo a floats;
- `H`, here, nella posizione indicata nel codice.

Floating objects, parte 4: struttura di un ambiente per float

```
\begin{floatname}[positioning]  
    contenuto del float...  
    \caption{didascalia...}  
\end{floatname}
```

Il parametro `positioning` può avere i seguenti valori:

- t, top, in cima alla pagina;
- b, bottom, in fondo alla pagina;
- p, page, in una nuova pagina dedicata solo a floats;
- H, here, nella posizione indicata nel codice. (e che vuol dire?)

Warning!

L^AT_EX è tristemente noto per la sua particolare propensione a posizionare le immagini come vuole. Tuttavia, [H] è spesso un salvavita, al contrario del suo predecessore [h].

Dalla documentazione del pacchetto float:

It seems that the [h] specifier should do that [inserire immagini nel punto indicato nel codice], but in fact it only suggests to L^AT_EX something along the lines of “put the float here if it’s OK with you”. As it turns out, L^AT_EX hardly ever feels inclined to actually do that. This situation can be improved by judicious manipulation of float style parameters.

Warning!

L^AT_EX è tristemente noto per la sua particolare propensione a posizionare le immagini come vuole. Tuttavia, [H] è spesso un salvavita, al contrario del suo predecessore [h].

Dalla documentazione del pacchetto float:

It seems that the [h] specifier should do that [inserire immagini nel punto indicato nel codice], but in fact it only suggests to L^AT_EX something along the lines of “put the float here if it’s OK with you”. As it turns out, L^AT_EX hardly ever feels inclined to actually do that. This situation can be improved by judicious manipulation of float style parameters.

Immagini, parte 1: l'ambiente figure

L'ambiente per impaginare l'inserimento di un'immagine è `figure`. Inserendo un'immagine (e.g. un file `.png` o `.jpg`) nella stessa cartella in cui è contenuto il file `.tex`, possiamo inserire l'immagine con il comando

```
\includegraphics[options]{filename}
```

Esempio:

```
\begin{figure}[positioning]
    \includegraphics[options]{filename}
    \caption{Una figura.}
\end{figure}
```


Immagini, parte 1: l'ambiente figure

L'ambiente per impaginare l'inserimento di un'immagine è `figure`. Inserendo un'immagine (e.g. un file `.png` o `.jpg`) nella stessa cartella in cui è contenuto il file `.tex`, possiamo inserire l'immagine con il comando

```
\includegraphics[options]{filename}
```

Esempio:

```
\begin{figure}[positioning]
    \includegraphics[options]{filename}
    \caption{Una figura.}
\end{figure}
```

Le opzioni del comando `includegraphic` sono:

- `width` - Imposta la larghezza dell'immagine.
- `height` - Imposta la lunghezza dell'immagine.
- `keepaspectratio` - Se impostato (basta scriverlo nelle opzioni), \LaTeX cerca di riempire più spazio possibile mantenendo le proporzioni dell'immagine.
- `scale` - Riscalda l'immagine del fattore assegnato.
- `angle` - Ruota l'immagine in senso antiorario dei *gradi* assegnati.
- `trim` - Taglia l'immagine lungo i quattro bordi.

Immagini, parte 3: sotto-immagini

Per incasellare più immagini nella stessa figura, includiamo il pacchetto subcaption.

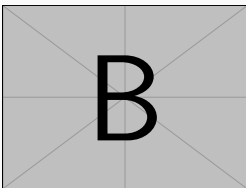
```
% aggiungere \usepackage{subcaption} nel preambolo
\begin{figure}
  \begin{subfigure}[t]{0.3\textwidth}
    \includegraphics[width=\textwidth]{example-image-a}
    \caption{Immagine A.}
  \end{subfigure}
  \begin{subfigure}[t]{0.3\textwidth}
    \includegraphics[width=\textwidth]{example-image-b}
    \caption{Immagine A.}
  \end{subfigure}
  \begin{subfigure}[t]{0.3\textwidth}
    \includegraphics[width=\textwidth]{example-image-c}
    \caption{Immagine A.}
  \end{subfigure}
  \caption{Immagini A, B e C.}
\end{figure}
```

Immagini, parte 3: sotto-immagini

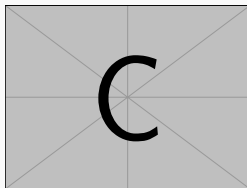
Per incasellare più immagini nella stessa figura, includiamo il pacchetto `subcaption`.



(a) Immagine A.



(b) Immagine B.



(c) Immagine C.

Figura 2: Immagini A, B e C.

L'ambiente subfigure ha la seguente struttura

```
\begin{subfigure}[positioning]{width}  
    % contenuto...  
    \caption{contenuto...}  
\end{subfigure}
```

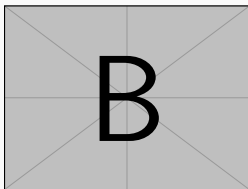
Il parametro width impone la larghezza della sottofigura.

Immagini, parte 4: l'ambiente subfigure in dettaglio

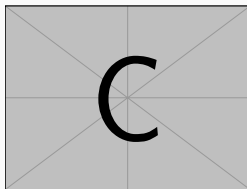
Il parametro `positioning` (`t`, `b` o `c`) impone *il posizionamento relativo delle caption delle sottofigure*.



(a) Una caption più lunga per capire meglio il posizionamento relativo.



(b) Una caption più lunga, ma diversamente dalla prima figura, per capire meglio il posizionamento relativo.

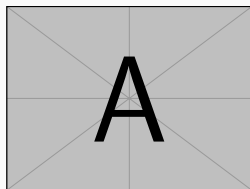


(c) Caption corta a fine trittico.

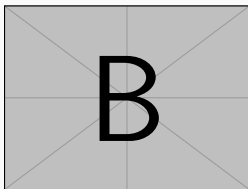
Figura 3: Sottofigure con posizionamento `t`

Immagini, parte 4: l'ambiente subfigure in dettaglio

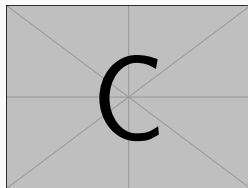
Il parametro `positioning` (`t`, `b` o `c`) impone *il posizionamento relativo delle caption delle sottofigure*.



(a) Una caption più lunga per capire meglio il posizionamento relativo.



(b) Una caption più lunga, ma diversamente dalla prima figura, per capire meglio il posizionamento relativo.

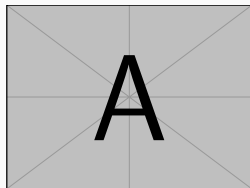


(c) Caption corta a fine trittico.

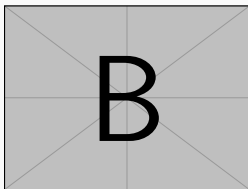
Figura 4: Sottofigure con posizionamento `c`

Immagini, parte 4: l'ambiente subfigure in dettaglio

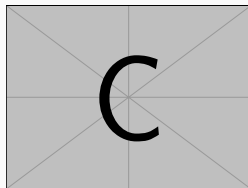
Il parametro `positioning` (`t`, `b` o `c`) impone *il posizionamento relativo delle caption delle sottofigure*.



(a) Una caption più lunga per capire meglio il posizionamento relativo.



(b) Una caption più lunga, ma diversamente dalla prima figura, per capire meglio il posizionamento relativo.



(c) Caption corta a fine trittico.

Figura 5: Sottofigure con posizionamento `b`

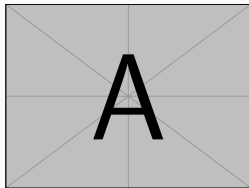
Immagini, parte 5: figure spezzate su più pagine

L^AT_EX tenta di posizionare le immagini cercando di ottimizzare lo spazio disponibile. In **nessun caso** la figura sarà spezzata su più pagine. Se è però necessario, si può fare "manualmente":

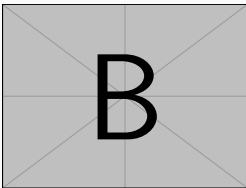
```
\begin{figure}[H]
    % contenuto della prima parte dell'
    ambiente, ad esempio alcune
    subfigures...
\end{figure}
% Cambio pagina!
\begin{figure}[H]\ContinuedFloat % <-- nota bene
    % contenuto della seconda parte dell'
    ambiente
    \caption{contenuto...}
\end{figure}
```

Esercizio 2: un album fotografico

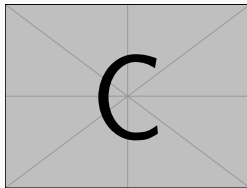
Replicare il seguente output:



(a) Helsinki, 1915.



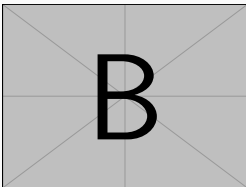
(b) Ivrea, 1901.



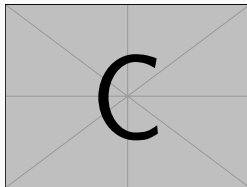
(c) Dublino, 1899.



(d) Dresda, 1914.



(e) Empoli, 1901.



(f) Napoli, 1916.

Figura 6: Belle città!

Warning!

Non catturate da schermo i plot di MATLAB!

Dopo avere settato i parametri di `figure()`, usate il comando

```
figure
...
print('filename', '-depsc')
```

per esportare il file `filename.eps`. Importatelo su \LaTeX con `\includegraphics`! Vantaggi: grafica vettoriale, sfondo trasparente.

Warning!

Non catturate da schermo i plot di MATLAB!

Dopo avere settato i parametri di `figure()`, usate il comando

```
figure
...
print('filename', '-depsc')
```

per esportare il file `filename.eps`. Importatelo su \LaTeX con `\includegraphics`! Vantaggi: grafica vettoriale, sfondo trasparente.

Tabelle, parte 1: un esempio

Val 1	Val 2	Val 3
12	48	132
	3.1415	Hello!
2.71	abc	
α	β	γ

Tabella 1: Una tabella.

Tabelle, parte 1: un esempio

```
\begin{table}[H]
  \begin{tabular}{l|c|r}
    \textbf{Val 1} & & \\
    \textbf{Val 2} & & \\
    \textbf{Val 3} & & \\
    \hline \hline
    12 & 48 & 132 \\
    & 3.1415 & Hello! \\
    \hline
    2.71 & abc & \\
    $\alpha$ & $\beta$ & $\gamma$ \\
  \end{tabular}
  \caption{Una tabella.}
\end{table}
```

L'ambiente `tabular` racchiude gli effettivi contenuti della tabella.

L'argomento obbligatorio è una successione di `l` (left), `r` (right) e `c` (center), i quali possono essere divisi da una stanghetta verticale.

Serve un carattere *per colonna*, ed ogni carattere indica l'allineamento del contenuto nella cella.

Colonne e righe sono divise da `&` e `\\`, come di consueto. Le righe orizzontali sono prodotte con il comando `\hline`.

```
\caption{Una tabella.}
```

All'interno della caption è opportuno inserire una breve descrizione del float. **Non** è necessario indicare la natura del float, \LaTeX gestisce automaticamente nomenclatura e numerazione.

Attraverso il pacchetto `multirow` è possibile creare tabelle le cui celle non hanno grandezze uniformi.

Value 1	Value 2	Value 3
α	β	γ
12	1.33	a
	10.1	b
Bye bye!		

Tabella 2: Tabella con multicelle.

Il comando `\multirow{number of rows}{width}{content}` permette di creare una cella multiriga. Inserire `*` al posto di `width` per lasciare a $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ il compito di dimensionare la cella.

Il comando

`\multicolumn{number of columns}{alignment}{content}` crea una cella multicolonna. L'argomento `alignment` specifica l'allineamento del contenuto della multicolonna (linee verticali indicabili con `l`).

Tabelle, parte 4: multicelle

```
\begin{table}[H]
  \begin{tabular}{|l|c|r|}
    \textbf{Value 1} & \textbf{Value 2} & \textbf{Value 3}\\
    $\alpha$ & $\beta$ & $\gamma$ \\
    \hline
    \multirow{2}{*}{12} & 1.33 & a\\
    & 10.1 & b\\
    \hline
    \multicolumn{3}{|c|}{Bye bye!} \\
    \hline
  \end{tabular}
  \caption{Tabella con multicelle.}
\end{table}
```

Tabelle, parte 5: il pacchetto booktabs

Il pacchetto booktabs mette a disposizione qualche ulteriore comando. Ad esempio, è possibile personalizzare le linee orizzontali con `\toprule[width]`, `\midrule[width]` e `\bottomrule[width]`.

```
\begin{tabular}{ccc}  
  \toprule  
  \textbf{Val 1} & &  
  \textbf{Val 2} & &  
  \textbf{Val 3} & \&  
  \midrule  
  12 & 48 & 132 \&  
  \bottomrule[0.1 cm]  
\end{tabular}
```

Val 1	Val 2	Val 3
12	48	132

Esercizio 3: una tabella un po' particolare

Replicare il seguente output:

Col. 1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6
$\frac{2}{3}, \frac{1}{2}, \dots$	β	γ	δ^ξ	$\sin \lambda$	$\log \pi$
Tre righe!	a	b	c	d	e
	f	g	h	i	j
	k	l	m	n	o
E questa?		Lascio aperto qui →			
→ e rientro qui!		bottomrule da 1 mm ↓			

Tabella 3: Una tabella un po' particolare.

In \LaTeX , esistono due tipi di lunghezze macro:

- lunghezze *fisse*, ovvero tutte le lunghezze che rappresentano una precisa quantità di spazio, a prescindere dal contesto;
- lunghezze *elastiche*, ovvero il cui valore è determinato da ciò che le circonda. Il valore è calcolato per riempire tutto lo spazio disponibile.

`\hfill` è una lunghezza elastica, dice a \LaTeX : "riempi tutto lo spazio orizzontale che puoi".

Introduciamo qualche altra lunghezza macro fissa utile:

- `\textwidth`: larghezza tra i margini verticali del foglio (non la larghezza del foglio);
- `\textheight`: altezza tra i margini orizzontali del foglio (non l'altezza del foglio);

Il pacchetto `layout` fornisce il comando `\layout`, che permette di visualizzare tutte le *fixed macro* relative alle dimensioni del foglio (try it!).

Introduciamo qualche altra lunghezza macro fissa utile:

- `\hspace{width}`: generica lunghezza orizzontale;
- `\vspace{height}`: generica lunghezza verticale.

Se i loro argomenti sono lunghezze fisse, generano lunghezze fisse. Possono essere rese elastiche inserendo come parametro `\stretch{factor}`. Il parametro `factor` indica la grandezza relativa allo spazio generato da `\stretch{1}`, e dimensiona opportunamente più lunghezze elastiche insieme.

Quando si vogliono giustapporre due o più float di diversa natura, oppure float e testo, o semplicemente quando si vuole dividere la pagina in sottoparti (e.g. due colonne), si può usare l'ambiente minipage:

```
\begin{minipage}[positioning][height][inner-pos]{width}  
    contenuto...  
\end{minipage}
```

L'argomento obbligatorio width esprime la larghezza della minipage (fissa).

Questioni di impaginazione, parte 2: l'ambiente minipage

Quando si vogliono giustapporre due o più float di diversa natura, oppure float e testo, o semplicemente quando si vuole dividere la pagina in sottoparti (e.g. due colonne), si può usare l'ambiente minipage:

```
\begin{minipage}[positioning][height][inner-pos]{width}  
    contenuto...  
\end{minipage}
```

L'argomento `positioning` esprime la posizione della minipage rispetto a ciò che la accosta.

- `c` - Allineamento al centro della minipage (default).
- `t` - Allineamento alla prima riga.
- `b` - Allineamento all'ultima riga.

Quando si vogliono giustapporre due o più float di diversa natura, oppure float e testo, o semplicemente quando si vuole dividere la pagina in sottoparti (e.g. due colonne), si può usare l'ambiente minipage:

```
\begin{minipage}[positioning][height][inner-pos]{width}  
    contenuto...  
\end{minipage}
```

L'argomento `height` esprime l'altezza della minipage (fissa, default quanto possibile).

Questioni di impaginazione, parte 2: l'ambiente minipage

Quando si vogliono giustapporre due o più float di diversa natura, oppure float e testo, o semplicemente quando si vuole dividere la pagina in sottoparti (e.g. due colonne), si può usare l'ambiente minipage:

```
\begin{minipage}[positioning][height][inner-pos]{width}  
    contenuto...  
\end{minipage}
```

L'argomento inner-pos esprime il posizionamento dei contenuti della minipage.

- t - Allineato in cima alla minipage.
- c - Allineato al centro della minipage.
- b - Allineato in fondo alla minipage.
- s - Riempie quanto più spazio possibile utilizzando lunghezze elastiche verticali.

Questioni di impaginazione, parte 2: l'ambiente minipage

```
\begin{minipage}{0.3\textwidth}  
    Colonna 1...  
\end{minipage}  
\hfill  
\begin{minipage}{0.3\textwidth}  
    Colonna 2...  
\end{minipage}  
\hfill  
\begin{minipage}{0.3\textwidth}  
    Colonna 3...  
\end{minipage}
```

Colonna 1

Colonna 2

Colonna 3

Questioni di impaginazione, parte 3: qualche avvertimento

1. \LaTeX cerca di inserire le minipage una accanto all'altra. Se è ciò che volete, assicuratevi che non siano troppo larghe: il totale delle `width` lungo una riga deve stare sotto `\textwidth`.
2. Le minipage non sono, in teoria, float environments. Tuttavia, il magico pacchetto `float` permette di inserire al loro interno figure e tabelle con il posizionamento H.
3. Le lunghezze macro cambiano in base a dove sono chiamate. Ad esempio, la `\textwidth` di una minipage è pari alla sua larghezza!
4. Impaginazioni complesse in \LaTeX sono essenzialmente dedizione, sofferenza e compromesso.

Questioni di impaginazione, parte 3: qualche avvertimento

1. \LaTeX cerca di inserire le minipage una accanto all'altra. Se è ciò che volete, assicuratevi che non siano troppo larghe: il totale delle `width` lungo una riga deve stare sotto `\textwidth`.
2. Le minipage non sono, in teoria, float environments. Tuttavia, il magico pacchetto `float` permette di inserire al loro interno figure e tabelle **con il posizionamento H**.
3. Le lunghezze macro cambiano in base a dove sono chiamate. Ad esempio, la `\textwidth` di una minipage è pari alla sua larghezza!
4. Impaginazioni complesse in \LaTeX sono essenzialmente dedizione, sofferenza e compromesso.

Questioni di impaginazione, parte 3: qualche avvertimento

1. \LaTeX cerca di inserire le minipage una accanto all'altra. Se è ciò che volete, assicuratevi che non siano troppo larghe: il totale delle `width` lungo una riga deve stare sotto `\textwidth`.
2. Le minipage non sono, in teoria, float environments. Tuttavia, il magico pacchetto `float` permette di inserire al loro interno figure e tabelle **con il posizionamento H**.
3. Le lunghezze macro cambiano in base a dove sono chiamate. Ad esempio, la `\textwidth` di una minipage è pari alla sua larghezza!
4. Impaginazioni complesse in \LaTeX sono essenzialmente dedizione, sofferenza e compromesso.

Questioni di impaginazione, parte 3: qualche avvertimento

1. \LaTeX cerca di inserire le minipage una accanto all'altra. Se è ciò che volete, assicuratevi che non siano troppo larghe: il totale delle `width` lungo una riga deve stare sotto `\textwidth`.
2. Le minipage non sono, in teoria, float environments. Tuttavia, il magico pacchetto `float` permette di inserire al loro interno figure e tabelle **con il posizionamento H**.
3. Le lunghezze macro cambiano in base a dove sono chiamate. Ad esempio, la `\textwidth` di una minipage è pari alla sua larghezza!
4. Impaginazioni complesse in \LaTeX sono essenzialmente dedizione, sofferenza e compromesso.

Questioni di impaginazione, parte 4: allineamento

Gli ambienti `flushleft`, `center`, `flushright` forzano l'allineamento del loro contenuto a sinistra, al centro ed a destra rispettivamente. Con un `float`, usarli dentro al loro ambiente più esterno.

Esempio:

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

Questioni di impaginazione, parte 4: allineamento

Gli ambienti `flushleft`, `center`, `flushright` forzano l'allineamento del loro contenuto a sinistra, al centro ed a destra rispettivamente. Con un `float`, usarli dentro al loro ambiente più esterno.

Esempio:

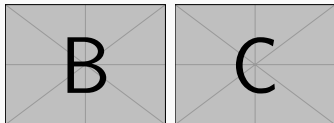
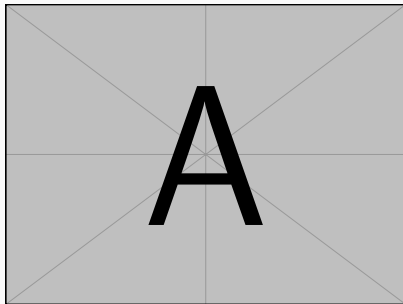
```
\begin{figure}  
    \begin{center}  
        \includegraphics{...}  
    \end{center}  
\end{figure}
```

Esercizio 4: l'occhio vuole la sua parte!

Replicare il seguente output:

La prima lettera dell'alfabeto è A. Altre lettere includono la B e la C, meno frequenti, illustrate di seguito.

Appare, nella lingua italiana, con una frequenza del 11.74% circa (fonte Wikipedia).



La tabella riassume le loro frequenze alla seconda cifra decimale:

A	11.74%
B	0.92%
C	4.50%

Grazie dell'attenzione!

Prossima lezione: 9/12/2021 in 3.1.6